DOCUMENT RESUME

ED 354 863 IR 015 952

AUTHOR	Okinaka, Russell T.
TITLE	Microprocessors: An Understandable Guide for the Classroom Teacher.
PUB DATE	Jan 93
NOTE	20p.
PUB TYPE	Reports - Descriptive (141)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	Computer Peripherals; Computer Software; Desktop Publishing; Elementary Secondary Education; Hypermedia; *Microcomputers; *Microelectronics; Teachers

ABSTRACT

A microprocessor constitutes the heart and soul of a personal computer. Indeed, the quality of a personal computer is determined largely by the type of microprocessor that is included within its circuitry. Since the microcomputer revolution began in the late 1970s, these special chips have gone through a series of improvements and modifications. Today's personal computers are much faster and more efficient than the personal computers of 10 or 15 years ago. We are witnessing developments that were unthinkable in the areas of hardware and software. Graphics, multi-media, and desktop publishing are just a few of the fields that have been made possible due to superior hardware capabilities. While the developments we are seeing are amazing, we are also finding uses for these new machines as fast as they are being developed. This article explores the evolution of microprocessors. It details the changes in data path size, miniaturization, instruction sets, use of alternative materials, and other innovations. It does this in a way that is intended to be understandable for the average classroom teacher and will allow teachers to see the fundamental differences that exist between various computer platforms and models within those platforms. Data about the different models of processors and a glossary of related terms are appended. (Contains 8 references.) (Author/ALF)

Microprocessors: an understandable guide for the classroom teacher

By

Russell T. Okinaka, Ph.D.

California State University, San Bernardino

Instructional Technology Program

Department of Secondary & Vocational Education

5500 University Parkway

San Bernardino, CA 92407-2397

Abstract

A microprocessor constitutes the heart and soul of a personal computer. Indeed, the quality of a personal computer is determined largely by the type of microprocessor that is included within its circuitry. Since the microcomputer revolution began in the late 1970s, these special chips have gone through a series of improvements and modifications. Today's personal computers are much faster and more efficient than the personal computers of 10 or 15 years ago. We are witnessing developments that were unthinkable in the areas of hardware and software. Graphics, multimedia, and desktop publishing are just a few of the fields that have been made possible due to superior hardware capabilities. While the developments we are seeing are amazing, we are also finding uses for these new machines as fast as they are being developed.

This article explores the evolution of microprocessors. It details the changes in: data path size; miniaturization, instruction sets, use of alternative materials, and other innovations. It does this in a way that is intended to be understandable for the average classroom teacher, and will allow these individuals to see the fundamental differences that exist between various computer platforms and models within those platforms.

Introduction

Microcomputers have become an integral part of education. There are more teachers using the computer as a classroom tool than ever before. Many excellent products have been developed for classroom use--ranging from Computer Assisted Instruction programs (CAI), multi-media packages, and of course, utility software that can be used for managing the bureaucratic chores associated with teaching. The problem, however, is that there are also more brands and models of computers than ever before, and the task of selecting one that is appropriate can be a daunting. Companies like Apple, IBM, and host of "clone" manufacturers advertise products with features that often sound like so much "gibberish" to the typical layperson. There is one element, however, which distinguishes one model of personal computer from another, and which essentially allows a person to determine the overall quality of the computer. This element is the micro-processor.

Background

To adequately explain microprocessors, it is first necessary to describe the operation of a computer. At this point, many articles or textbooks on the subject would be bantering jargon like *bits, bytes, floating point units, random access memory,* and other such terminologies. The difficulty in understanding the operation of the computer lies not in how it actually works, but how it is explained. Essentially, a computer can be thought of as a device that accomplishes the following:

1) It receives input (information or data)
2) It processes that input (by performing calculations or manipulations on the input data)
3) It produces some sort of useful output.

(McKeown, 1987).

The advantage to using the computer is that it performs these operations very quickly, efficiently, and with great accuracy. A diagram of the theoretical operation of the computer would look like the one shown in figure 1. It should be noted that in examining the operation of the computer, there is one device which is central to the operations of the machine, and that is the Central Processing Unit (CPU), also known as the computer's *microprocessor*. The speed and efficiency of the computer is determined by how quickly the CPU is able to handle that data load.

