RECENT TECHNICAL ADVANCES IN THE COMPUTER INDUSTRY AND THEIR FUTURE IMPACT

by

Stuart E. Madnick

February 1973 645-73

INTRODUCTION

The past two decades have brought tremendous advances in computer systems. In the next 5-10 years we can expect advances of even greater scope. These changes can be sub-divided into three categories: (1) technological cost/performance* breakthroughs in computer manufacturing, (2) evolution of computer system architecture, both hardware and software, and (3) major steps toward meeting user requirements and capabilities. In this report recent work in these three areas will be reviewed and projected future developments discussed** Key problem areas are identified. It is shown that dramatic changes in computer usage and structure are required to reap the full potential of these advances.***

---

* Cost and price figures are provided as examples only. Real costs are highly volatile, difficult to determine, and strongly influenced by production volume Prices are determined usually by market conditions and have little technical significance. Prices may be excessively high (if market exists) or even below "cost" (to develop market).

** Much of the research discussed in this paper has not been fully reported in the published literature. As an aid to the interested reader, specific references are made to universities, research groups, and companies active in these fields. These references, though carefully selected, are not necessarily complete nor should they be viewed as a recommendation of the referenced organization.

*** Comments and criticism on the details and conclusions presented in this paper are appreciated by the author.

I. TECHNOLOGICAL COST/PERFORMANCE BREAKTHROUGHS

Technological developments in computer system hardware can be anticipated to have significant effects in many areas such as cost, performance, size, and reliability. Since there is extensive activity in computer technology, only major trends will be singled out.

1. Microprocessors

The development of semiconductor microprocessors is likely to have tremendous impact in many directions. A microprocessor can be defined as a small computer processor that consists of one or two semiconductor integrated circuits (large-scale integration, LSI).[1] Such a processor's size is measured in inches. A slightly simplified form of this technology has sparked the recent growth of hand-held electronic calculators, such as Hewlett-Packard's sophisticated HP-35.

There are no serious technical problems to the development of microprocessors. They are typically based upon standard MOS/LSI (Metal-Oxide-Silicon Large-Scale-Integration) technology. The present problems are largely customer acceptance.[2] The manufacturing cost is very sensitive to production quantity. Due to complex engineering and pruduction set-up requirements, it cost only slightly more to manufacture 10,000 microprocessors as it would to produce 1000. (The joke that "the only raw material required is a few shovels of beach sand", though not technically true, indicates the situation). The marketing problem becomes quite apparent when one recalls that after 25 years there are still less than 200,000 conventional computers in the world.

In spite of these problems, the participants in the microprocessor sweepstakes reads like a Who's Who of the semiconductor industry:[1] Intel (MCS-4 and MCS-8), Fairchild (PPS-25 programmed processor), American Micro - Systems (7200 processor), National Semiconductor (GPC/P - General Purpose Controller/Processor and MAPS - Microprogrammable arithmetic processor system). Companies enter and leave and re-enter this list continually. The mainframe computer manufacturers, of course, are also pursuing the microprocessor but they prefer to maintain a much lower profile.

Characteristics of Microprocessors

There are many differences among the various microprocessor approaches. The Intel MCS-4, one of the earliest and simplest, will be described. The MCS-4 is a four-bit parallel processor with 45 instructions and a 10.8 microsecond instruction cycle (i.e. can execute almost 100,000 instructions per second).* For comparison, a conventional computer, such as IBM's 370/135, has a 16-bit parallel processor, over 100 instructions, and an instruction cycle around 1 microsecond or below (i.e. about 1,000,000 instructions per second). The microprocessor does have important points in the cost department.

* A conventional computer, such as IBM's 370/135, usually comes as a complete ready-to-use system including memory, I/O devices, and software.

The heart of the MCS-4, the 4004 control and arithmetic unit, sells for $30 each in 100-unit lots. The entire 4-chip MCS-4 system sells for $48 in 100-unit lots.

The long range characteristics are difficult to project. On the negative side, the present sales prices are probably artificially low since none of the manufacturers are in full-scale production. Although these prices are attainable, one is tempted to recall the Viatron experience. On the positive side, there are improvements possible in all directions. Eight-bit and sixteen-bit parallel processors have been developed. At least one recent reference[3] has predicted microprocessor costs as low as $1 within 25 years with a processing rate up to 10,000,000 instructions per second , though such a long-range estimate is strictly vague conjecture.

Uses for Microprocessors

As noted above the immediate problem is in developing uses for microprocessors. Potential uses can be subdivided into three areas:

1. "Non-computer" applications
2. Small computers
3. New approaches to medium -scale computers.

