1-Introduction.

Computing machines have been around for a long time, hundreds of years. The Chinese abacus, the calculators with gears and wheels and the first analog computers are all examples of computing machinery; in some cases quite complex, that predates the introduction of digital computing systems. The computing machines that we're interested in came about in the 1940s because World War II artillery needed a more accurate way to calculate the trajectories of the shells fired from battleships. **Today**, the primary reason that computers have become so pervasive is the advances made in integrated circuit manufacturing technology. The modern computer has become faster and more powerful but the basic architecture of a computing machine has essentially stayed the same for many years.

Most of us use computers for a variety of tasks, from serious scientific computations to entertainment. The computer system can be divided into **computer hardware** and **computer software**. ***Computer hardware*** is the electronic circuitry that performs the actual work. Hardware includes things with which you are already familiar such as the processor, memory, keyboard, CD burner, and so on.

Computer software can be divided into application software and system software. A user interacts with the system through an application program. For the user, the application is the computer! For example, if you are interested in browsing the Internet, you interact with the system through a Web browser such as the Internet Explorer. For you, the system appears as though it is executing the application program (i.e., Web browser), as shown in Figure 1.

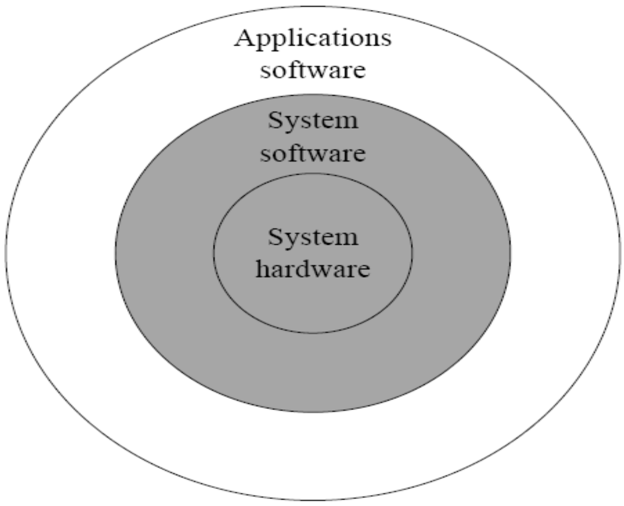

Figure 1.1 A user's view of a computer system.

The computer hardware, as represented by a desktop PC, can be thought of as being comprised of *four basic* parts:

1. Input devices can include components such as the mouse, keyboard, microphone, disks, modem and the network.

2. Output devices are components such as the display, disk, modem, sound card and speakers, and the network.

3. The memory system is comprised of internal and external caches, main memory, video memory and disk.

4. The central processing unit, or CPU, is comprised of the arithmetic and logic unit (ALU), control system and busses.

Computer architecture deals with the functional behavior of a computer system as viewed by a programmer. This view includes aspects such as the sizes of data types (*e.g.* using 16 binary digits to represent an integer), and the types of operations that are supported (like addition and subtraction). Also deals with the selection of the basic functional units

such as the processor and memory, and how they should be interconnected into a computer system.

Computer organization is concerned with how the various hardware components operate and how they are interconnected to implement the architectural specifications. **Computer organization** deals with structural relationships that are not visible to the programmer, such as interfaces to peripheral devices and the technology used for the memory.

Computers are complex systems. How do we manage complexity of these systems? We can get clues from looking at how we manage complex systems in life. Think of how a large corporation is managed. We use a hierarchical structure to simplify the management: president at the top and employees at the bottom. Each level of management filters out unnecessary details into the lower levels and presents only an abstracted version to the higher-level management. This is what we refer to as abstraction.

Different people view computer systems differently depending on the type of their interaction. We use the concept of abstraction to look at only the details that are necessary from a particular viewpoint. For example, if you are a computer architect, you are interested in the internal details that do not interest a normal user of the system. One can look at computer systems from several different perspectives. We have already talked about the user's view. We concentrate on the following views: (i) a programmer's view, (ii) an architect's view, and (iii) an implementer's view.

A ***programmer's view*** of a computer system depends on the type and level of language she intends to use. From the programmer's viewpoint, there exists a hierarchy from low-level languages to high-level languages.