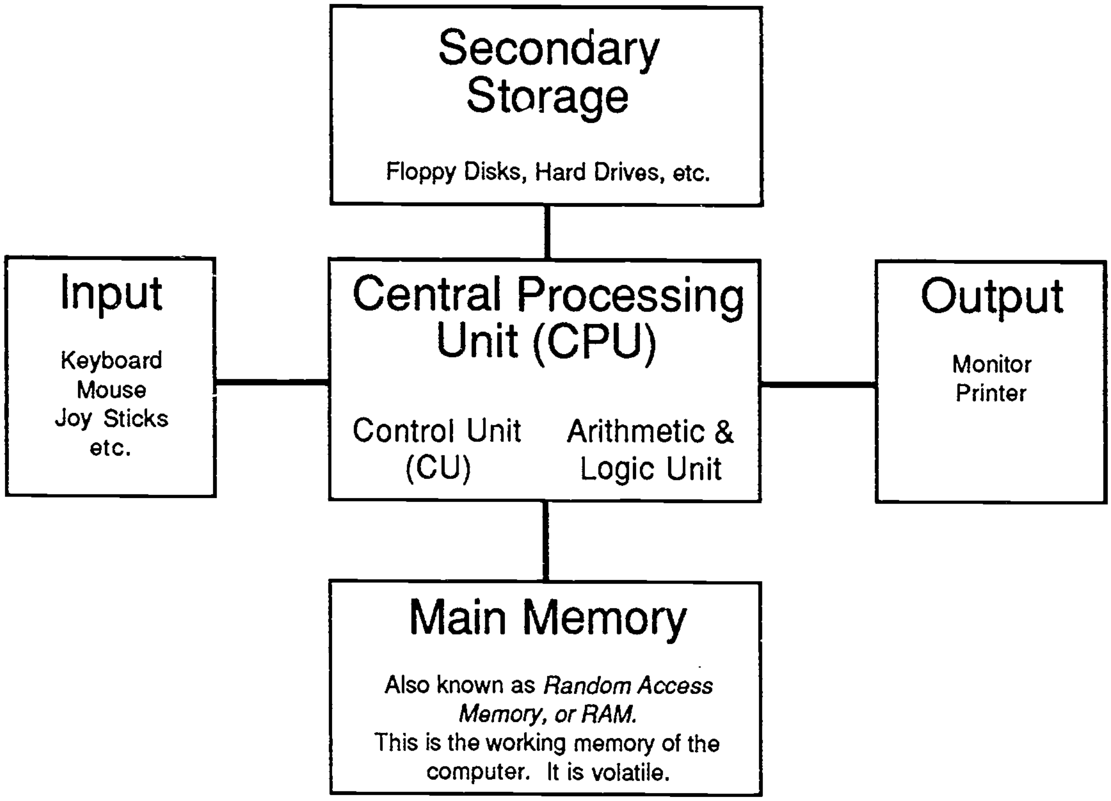

Figure 1. The theoretical operation of a computer.

It would be appropriate at this point to briefly describe each of the components comprising the "working pieces" of the computer.

1) Input. Information must be entered into the computer through the use of input devices. The most commonly used input device is the keyboard. However, there are many other input devices that have been developed over the years, such as the mouse, the light pen, OCR (which stands for Optical Character Recognition) etc.

2) Central Processing Unit (CPU). The CPU acts as the "main brain" of the computer. Its job is to process the data (information) and provide some sort of useful output. The CPU is broken up into two sub-units: the Control Unit, or CU; and the Arithmetic and Logic Unit, or ALU. The

Control Unit is responsible for ensuring that information goes to the various parts of the computer in the correct order. It can be analogized to a "traffic cop." The ALU performs the computer's math and logic operations. The ultimate speed of the processor is tied to the function of this unit. As you will see later, the ALU performs its calculations in a series of steps.

3) Main Memory. This is the working memory of the computer. Computers need to have memory in order to store data which can then be processed or manipulated. For example, as a person writes a paper using a word processing program, the information that he types in is stored in the computer's working memory. Main memory responds very quickly--that is, data can be transferred from main memory to the CPU in a small fraction of a second. Main memory does have the characteristic of being volatile. This means that if the power goes out for any reason, or if the computer is turned off, the contents of main memory are lost.

4) Secondary Storage--This generally refers to magnetic devices that are used to store information. Magnetic storage, like that of a floppy disk or a hard disk, is permanent (provided that the disks are not brought next to any type of magnetic field). Data that is stored magnetically can be retrieved later on. Data on a floppy disk or hard disk is not lost when the power is turned off.

5) Output--This is the useful information provided by the computer. Output is generally viewed on the screen, or from a printer. When you print information using a printer, the printout is referred to as a "hardcopy."

(Lockard and Abrams, 1987).

The basic operation of the computer (as shown above) has not changed significantly since the advent of electronic computers in the 1940s (Bitter, 1984; Frenzel, 1992). Of course, computers have become faster and more miniaturized, but the basic flow of information is handled in essentially the same way. It can be seen from the diagram that the *bottleneck* in the process is the CPU. Since everything is channeled through the CPU, the speed with which the computer responds to various tasks is limited by the speed of the microprocessor. Therefore, engineers and scientists over the years have attempted to increase the speed at which the microprocessor is able to function. There are several techniques that have been employed to accomplish this end.