1. "Non-Computer" Applications

By "non-computer" applications we mean integrating the microprocessor into a larger system. This has been done for years with minicomputers (computers selling for $10,000 or less) but will be accelerated by the low-cost microprocessor. The possibilities are limitless and can not be ennumerated here (see references 1 and 3). Typical applications cited include: machine-tool control, telephone switching, point-of-sale systems, medical electronics, automotive controls, digital watches, and parking garage fee computer.

2. Small computers ("personalized computers")

There are many applications where a low-performance, if low-cost, computer is quite adequate. Compared with human skills, a "slow" 100,000 arithmetic operations per second microprocessor is quite impressive. To illustrate the possible effects, it has been reported that Japan's Telephone Company has recently cancelled its low-cost touch-tone-input/audio-response-output time-sharing computer business due to the advent of powerful low-cost hand-held calculators. There are many other applications for such personalized and slightly specialized computers both in the home and at work. A considerable amount of enterprenerial initiative will be necessary to introduce these devices.

In addition to the personalized computer, microprocessors will have impact upon sophisticated ("intelligent") remote computer terminals. In fact, much of the current microprocessor activity is jointly sponsored by terminal manufacturers and the semiconductor companies. At present, microprocessors are experiencing difficulties in attaining the speeds needed to control high-performance CRT video terminals.

Finally, the larger and more powerful microprocessors could pose a threat to the current minicomputer market. These microprocessors could also evolve into small stand-alone business computers (accounts receivable, payroll, etc.)

3. Medium-scale computers (multiprocessor and multicomputer configurations)

Medium-scale computers, such as the IBM 370/135, 370/145, and 370/155 represent the major segment of the current computer market. The microprocessors and even their slightly more expensive and powerful cousins, the minicomputers, do not have the processing power nor I/O flexibility to compete with the medium-scale computer individually.

In fact, medium and large-scale contemporary computers are becoming more and more decentralized. Separate processors are being used to serve different functions, in addition to the central processors, there are I/O processors (often called channels) and special I/O device processors (often called control units). Traditionally these processors were all different, specialized according to their purpose.

As the cost of microporcessors drops and performance improve, one is tempted to use these general-purpose units in place of the many different specialized processors in a medium-scale computer. This trend is already apparent in the recent introduction of the Burroughs B1700 system, based upon the Burroughs "D" machine multiprocessor military computer, and the IBM System/370 Model 125 that uses several separate microprocessors internally.

There are many factors likely to accelerate this trend toward multiprocessor and multicomputer configurations. (Technically, "multiprocessor" implies more dependence between the processors than the term "multicomputer". For simplicity we will use the terms interchangeably). One intriguing argument is based upon the impact of the rapidly changing state-of-the-art on the development cycle. Typically, the development cycle (i.e., the time from initial product conception to full production) is about 5 years for medium scale computers. The simpler minicomputer often breezes through in 2 years, in some cases less than 12 months. Many important decisions must be made early in the design, such as system performance, circuit types, memory modules, packaging approaches, etc. The designer of the medium-scale computer must make these decisions 4 to 5 years before production. If he chooses to use their existing technology,the final product will be several years behind the state-of-the-art. If he extrapolates and projects the availability of future technology, he is likely to make some bad guesses necessitating last minute redesign and inefficiencies. The replacement of the IBM System/370 Models 155 and 165 within 2 years of initial delivery by the newer Models 158 and 168 is at least partially explainable by this phenomenon. By building medium-scale computers out of multiple general-purpose microprocessors, the final design may be slightly "less optimal". This is more than offset by the increased flexibility and ability to use more current state-of-the-art, as well as the advantages of much larger quantity mass production which dramatically reduces the cost of such a system.

Practical Considerations

The dramatic reduction in microprocessor costs must be carefully considered. On the negative side, the current manufacturing costs of a computer system represents about 25% of the sales price (the percentage is even lower for IBM).* The processor may represent only 1/3 of the system, the other 2/3 being memory and I/O devices. Finally, the processor electronics, excluding cabinetry, power supplies, etc., may represent less than 1/3 of the processor's cost. Thus, we have been talking about advances that effect less than 3% of a medium-scale computer system's price!

On the positive side, we can expect dramatic reductions in the costs of the memories and I/O devices. The miniaturized size of the processors will allow considerable reductions in cabinetry, power supplies, etc. The more

* Marketing, software development, research, and profit account for the rest of the sales price.

modular construction impacts other costly areas, such as servicing. (Would you believe disposable processors?).