A ***computer architect*** looks at the design aspect from a high level. She uses higher-level building blocks to optimize the overall system performance. A computer architect is much like an architect who designs buildings. For example, when designing a building, the building architect is not concerned with designing the elevator; as far as the architect is concerned, the elevator is a building block someone else designs. Similarly, a computer architect does not focus on low-level issues. From ***the architect's viewpoint***, a computer system consists of three main components: a processor or central processing unit (CPU), a memory unit, and input/output (I/O) devices.

Implementers are responsible for implementing the designs produced by computer architects. This group works at the digital logic level. At this level, logic gates and other hardware circuits are used to implement the various functional units.

2-The generations of computers.

This section traces the history of computers from their mechanical era. Our treatment is very brief.

1- **The first generation.**

Back to the beginning; the first generation of computing engines was comprised of the mechanical devices (called calculating machines). They were built using gears and powered by a hand-operated crank. The abacus, the adding machine, the punch card reader for textile machines fit into this category. Perhaps the most well-known mechanical system, called the *difference engine*, was built by Charles Babbage.

2- The second generation.

The next generation spanned the period from 1940–1960. Here electronic devices—vacuum tubes—were used as the active device or switching element. Even a vacuum tube is millions of times larger than the transistor on a silicon wafer. It consumes millions of times the power of the transistor, and its useful lifetime is hundreds or thousands of times less then a transistor. Although the vacuum tube computers were much faster than the mechanical computers of the preceding generation, they are thousands of times slower than the computers of today. Program instructions were given in machine language, which is a code composed entirely of 0s and 1s. These computers were slow, unreliable, expensive, and tedious to program.

3- The third generation.

The third generation covered roughly the period of time from 1960 to 1968. Here the transistor replaced the vacuum tube, and suddenly the computers began to be able to do real work. Companies such as IBM®, Burroughs® and Univac® built large mainframe computers. The IBM 360 family is a representative example of the mainframe computer of the day. Also at this time, Xerox® was carrying out some pioneering work on the human/computer interface at their Palo Alto Research Center, Xerox PARC. Here they studied what later would become computer networks, Windows® operating system. Programmers stopped programming in machine language and assembly language and began to use FORTRAN, COBOL and BASIC.

4- The fourth generation.

The fourth generation, roughly 1969–1977 was the age of the minicomputer. The minicomputer was the computer of the masses. It wasn't quite the PC, but it moved the computer out of the sterile environment of the "computer room," protected by technicians in white coats, to a computer in your lab. The minicomputer also represented the replacement of individual electronic parts, such as transistors and resistors, mounted on printed circuit boards (called discrete devices), with integrated circuits (IC), or collections of logic functions in a single package. It is small, faster, and more reliable than separate transistors. Here was the introduction of the small and medium scale integrated circuits. Companies such as Digital Equipment Company (DEC), Data General and Hewlett-Packard all built this generation of minicomputer. Also within this timeframe, simple integrated-circuit *microprocessors* were introduced and commercially produced by companies like Intel, Texas Instruments, Motorola, MOS Technology and Zilog. Early microcomputer devices that best represent this generation are the 4004, 8008 and 8080 from Intel, the 9900 from Texas Instruments and the 6800 from Motorola. The computer languages of the fourth generation were: assembly, C, Pascal, Modula, Smalltalk and Microsoft BASIC.

5- The fifth generation.

We are currently in the fifth generation, although it could be argued that the fifth generation ended with the Intel® 80486 microprocessor and the introduction of the Pentium® represents the sixth generation. We'll ignore that distinction until it is more

widely accepted. The advances made in semiconductor manufacturing technology best characterize the fifth generation of computers.

Today's semiconductor processes typify what is referred to as Very Large Scale Integration, or VLSI technology. Ever since ICs were made possible, the density has been growing at a phenomenal rate. By the mid-1970s, more than 10,000 components could be fabricated on a single chip. The next step, Ultra Large Scale Integration, or ULSI is either here today or right around the corner.

The fifth generation also saw the growth of the personal computer and the operating system as the primary focus of the machine. Standard hardware platforms controlled by standard operating systems enabled thousands of developers to create programs for these systems. In terms of software, the dominant languages became ADA, C++, JAVA, HTML and XML. In addition, graphical design language, based upon the universal modeling language (UML), began to appear.