The Size of the Data Path

It should be mentioned that computers convert all data that is entered via input devices into binary numbers. The reason why this is done is that computers are only able to store binary digits. This is because computer memory is comprised of units (transistors) which act like "switches" or "electronic gates." A switch is capable of occupying one of two positions at a given time--namely "on" or "off." Because of this simplistic configuration, computers must rely on a number system which is compatible with this "on/off" configuration. Thus, the binary number system is employed (The binary number system has two numbers--*zero* and *one*. A *zero* is equivalent to an "off" switch, and a *one* is equivalent to an "on" switch) (McKeown, 1987).

A question that often arises at this point is: "What about letters of the alphabet?" The answer is that a special code must be employed to create a one to one correspondence between letters of the alphabet, and a string (group) of binary numbers (Simonson and Thompson, 1990). Several years ago, a group of computer scientists agreed on a code known as ASCII, which stands for the American Standard Code for Information Interchange (Bitter, 1984). ASCII uses groups of eight binary numbers to represent a single character. For example:

A = 0100 0001

B = 0100 0010

C = 0100 0011

When a person types the letter "A" from the keyboard, the group of numbers "0100 0001" is actually stored in the computer's memory. In this way, the computer is able to represent all of the letters, special characters, and symbols from the keyboard.

It should be mentioned that pictures and other forms of information are ultimately "digitized," or converted into binary number equivalents as well.

While all of this is occurring, it is still the computer's microprocessor that is handling all of this information. When microcomputers first entered the market in the mid to late 1970s, standard microprocessors had what was known as an 8 bit capacity. The term *bit* stands for *binary digit*, meaning that microprocessors were handling 8 binary digits or one character at a time. Examples of computers that used 8 bit processors include the following: Commodore 64; Apple II series; Atari 400/800; Osborne; and a plethora of other brands that proliferated when the microcomputer revolution began (Bitter, 1984).

In 1981, IBM introduced its "PC" line of computers, and changed the industry by equipping this machine with a superior 16 bit processor. This meant that the processor was able to handle 16 bits at a time, or two characters simultaneously. The inclusion of this processor (known as the 8086 made by Intel Corporation) meant that the IBM was a much more powerful machine than its 8 bit predecessors. Businesses became very interested in using the IBM, because this was a computer that could handle very complex tasks more quickly and efficiently than other microcomputers. Software companies focused upon business application programs that could take advantage of the PC's superior power. As the years went by, IBM modified its PC to improve its performance, and Intel Corporation supplied the company with better 16 bit chips (McKeown, 1987; Magid, 1992).

Sixteen bit processors were later replaced with 32 bit processors (chips that could handle 4 characters at a time). Among the computers to first support 32 bit processors were the Apple MacIntosh and the Commodore AMIGA computers. The microprocessors for these machines were supplied by a company known as Motorola, which manufactured the 68000 series of processors. Once again, 32 bit processors were a significant improvement over their 16 bit predecessors (See figure 2). They could execute their tasks more quickly, and they allowed computers to perform in ways that would significantly revolutionize the way in which users interacted with their machines. Intel's first 32 bit processor was the 80386 (Magid, 1992).

Clockspeed

Besides the size of the datapath, it is important to know how many cycles a processor is able to accomplish in a given time. This helps to determine the number of calculations, and instructions the processor can handle in real time. The number of cycles per second is known as "clockspeed." In physics, the number of cycles per second is known as *frequency*. Therefore, the clockspeed is also known as the "frequency of operation" (Frenzel, 1987). Most processors are able to handle.