Thus, we can expect tremendous advances in processor technology but the impact upon the end user will be minimal unless there are dramatic changes in these other areas.

2. Main Storage

Main storage, also called main memory or "core memory", is a major component of conventional computer systems. It typically represents about 1/3 of the system's cost. The breakthroughs in this area both attained and predicted have received considerable attention.[4,5,11,13]

Due to the very high volumes, simplicity of structure, and modularity, all of the semiconductor technology benefits of the preceeding section apply. With the announcement of the new System/370 Models 158 and 168, IBM has shifted entirely toward semiconductor main storage for all the System/370 models and away from the traditional ferrite core memory.

The semiconductor memory market is extremely competitive in both price and technology. Versions of the Intel 1103, a 1024-bit MOS semiconductor memory circuit, have become standard components and are manufactured by several companies. Due to economies of scale and a very steep "learning curve", the cost of such circuits has dropped by a factor of 10 in a little over one year. With the eventual commercial maturity of even larger semiconductor memory chips (4096 bits and above), the cost per bit of memory is likely to drop by another factor of 10 in the next few years.

We can expect future computers that have larger capacity, smaller size, and less costly main storage. As a result of the decreased size, it is likely that more memory will be packaged in the same cabinets with the processors. As an indication of this trend, the minimum size IBM 370/135 (at 98K bytes) is larger than the maximum size IBM 360/30 (at 64K bytes), its predecessor. This increased storage at lower cost will greatly reduce the problems of program development and facilitate the use of sophisticated operating systems.

General Note

The traditional distinction between high-speed main storage (e.g., ferrite core memory) and lower-speed secondary storage (e.g., magnetic disks, drums, and tapes) is rapidly disappearing and is being replaced by a more continuous storage hierarchy. The main frame storage complex of a contemporary computer system consists of various subunits, such as, control storage, program/data storage, scratchpad storage, and cache buffer storage. Each of these subunits have slightly different cost/performance requirements. The preceeding section concerning main storage and semiconductor technology still apply. In the following section we will discuss the traditional view of secondary storage; the reader should refer to the Storage Hierarchy discussion in the System Architecture Section.

3. Secondary Storage

The use of high-performance direct-access secondary storage was accelerated by the IBM System/360 introduced in 1964. In the past 8 years, this trend has boomed and is an area of considerable competitive pressure. The basic IBM 3330 Disk Storage Unit alone is expected to reach a volume of over 1 billion dollars. A single 3330 module, storing 100,000,000 bytes (characters), has more than 10 times the capacity and double the access speed of its ancestor, the IBM 2311, introduced less than 10 years ago.

Current secondary storage devices are based upon rotating magnetic media technology (e.g. magnetic disks, magnetic drums, magnetic tape stips, etc.). This technology can probably be pushed another factor of 10 in capacity and 2 in speed in the next 5 years. Beyond that point there are at least 3 limitations to the electromechanical approach: (1) As with the ferrite core main memory, we are rapidly approaching physical limitations in magnetic media recording capacity and speed, (2) due to the need for costly mechanical motors, the cost per bit of storage may decrease by increasing capacity but the unit cost continues to increase (i.e., there is no such thing as "small size" anymore), (3) the mechanical approach has inherent speed and reliability limitations.

There are numerous technologies being pursued that lead to storage devices that fall between the traditional high-cost high-performance main storage and the lower-cost low-performance electromechanical secondary storage. All of these intermediate storage approaches have been successfully demonstrated in the laboratory and, in some cases, in limited production. The most successful should be in full production within 5 years. These technologies include:

- MOS/LSI shift registers[8]
- Charge-Coupled Devices(CCD)[11,17]
- Magnetic Bubbles[6,11]
- Optical (Laser Beam) Storage[11,13]
- Electron Beam Storage
- Bucket-Brigade Devices[11,17]

The expected prices now and in 1975 are indicated below:

	Cost (¢/byte)	Typical Unit Capacity (million bytes)	Random Access Time (microseconds)
Main Storage (now)	10¢	.1M - 1M	0.6us
Main Storage (1975)	1¢	.1M - 2M	0.2us
Intermediate Storage (1975)	.01¢ - .1¢	1M - 20M	1us - 1000us
Secondary Storage (1975)	.002¢	200M	20,000us
Secondary Storage (now)	.01¢ 2¢	100M 10M	40,000us ("moving head") 5,000us ("fixed head")

It is important to note that these intermediate storage devices are both cheaper and faster than the "fixed head" secondary devices (e.g., magnetic drums). But, "fixed head" storage represents a small fraction of the secondary storage market. These new intermediate storage devices will likely coexist with the faster though more costly main storage and the slower though less costly "moving head" secondary storage. The future of these storage levels will depend upon changes in computer system architecture and applications to take maximal advantages of each level's unique characteristics.