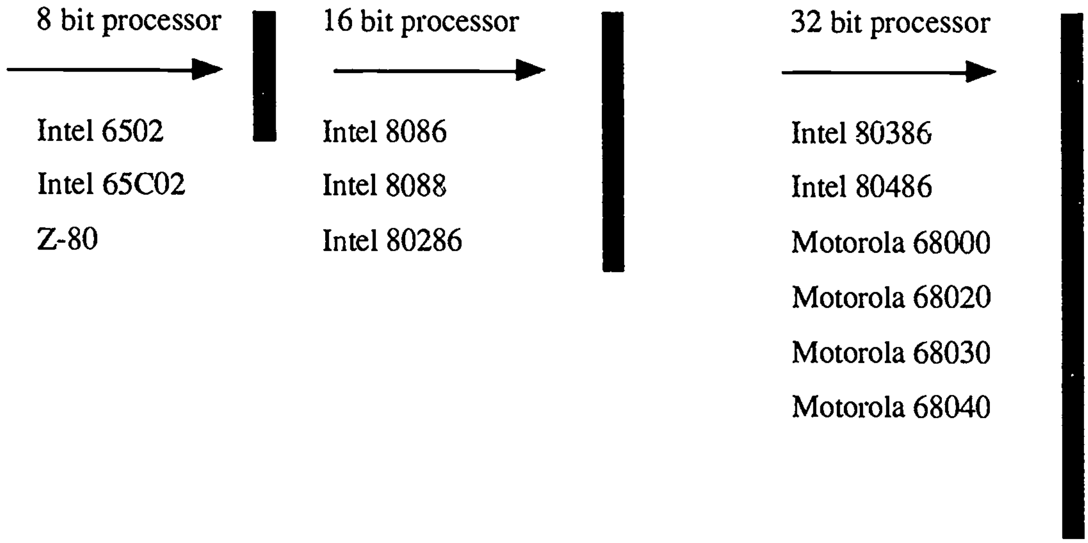

Figure 2. The size of the data path.

millions of cycles per second. The term "megahertz" is equivalent to one million cycles per second. Early 8 bit processors typically functioned at about 1 megahertz. Intel's 8086 had a clockspeed of about 4.77 megahertz when it was introduced with the first IBM PC. Many 32 bit processors today are capable of running at clockspeeds exceeding 50 megahertz. This clearly indicates that modern microprocessors are able to outperform their 8 bit ancestors many times over. Engineers have consistently grappled with the problem of increasing the clockspeed of microprocessors, and they have made some significant strides in the last decade.

The *Amount of work* accomplished per cycle

A fact that is a source of confusion for many people is that some computers have processors with seemingly higher clockspeeds than other computers, and yet get outperformed by the computers with the slower clockspeeds. An example of such a case would be a MacIntosh FX equipped with a 40 megahertz 68030 processor, and a MacIntosh Quadra equipped with a 25 megahertz 68040 microprocessor. At face value, it would seem that the MacIntosh FX should be about 60 percent faster than the MacIntosh Quadra. It turns out, however, that the Quadra runs about 33 percent faster than the FX. How can this be? The answer lies in the fact that the Quadra's 68040 is able to process larger amounts of information per cycle than the FX's 68030. You can analogize this to a pair of coal miners who are able to push their carts along the track with equal speed. However, one coal miner has a bigger cart than his counterpart. Therefore, at the end of the day, even though each miner has made 100 trips, the one with the bigger cart has accomplished more work because his cart can carry more. The bottom line is that given a processor like the 68030 running at 20 megahertz, and a "next generation" processor like the 68040 running at the same clockspeed of 20 megahertz, the "next generation" processor will do alot more work.

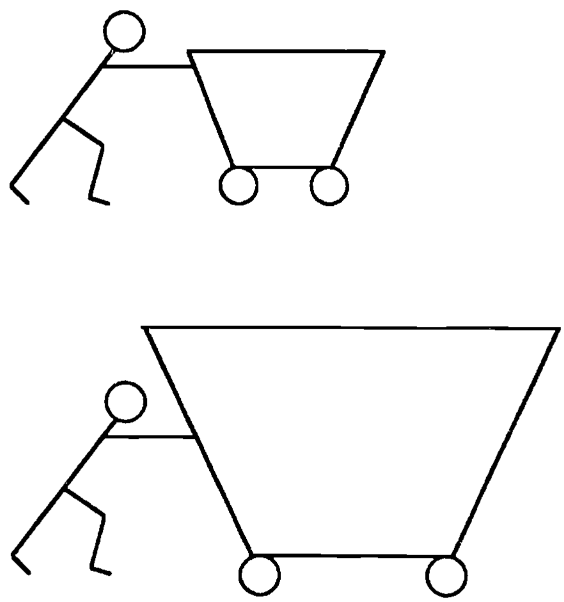

Figure 3. Two coal miners, each able to push their carts at the same speed. However, because the second coal miner has a bigger cart, he gets more work done per "trip" than his colleague. This is analogous to computer processors, in that some processors are able to handle "larger loads" per cycle than others. Therefore, even though one processor may have a higher clockspeed than another, it may actually function more slowly.

Engineers have tried to make each successive generation of processor handle more data per cycle. Thus, a Motorola 68000 processor handles less data per cycle than a Motorola 68020, which handles less data per cycle than a Motorola 68030, and so on. The same holds true for successive generations of Intel processors (Wood, 1992).

It should be mentioned that successive generations of processors have also been able to access larger and larger amounts of RAM, making it possible to run more sophisticated applications. Intel's original 8086 was able to access 640 K of RAM. By comparison, it is possible to connect an 80386 to a mother board with 32+ Megabytes of RAM. Better processors are able to access greater numbers of memory addresses.

Top of the line computers employ the most sophisticated processors--essentially meaning that they are the fastest, and most powerful. Better processors add to the cost of the machine because of superior performance. They also cost more to make (the components are usually smaller, and there are more of them). There is also a significant expenditure of time and effort in designing a new processor, which must be compensated. Therefore, you can expect to spend significantly more for a microcomputer with the best processors. (The *positive* side effect, however, is that the cost of the previous generation of processors tends to go down).

What other techniques are used to increase processor speed?

A) Miniaturization. The easiest, and perhaps most obvious way to increase the working speed of a processor (or any electric circuit for that matter), is to decrease its size. By making the circuit smaller, you decrease the distance that an electron has to travel to complete the "loop." Of course, miniaturization has other advantages, such as decreasing the overall size and weight of the machine, making it less bulky and more portable (Wood, 1992).

Decreasing the size of the chip raises a number of engineering difficulties, however. When circuit paths are miniaturized, impurities in the material comprising the chip can literally act as "road blocks," preventing the circuit from operating because they literally obstruct the path of electron flow. It therefore becomes extremely important to ensure that the material comprising the chip is as pure as possible. Computer chips are made of a material called silica, which is the primary component of beach sand. By running the chips through a special manufacturing process, it is possible to eliminate most of the impurities from the raw material. It is important to minimize impurities that can cause defects, because for every chip that has to be scrapped, the cost of making the chips goes up proportionately. Engineers throughout the world, therefore, are attempting to come up with better ways of purifying, and ensuring that the production of chips carries on with the fewest number of errors. As chips become smaller, the elimination of defects and errors becomes a problem that is significantly more complex (Wood, 1992).

B) Cache Memory. The speed of a processor can also be increased significantly by providing something known as "cache" memory. Cache memory can be thought of as "supercharged RAM." It is able to respond to the microprocessor with much greater speed than ordinary RAM (which is quite fast in itself). When a microprocessor executes instructions, or performs calculations, there are certain routines that are commonly utilized. Under ordinary circumstances (like with earlier microcomputers), these routines were stored in ordinary RAM, and called up when needed. However, ordinary RAM is much slower than cache RAM, which is engineered to respond more quickly. Computers equipped with cache memory retain the commonly used routines in this area. The processor is therefore, able to operate with greater efficiency, because the information it needs right away is readily available.

C) Gallium Arsenide. As mentioned earlier, silica has been the material of choice for computer chips. This is true largely because of its accessibility, and abundance. Recently, however, scientists and engineers have explored another semi-conductor possibility known as the *gallium arsenide chip*. Gallium arsenide has the potential to greatly improve computer efficiency, because it allows electrons to move more quickly. Some scientists estimate that the speed of computers could easily be improved 10 fold through the use of gallium arsenide semi-conductors. (Frenzel, 1987).

D) Co-processing. To relieve some of the burden that the CPU must endure, many computers now include *co-processing units*. The idea is not new--Intel introduced a math co-processor for the 8086 series of processor in the early 1980s. Essentially, a co-processor is a specialized chip whose purpose is to perform a specific type of function, like math operations, or graphics manipulations. Because the co-processor is designed *specifically* for the purpose of handling math (often referred to as floating point operations) or graphics, it performs its functions much more efficiently than the CPU, which is a more generalized unit. It should be mentioned, however, that the program must be written to access the co-processor, or it will have no effect. For example, a businessman using a spreadsheet like Lotus 1-2-3 on an IBM PC equipped with an 8086 microprocessor will find that the addition of Intel's 8087 math chip will greatly increase the performance of Lotus since the program was written to make use of the math chip if it is available. However, the same businessman using WordPerfect (a commonly used word processing program), will find no change in performance with that program, because it was not designed to access the math chip (Wood, 1992).

Co-processing is becoming a more popular idea because there are so many new programs that make use of things like graphics, animation, and sound (multi-media programs). Pictures and sound represent large amounts of data, and can prove burdensome to even the fastest processors. Therefore, if there is a co-processor available to help handle some of the work load, the overall speed of the computer can be increased significantly. A person must weigh their individual needs, however, to determine whether or not they wish to incur the expense of installing a specialized processing unit. It should be done only in instances where there is a clear need to improve performance in a specific type of application.

E) RISC (Reduced Instruction Set Chips). Historically, microprocessors have become more complex over the years. This includes their ability to execute numerous instructions. Early computers had processors with limited instruction sets, which included only the most basic operations. More complex operations were implemented by writing appropriate software routines. To illustrate this point, multiplication and division were carried out by *programs* (written software) in early computers. Today, however, most computers with their newer microprocessors have *built in* multiply and divide instructions. The most current microprocessors have literally hundreds of different instruction sets, and are therefore known as Complex Instruction Set Chips (CISC). The design of these chips is a time consuming and cumbersome task. Each new generation of chip requires further miniaturization to accommodate all of the new components, in addition to extensive prototyping and testing. As the number of *built in* instructions increases, the chip starts to become "top heavy." The complexity and costs leads to a case of "diminishing returns." Essentially, the time and effort needed to come up with these more complex architectures is met with limited increase in performance. Many engineers feel the need to do away with the burgeoning complexities, and to return to the simpler architectures that existed in the past (Frenzel, 1987).