4. Archival Storage

By standards of a decade ago, today's secondary storage devices have enormous capacity. A single 8-module IBM 3330 Disk Storage Unit has a capacity of 800 million bytes (characters). If we assume that there are about 4000 characters on a dense single-spaced 8 1/2"x 11" sheet of paper, 800 million bytes is comparable to 250,000 sheets of paper. Yet, potential information storage requirements greatly exceed this capacity. (For example, as part of its anti-trust defense, IBM had submitted over 27 million documents as of January 1973).

To satisfy these needs, there has been considerable activity in the development of archival storage devices capable of storing enormous amounts of information economically. These devices are often called Terabit Memories since they are designed to hold over 1 trillion bits of storage (1 trillion bit is about 120 billion characters or the equivalent of 40 million 8 1/2" x 11" sheets of paper). There are several commercial archival storages already on the market, including:

- Grumman's MASSTAPE[10]
- Ampex's Terabit Memory (TBM)[9]
- Precision Instruments' UNICOM[7]
- IBM's 1360 Photo-Digital Storage (PDS)[9,12]

These units provide direct access to over a trillion bits of storage with a maximum delay of a few seconds. Typical cost per byte is around .001¢/byte. In some cases, such as MASSTAPE and TBM, the information is erasable and rewritable - similar to conventional computer storage devices. In many cases, such as UNICOM and PDS, the information is permanently written and not erasable (like using ink pens!).

Even today an archival storage unit can hold the capacity of 10,000 conventional magnetic tape reels on line. The recording media is usually removable and can be stored offline similar to magnetic tape libraries, but requiring a fraction of the space.

The offline storage cost for the recording media itself drops to around .00005¢/byte. At least one installation was justified by eliminating the cost of purchasing and the space for storing thousands of reels of magnetic tape.

An 8 1/2" x 11" document, assuming 4000 characters of text again, could be stored in computerized form for 4¢ online or .2¢ offline. That is comparable to the price of paper. Thus, it may be cheaper to store information in a computer than on paper! The implications of archival storage units are not yet fully understood. Experiments in the future uses of archival storages are just beginning, such as the DATACOMPUTER[7] and TABLON[9] projects. This is an area capable of tremendous impact upon society.

The capacities, speeds, and prices discussed above are already available on the market. In the next 5 years, we can expect significant advances, especially in the laser and electron beam approaches similar to the UNICOM and PDS units.

5. Other Trends

The coming decade will bring many advances in computer-related technology. System reliability will increase due to extensive use of electronic circuits, error-checking and -correcting techniques, and economical redundancy.

The marriage between computers and communications (and their offspring - the terminals) will intensify. Computers are already being used to control communications in A.T.&T.'s Electronic Switching System (ESS). Digital communication, as contrasted with voice, is increasing rapidly. This area of tremendous potential is complicated by many factors, such as : (1) technology, (2) FCC regulations, (3) relatively inexperienced and rapidly growing competition, and (4) A.T.&T. There are numerous references on this subject, such as, "Regulatory and Economic Issues in Computer Communications" by Stuart Mathison and Philip Walker in the Nov. 1972 issue of the Proceedings of the IEEE.

II. COMPUTER SYSTEM ARCHITECTURE

In Section I we presented current and anticipated technological advances that will have significant affects upon the way that computers will appear and will be used. Another source of major change is in the area of computer system architecture and an understanding of basic concepts. As a recent conference speaker stated, "after 25 years of growth, the computer industry has reached its infancy." The early computers were primarily high-speed calculators used to generate ballistic trajectories. Systems are now being used for purposes undreamed of 25 years ago, yet the basic computer structures have not changed much over the years. Research during the past decade is about to pay off in new and more effective approaches to computer architecture.

1. Multiprogramming

The technique of multiprogramming, the interleaved execution of two or more programs, is standard on most medium and large-scale systems. The procedures presently required to accomplish multiprogramming are often awkward, require a large sophisticated operating system, and frequently introduce considerable overhead and performance degradation.

By analyzing the fundamental requirements needed to support multiprogram operation and incorporating these features into the basic computer hardware, operating systems become much simpler and efficient. Rudimentary attempts to accomplish this can be seen in the old Honeywill 800 series and the recent Singer Ten System. Far more significant approaches can be found in Venus Project at MITRE. Similar experiements exist in the advanced development laboratories of most major computer manufacturers.