RISC CPUs have fewer instruction sets when compared to their CISC counterparts (typically ranging from 30 - 100, instead of having hundreds like in CISC). Tests using larger computers determined that very complex instructions are utilized infrequently. In fact, about 80% of the available instructions were only used about 20% of the time. This means that 20% of the available instructions are used 80% of the time. It is clear that a *limited number* of instructions are the most important (Frenzel, 1987).

When instruction sets are small, it is possible to use a more simplistic form of wiring to control the CPU. Larger, more expensive, and slower micro-programming is eliminated.

Most RISC instructions operate more quickly than CISC instructions. Often a RISC instruction will take only 1 clock cycle to execute. This compares with CISC chips, which will commonly require several cycles for their instructions to execute. This means that RISC instructions are extremely efficient when compared to their CISC counterparts.

The underlying concept is that simpler is often better. It works more efficiently, and there is less to go wrong. RISC architectures are now being employed in a number of different computer platforms and peripheral devices. The increased performance that has been achieved through the use of these chips has been significant. Programming these machines is sometimes more difficult because complex instructions must be written separately, but overall, the results have been very positive. Look to see more RISC implementation in the future.

F) Parallel Processing. The "bottleneck" of having a single processor is the ultimate limiting factor affecting the performance of a computer. To overcome this problem, engineers have suggested the idea of splitting the workload among several processors instead. This idea has become known as *parallel processing*. In a parallel processing architecture, there are essentially several CPUs that each handle a portion of the work. The increase in speed is tied to the number of processors added to the system. The real advantage is that there is no theoretical limit to the number of processors that can be included.

Parallel processing increases the speed of every program that is run on the system. This is different from co-processing in that co-processors only handle specialized tasks. True parallel processing computers split the entire workload among the various processors. Therefore, no matter what application you run, you should see a significant increase in performance.

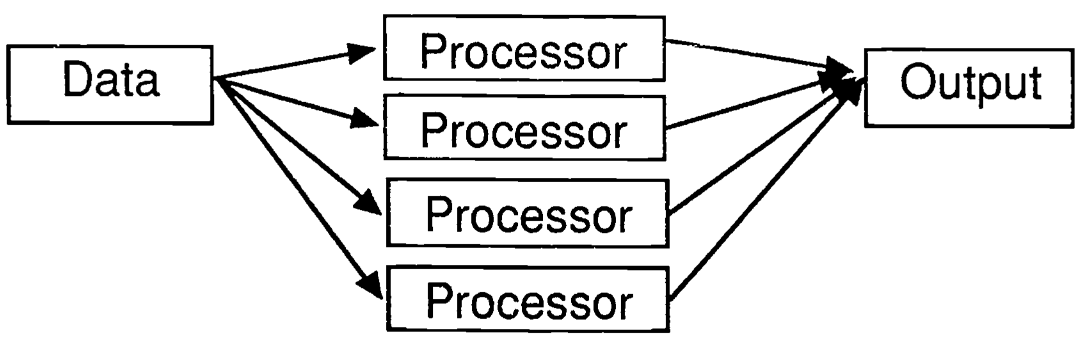

Figure 4. A parallel processing architecture

There are several mainframe "supercomputers" which now make use of parallel processing. One such computer makes use of over 200 processors. The performance of these machines is superior in almost every respect. The cost of manufacturing these machines, however, is quite high because this is a relatively new technology (Wood, 1987). Many scientists and engineers believe that parallel processing will be the wave of the future. You might expect to see parallel processing PCs within the next 10 years.

Why the need for speed?

One of the most dominant trends in microcomputers in the last 10 years has been the move towards greater *user friendliness*. Apple computer pioneered this effort with the introduction of the MacIntosh series of computer in 1984. Computers now have what are known as ICON based operating systems, which essentially means that you have pictures on the screen representing the various functions that can be handled by the computer. This compares with *command based* systems, which were used almost exclusively before the MacIntosh. To illustrate how a command based system is more difficult, let us compare the difference between copying a disk on the MacIntosh as opposed to copying a disk using an IBM PC with MS-DOS. Figure 5 provides us with an illustration of a typical MacIntosh screen with all of its ICONs and pull down menus.

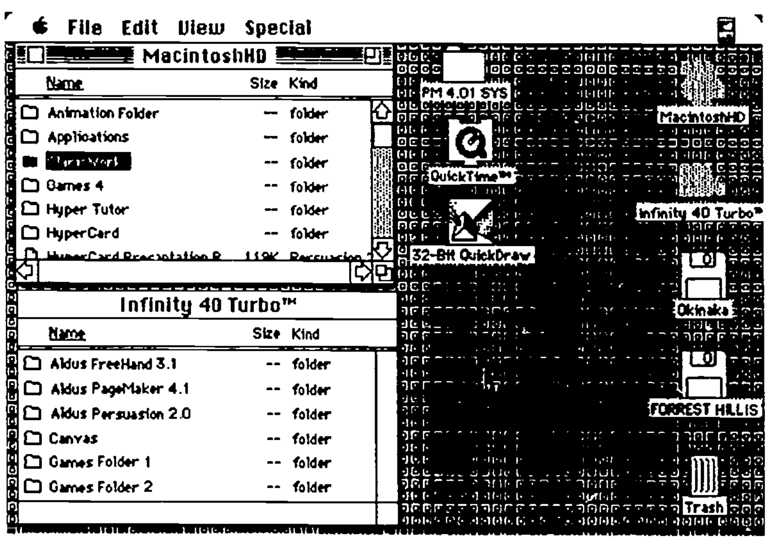

Figure 5. A sample MacIntosh Screen.

Floppy disks in this system are represented by pictures that actually look like floppy disks.

In this case, there are two disks: *Okinaka*, and *Forrest Hillis*. To copy the contents of *Okinaka* to *Forrest Hillis*, the user would "click" on the disk "Okinaka," and drag it to the disk "Forrest Hillis." The copying routine would be initiated, and be completed in about half a minute.

The same operation on an IBM PC would require the following *commands*.

COPY A: B:*.*

where A is the first drive, B is the second drive, and *.* indicates that you wish all of the contents of drive A to be copied to drive B. The omission of the colons, or the failure to get any of the characters in the correct sequence would result in the computer not responding properly. Because this type of system is more difficult to learn, and intimidating, ICON based systems tend to be preferred by a much wider variety of users.

While such systems are easier for the end user, the difficulties involved in programming such a system are much greater. Such systems also require greater amounts of processing power because the graphics and sophisticated commands that are programmed involve larger amounts of data. For anyone who has ever modified an IBM PC with Microsoft's Windows program (Windows is a program that tries to get the IBM PC to run more like a MacIntosh--with the ICON based commands), it is very clear that a better processor is needed than those that came with the original PCs. Windows, with all of its graphics, performs very sluggishly under the older 8086 and even 80286 microprocessors. It becomes more tolerable with a 32 bit 80386 machine.

Recently, there have been a much greater focus in areas like desktop publishing and multi-media. Desktop publishing is where the personal computer is converted into a print shop. It is possible to create newsletters, flyers, advertisements, school newspapers, etc. using the PC and appropriate software. This means that you can include graphics (illustrations, photographs, etc.) within the document, so that it is possible to create a very professional looking product. Anyone involved in doing this on a regular basis would find the need for a better processor, because again, large amounts of data must be handled when conducting desktop publishing.

Multi-media is an up and coming topic in the field of education. There are many programming languages like HyperCard for the MacIntosh, which allow the average person to create instructional presentations and programs of a very high caliber. It is possible to include graphics, sound, and animation within such programs, which can then be used to convey a multitude of concepts to students. Additionally, there is now a standardized on-screen video format for the MacIntosh known as QuickTime that allows actual video footage to be included within such programs. A person who writes a program on a topic like biology could actually include video segments which could be used to illustrate abstract ideas and principles such as "mitosis." The implications for education are staggering. Multi-media has the potential to revolutionize the way in which teachers teach in the classroom.

The problem, however, is that again, illustrations, and video segments represent tremendous volumes of data. A 6 minute QuickTime video can take up more than 50 Megabytes of storage space. To record and process such large volumes of data requires that the computer's microprocessor be able to work quickly and efficiently (Ito, 1992). Therefore, better processors are required for individuals wishing to conduct such work. Because of the educational possibilities that technology has brought forth, there are many teachers who are now pursuing course work in instructional technology. These same individuals need to make choices about which computer to buy. For these people, the computers with the better processors will be a necessity.

Supercomputers are now charged with the task of handling the computations necessary for predicting earthquakes and long term weather patterns. Parallel processing systems are necessary to handle these jobs, which have large numbers of variables, and are incredibly complicated. Weather patterns, in the long term, for example, require the programmer to account for interactions between the various layers of the atmosphere, the effect of land masses on air flow, variables like atmospheric pressure, incoming solar radiation (which is affected by the distance of the earth from the sun, and the angle of the earth relative to the incoming solar radiation), humidity, interactions between air and water, etc. The computations used in this process are voluminous and beyond the capabilities of ordinary "single processor" architectures, which is why scientists have always had problems when they have tried to predict weather in the long term. Parallel processing technologies may help scientists to get a firmer grip on topics of this nature.