Many of these multiprogramming facilities can be introduced in a compatible manner (i.e., without impacting existing user programs). But to attain even greater effectiveness, especially in a multiple microprocessor environment as explained in the preceeding section, it will be necessary to develop new programming styles. IBM's PL/I (Programming Language/One) provides some of the necessary features, but other programming languages are needed and are being developed.

2. Microprogramming and Control Hierarchies

The early computers were relatively simple, though volumenous, performing additions, subtractions, comparisons, etc. As users developed requirements for advanced mathematical processing (e.g. vector and matrix operations), extensive data base processing, and intricate problem-solving, far more sophisticated computers were desired. The microprogramming manufacturing technique makes it feasible and economical to produce such systems.

In the Venus Project, mentioned earlier, most of the traditional operating system functions have been incorporated into the basic computer hardware. This approach has also been used to greatly simplify and speed-up the operation of high-level programming languages, such as COBOL, FORTRAN and PL/I on the Burroughs 1700 system and APL on an experimental IBM system. This trend will continue on future systems thereby providing far more powerful and efficient programming facilities to the user.[14]

3. Virtual Storage and Storage Hierarchies

IBM has recently popularized the concept of "virtual storage", the automatic management and movement of information between main storage and secondary storage. Similar approaches have been used in many earlier systems by other manufacturers, such as Burroughs and RCA (now UNIVAC). Virtual storage greatly simplifies the tasks of the programmer - the major cost in application development as well as improving system performance.

The effective use of intermediate storage technologies, described in Section I, requires an automatically controlled storage hierarchy that provides a virtual storage encompassing main, intermediate, and secondary storage. Research in this approach is going on in the development laboratories as well as universities, such as MIT.[16] The current problems will probably be resolved in time to allow the use of intermediate storage devices in storage hierarchies for the next generation of computer systems. The combined effect should further reduce programming costs while increasing system efficiency.

4. Communications

One of the most significant impacts of information processing systems will be on communications, including time-sharing, centralized data bases, and computer networks.

Time-sharing, although not meeting the lofty projections of its advocates during the mid 1960's, has and will continue to have tremendous effects. Most of the earlier technical problems have been long overcome and the reduced system costs make time-sharing systems very attractive. Many people, "burnt" during the expensive time-sharing fever of the 1960's, are surprised to find that powerful multiple user online-programming/interactive-problem-solving systems are commercially available for less than $20,000/month from IBM (using the VM/370 operating system on the System/370 Models 135 or 145). More limited time-sharing systems, such as those utilizing minicomputers and restricted to the BASIC programming language, are available for a fraction of IBM's price.

The full impact of time-sharing systems has been stalled by the lack of entrepreneurial efforts in application areas. We are now beginning to see the emergence of companies that use the time-sharing concept to provide useful and convenient facilities directly to the end-user. These application areas range from online advertising media analysis, to engineering/manufacturing/production control systems to sophisticated lens design programs. Many of the larger time-sharing services companies have already found that the majority of their revenue is derived from their application (proprietary) program services rather than their traditional "raw" computer services.

The identification, development and marketing of these applications-oriented time-sharing services is a serious problem. The successes already attained indicate that these problems can, and probably will be, overcome.

As the complexity of modern-day business increases, it is necessary to place increased reliance upon computer assisted controls. New information handling concepts coupled with the economics of secondary and archival storage devices make centralized data bases feasible. Many of the earlier disasters at "online real-time total management information systems" can be attributed to the naivety of the users and implementors rather than an indictment of the basic concepts. A typical example is the confusion between the <u>currency</u> and

and responsiveness of an online data base system. In considering the location for a new warehouse, the long-range planner may wish immediate information (rapid response) on the yearly volumes handled by the present warehouses. But, it probably makes very little difference whether the volumes provided are current as of yesterday or last week! As users and designers become more realistic at identifying the real problems and requirements, we should see many more "success stories".

As the need develops for more global optimization of large systems, such as a decentralized manufacturing company, the Federal Reserve System, etc., it is necessary for the local computer systems to communicate and exchange information. Advances in this area are being pursued by projects such as the government-funded ARPANET[15] and Michigan's MERIT system. Commercial versions of these systems are already appearing on the market (e.g., Bolt,Beranek and Newman's Interface Message Processor, IMP, systems originally developed for ARPANET).

5. Protection

The topics of information system protection and security have received considerable attention in the press recently. The full implications are probably not apparent to most observers, though. It is unlikely that anyone would make a serious attempt to steal your company's payroll program or even the customer list - although it is possible. On the other hand, consider a multi-million dollar software development company whose major assets, proprietary application programs, may be represented by a single magnetic tape reel. The lack of effective technical and legal safeguards has been a major obstacle to the growth of the application-oriented time-sharing services market. Fortunately, many, though not all, of these problems have been solved.

When considering the protection of information in a computer system it is useful to divide the problems into three parts:

1. Validation (How do you keep bad information out of the system?)
2. Integrity (How do you prevent your information from being destroyed or lost?).
3. Security (How do you prevent unauthorized access to the information?).

It makes little sense to lock your information system in a lead vault guarded by Marines if the information is meaningless or incorrect. A simple example may clarify the point. Recently a New Jersey town, in preparing the data for computing the tax rate, incorrectly entered a value of $10,000,000 for one resident's house (misplaced decimal point or sleepy keypuncher). After the tax rate was computed, the mayor was pleased to announce that their town had one of the smallest rate increases in the state. Months later, when an irate resident complained of a $10,000,000 property assessment on his 8-room house, the error was uncovered. At last report, the mayor was looking at a sizable budget deficit.

Fortunately, there is considerable activity in these areas. IBM has recently initiated a 5-year $40 million joint research effort in conjunction with several university, industry, and government investigators.

III. SYSTEMS THAT ARE EASIER TO USE

In Sections I and II we presented advances that are driven primarily by technological innovation in hardware manufacturing and system architecture.

It is reasonable to ask, "what about the user?." It should be noted that many of the topics of the preceeding sections do result in increased user efficiency, effectiveness and convenience (e.g., larger main storage and virtual storage). In addition to the beneficent unselfish desire to make life more pleasant for the users, there are many important dollars-and-cents reasons that are accelerating manufacturers' and researchers'activities in this area.

If one takes the attitude that profits are unlikely to exceed revenues (and are usually much less), the underlying problem comes out quite fast. If the prices of hardware drop as anticipated, it will be possible to do the same processing next year for less than this year. One estimator claimed that the current world-wide inventory of processors and memories could be replaced at a cost of less than $1 billion by 1975. Thus, if there is not continual and massive growth of the market, the industry will stagnate and, in dollar revenues, shrink tremendously.

All the gloomy statements above have potentially applied over the past 25 years. Fortunately, the market always grew much faster than the prices could drop and the demand developed for even larger and more powerful systems. When the early ENIAC computer was built, reliable experts predicted that 100 such machines would satisfy the country's computational needs for the rest of the century; needless to say, the market was somewhat larger.

Market Bottlenecks

We will take the stand that there is enormous potential market growth for computerized processing and information systems. Instead we will identify three major bottlenecks to the growth of the market.

1. Salaries vs. Hardware

   In the development and operation of new application areas it is estimated that close to 70% of the cost is tied to humans (including salaries, office space, fringe benefits, etc.) and only about 30% to computer hardware. If the hardware costs were to drop to zero, there would be relatively little increased incentive to develop applications at a faster pace. The first bottleneck is tied to the cost of people to develop new systems.

2. Maintenance vs. Development

   In most mature data processing installations, about 80-90% of the personnel and costs are devoted to the operation and maintenance of existing applications. This leaves only about 10-20% of the budget for the development of new application areas. The second bottleneck is tied to the cost of operating existing systems.

3. User Sophistication and Education

   The two bottlenecks above relate primarily to current, relatively mature, users. Another, and even larger, market is found in the present non-users. These users are often rather small and unsophisticated with probably little education or understanding of computer systems. The third bottleneck is tied to making systems usable by the uneducated non-user.

In this section we will discuss approaches and techniques that have been developed, or are being studied, that attack these bottlenecks. We will not repeat any of the activities from Sections I and II, although some do provide facilities that ease the users burden.

1. <u>High-Level Languages (HLL) and Problem-Oriented Languages</u>

   High-level languages (HLL) and problem-oriented languages (POL) have been in use for many years. The basic concept is quite simple. The language (i.e. "machine language") of a conventional computer is awkward and tedious for human use in expressing a problem. Instead, the user expresses his problem in an "English-like" language, such as FORTRAN (FORmula TRANslator) or COBOL (COmmon Business Oriented Language), and the problem is automatically translated into machine language. The net effect is that it is easier and faster for the user to write computer programs.

The past difficulties with HLL's and POL's were: (1) It was expensive and difficult to build the automatic translators, called compilers, (2) Some HLL's and POL's were not much easier to use than machine language, and (3) Use of HLL's and POL's resulted in programs that were slower than manually translated programs. These problems have been largely overcome due to: (1) research which has resulted in techniques that produce economical, efficient and more powerful compilers,(2) the shift in cost from computer hardware to people, i.e., even if the translator is inefficient, a manual translation would be usually much more expensive and (3) the new computer architectures provide for efficient operation of HLL's and POL's.