Appendix I

There are essentially two main manufacturers of microprocessors. These are Intel Corporation, whose processors are used on IBM PCs and PC clones, and Motorola, whose processors are used on the MacIntosh, Commodore AMIGA, and Atari ST, as well as a number of peripheral devices. The following tables will provide the user with information about the different models of processor.

Microprocessor	Company	Data Path	Clockspeed
8086/88	Intel	16 bit	4.77 MHz
80286	Intel	16 bit	8 - 16 MHz
80386SX [a]	Intel	32 bit	16 - 20 Mhz
80386 [b]	Intel	32 bit	16 - 40 Mhz
80486DX [c]	Intel	32 bit	25 - 50 Mhz
80486SX [d]	Intel	32 bit	20 - 25 Mhz
80486DX2 [e]	Intel	32 bit	50 - 66 Mhz
68000	Motorola	32 bit	8 Mhz
68020	Motorola	32 bit	16 Mhz
68030	Motorola	32 bit	16 - 40 Mhz
68040 [f]	Motorola	32 bit	25 - 33 Mhz

Math Coprocessors (Floating Point Units, or FPUs)

Processor	Company	Data Path	Used with
8087	Intel	16 bit	8086/88
80287	Intel	16 bit	80286
80387	Intel	32 bit	80386
68882	Motorola	32 bit	68000 series [g]

a. 32 bit processor, but it communicates with the motherboard at 16 bit "chunks."
b. True 32 processing and communication with the motherboard.
c. Next generation above the 80386. It includes a built in math co-processor, or FPU.
d. Like the 80486DX minus the built in math co-processor.
e. Processes at 50 - 60 Mhz, but communicates with the rest of the computer at half that half that speed. About 85% as fast as a 50 Mhz 486DX chip, but much less expensive.
f. Built in math co-processor.
g. The 68882 with a compatible speed should be used with the corresponding 68000 series microprocessor.

Appendix II. Definition of Terms

The following terms are mentioned in the body of the paper. This page will serve as a quick reference for the user.

Arithmetic and Logic Unit--A part of the CPU that handles the calculations for the computer.

ASCII--Acronym standing for the *American Standard Code for Information Interchange*. This is a standard code used to represent the characters typed in from the keyboard. It utilizes groups of eight binary digits to represent individual characters.

Bit--Contraction for *binary digit*. This is the basic unit of computer memory.

Byte--Unit of computer memory equaling eight bits. Because it takes 8 bits to represent a character, one byte is equivalent to one character.

Cache Memory--Exceptionally fast memory that can respond very quickly to demands from the CPU. Used for the purpose of storing routines that are accessed frequently by the CPU, thereby increasing the efficiency of the computer significantly.

CISC--Acronym standing for *Complex Instruction Set Chip*. Microprocessors have progressively become more complex, and have been equipped with larger instruction sets over the years. Almost all current microcomputers are equipped with CISC chips.

Clockspeed--Term used to describe the number of cycles a microprocessor is capable of executing per second. This term is also known as the *frequency of operation*.

Control Unit--Part of the CPU that controls the flow of information for the computer.

Co-processing--Referring to a computer system that makes use of specialized processors above and beyond the regular CPU. These processors are used for specialized purposes, such as handling math or graphics functions. They handle these functions more efficiently than the regular CPU, and help to relieve some of the workload placed on the CPU.

CPU--Acronym standing for *Central Processing Unit*. This is the main control center of the computer. CPUs in microcomputers are also known as *microprocessors*.

Kilobyte--Unit of computer memory equaling 1000 bytes.

Megabyte--Unit of computer memory equaling 1,000,000 bytes.

Parallel Processing--Term describing a computer system that has several microprocessors that each share a part of the workload. Parallel processing systems increase speed of all applications by dividing the workload among the processors included in the system.

RAM--Acronym standing for *Random Access Memory*. This is the main electronic memory of the computer.

RISC--Acronym standing for Reduced Instruction Set Chip. This is a type of microprocessor that makes use of a simplified instruction set to increase efficiency.

References

Bitter, G.G. (1984). Computers in today's world. New York: John Wiley & Sons.

Frenzel, L. (1987). Crash course in artificial intelligence and expert systems, Indiana: Howard W. Sams and Co.

Ito, R. (1992). *QuickTime in action.* MacUser, 8 (5), pp. 84 - 92.

Lockard, J., and Abrams, P. (1987). Microcomputers for educators. Boston: Little, Brown and Company.

Magid, L. (1992). *If technology is pushed, prices fall.* Los Angeles Times, p. d3.

McKeown, P. (1988). Living with computers. New York: Harcourt Brace Jovanovich.

Simonson, M., and Thompson, A. (1990). Educational computing foundations, Ohio: Merrill Publishing Co.

Wood, L. (1992). *Mips, bips, and superchips.* Compute, 14 (3), pp. 58 - 66.