2. Generalized Application Packages

Although high-level languages make programming easier, they still require programmers. When one looks at typical computer usage, you notice tremendous similarities. Everyone seems to want a payroll program, an inventory control program, an accounts receivables program, etc. Why should every company develop its own payroll program, ...? Why isn't there a single payroll program that every company uses? The vast majority of current programmers are working on projects that have already been done at other companies.

The problem is usually not due to company secrecy. In fact, some companies have found that the marketing of their programs has brought in considerable extra revenue. The problem, in general, is that no two programs are exactly the same, even when used for similar purposes. For example,the ubiquitous payroll program may differ due to handling of salaried vs. non-salaried employees, whether the company is intra-state or inter-state, pension plans, etc. Thus, one company's payroll program may be worthless, or at least of minimal use, to another company.

Although a given application area, such as payroll, may have thousands or hundreds of thousands of variations, there are usually a much smaller number of mutually exclusive options (e.g., salaried vs. non-salaried, etc.). It is the numerous combinations and permutations of these basic options that result in the tremendous diversity. A generalized application program attacks this problem by providing capability to handle all of the possible options. The user merely specifies which options he needs and what particular values must be used (e.g., State Tax rate). IBM has pushed this approach quite far with its Applications Customizer facility used on its small-scale System/3 computers.

This approach has made the computer usable and economical to a large market, the small users. It is still necessary to develop these generalized application packages, in fact they are usually much more costly than any single non-generalized application program. Thus, only large companies with many customers can justify the initial cost. Furthermore, as we move into areas with even more options, such as production scheduling, market analysis, etc., and totally new areas, such as medical applications, the cost of developing the generalized program increases and the market size decreases. In the meantime, the growth of entrepreneurial software companies,in addition to the present computer manufacturers, will result in considerable activity in the generalized application package area for many years to come.

3. Information Handling

The actual programming is only part of the cost of developing new systems. In order to be used and maintained, tremendous amounts of documentation must be prepared. The use of high-level languages, described above, has made some programmers so effective that it takes more people (documentors) to explain what has been done and how to use it than it took to develop the system. In such projects the documentation is a serious bottleneck.

Several approaches have been developed to combat this problem. The use of "English-like" high-level languages that are easy to understand reduces the amount of additional maintenance and design documentation that is required. Documentation is still required though. The truly "self-explanatory" program still does not exist.

Just as compilers were developed to help in the programming and translation process, various tools have been developed to aid in the documentation process. Online manuscript processing systems, such as IBM's SCRIPT/370, make it convenient for the programmer to create and update system documentation without the additional cost and delays of requiring separate secretarial services. These tools, although originally developed by computer-people for computer-people, are finding increasing interest and usage for any type of project that requires substantial amounts of documentation that is revised and updated over a long period of time (e.g.

construction projects, military projects, legal affairs, etc.).

4. Intelligent Data Bases

The computational uses of computers is continually dropping in importance compared with the data storage and data processing uses. The decreasing cost coupled with the emergence of the archival storage devices, discussed in Section I, are accelerating this trend. Many companies during the 1960's made attempts to develop "online realtime total integrated management information systems" with disasterous results as noted earlier. In many cases the entire concept and planning were poorly handled. In other cases, the existing technology did not provide the necessary requirements.

For example, if a company's personnel data base contains the name of each employee's parents, it should be possible to inquire how many father/son pairs are currently employeed by the company. In most conventional systems such a query could not be requested unless it had been anticipated or unless the system were modified to handle it. This example can be extended even further by considering the question of how many grandfather/grandson pairs are currently employed. Note that the grandfather and grandson information may not be explicitly stored in the database. But, by using the information on parents and children of each employee, the grandfather/grandson pairs can be identified. Systems that are automatically able to make these discoveries are often termed "intelligent" data base systems. There is considerable activity in this area in industry and universities, such as MIT's Sloan School of Management. These intelligent data bases will greatly enlarge the present information systems market for both large and small users.

5. Intelligent Systems and Automatic Programming

A particular generalized application package, as described earlier, is only capable of handling a specific area. Even within that area, it can only cope with a limited number of options. The intelligent data base systems are not specialized to any particular application, but they can only handle information-related requirements. They would not be able to automatically produce a complete inventory control system since that involves application-specific background knowledge. By combining the application-specific knowledge of generalized packages with the "intelligence" of intelligent data base systems, one can hope to develop an "intelligent system". Although this is definitely a very difficult objective, it is not quite science fiction. The Advanced Research Projects Agency (ARPA) of the Department of Defense is funding research in related areas, under the heading of "automatic programming", at several universities such as MIT's Project MAC Automatic Programming Division. A similar effort exists at the University of Michigan. When, or if, these projects are successful, many of the bottlenecks mentioned at the beginning of this section will be eliminated. We will likely see results materializing within 5 years.

6. Artificial Intelligence

The term "artificial intelligence" often brings to mind robots, sophisticated chess-playing computers, and other far-out sounding concepts. In recent years, the field of artificial intelligence, although still very long-range, has developed many concrete results. Two significant advances have been goal-directed programming and natural (English) language capabilities.

Goal-directed systems, such as PLANNER and CONNIVER, differ from conventional programming techniques. The user merely expresses, in reasonably precise terms, <u>what</u> he wants done rather than <u>how</u> he wants it done which is required for conventional programming languages. This approach makes it easy and convenient to build larger and more complex systems and puts much of the mechanical problem solving burden on the computer rather than the human. Although these goal-directed systems are still experimental, they have been demonstrated to be effective. In fact, they have provided much of the necessary breakthroughs needed to accomplish the automatic programming described above.

One may visualize true English as the most convenient means of communication with a computer. Although true English may not always be the most desirable approach, recent advances in artificial intelligence indicate that this objective may be attainable, especially in somewhat limited contexts. The Blocks World system, developed by Professor Winograd at MIT's Artificial Intelligence Laboratory (AI Lab), allows the user to converse with the computer talking about children's blocks (e.g., Which block is on top of the red block? etc.). Although the subject matter may be a bit childish (pardon the possible pun), researchers and computer manufacturers are exploring ways to extend these techniques to match user needs. An example would be a data base system which can truly accept real English queries rather than stilted "English-like" queries.

The full impact of the present artificial intelligence research in areas such as robots and education may be years away, but we are already benefiting from many spin-offs that will have increasing impact upon the way that we use and relate to computer systems.

CONCLUSIONS

In this report we have attempted to identify significant trends in computer system development that are likely to have substantial impact upon the ways that computers are used. In addition, the current status of these trends was reviewed and the areas of major concern have been highlighted.

References

1. Gerald Lapidus, "MOS/LSI launches the low-cost processor," IEEE Spectrum, (Nov. 1972), 33-40.

2. John Rhea, "Processor on a chip - Will it sell?, Electronic News, (Sept. 25, 1972).

3. Caxton Foster, "A View of Computer Architecture," Communications of the ACM, (July 1972), 557-565.

4. G.A. Saxton & Co., Inc., "Computer Memory Technology," Industry Note No. 12, (March 22, 1971).

5. J. K. Ayling, "Monolithic Main Memory is Taking off," 1971 IEEE International Convention Digest, (March 1971), 70-71.

6. Andrew H. Bobeck and H.E.D. Sconil, "Magnetic Bubbles", Scientific American, (June 1971), 78-90.

7. Harold R. Dell, "Design of a High Desity Optical Mass Memory System," Computer Design, (August 1971), 49-53.

8. Stephen Fields, "Silicon Disk Memories Beat Drums," Electronics, (May 24, 1971), 85-86.

9. Richard B. Gentile and Joseph R. Lucas, Jr., "The TABLON Mass Storage Network," SJCC 38, (1971), 345-356.

10. J.B. Haines, "The Evolution of the MASSTAPE System," 1972 IEEE International Convention Digest, (March 1972), 140-141.

11. Harry Howard, "Memories: Modern Day 'Musical Chairs'," EDN/EEE, (August 15, 1971), 23-31.

12. Samuel J. Penny, Robert Fink, and Margaret Alston-Garnjost, "Design of a Very Large Storage System," FJCC 37, (1970), 45-51.

13. Steve Thompson, Jack Morton and Andrew Bobeck, "Memories: Future Storage Techniques," The Electronic Engineer, (August 1971), 33-39.

14. Caxton Foster, "The Next Three Generations," Computer, (March 1972), 39-42.

15. L.C. Roberts and B.D. Wessler, "Computer Network Development to Achieve Resource Sharing," SJCC, (1970), 543-549.

16. Stuart Madnick, "Storage Hierarchies," MIT Project MAC Technical Report, (1973).

17. Jules Gilder, "3 Years After Birth, CCDs Head for First Commercial Applications", Electronic Design, (January 4, 1973), 36-40.