Module 1 Lesson: one

**Introduction to Computer System**

The evolution of computers has been characterized by increasing processor speed, decreasing component size, increasing memory size, and increasing I/O capacity and speed. One factor responsible for the great increase in processor speed is the shrinking size of microprocessor components; this reduces the distance between components and hence increases speed. However, the true gains in speed in recent years have come from the organization of the processor, including heavy use of pipelining and parallel execution techniques and the use of speculative execution techniques (tentative execution of future instructions that might be needed). All of these techniques are designed to keep the processor busy as much of the time as possible. A critical issue in computer system design is balancing the performance of the various elements so that gains in performance in one area are not handicapped by a lag in other areas. In particular, processor speed has increased more rapidly than memory access Time. A variety of techniques is used to compensate for this mismatch, including caches, wider data paths from memory to processor, and more intelligent memory chips.

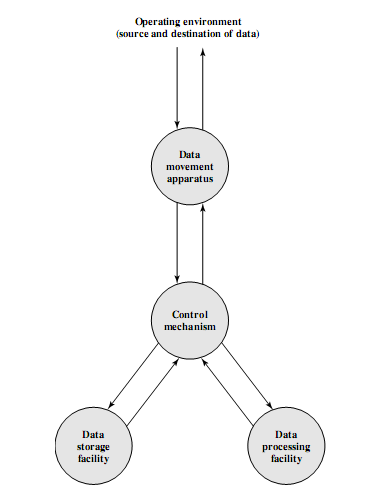
* 1. **Chapter Objectives**
* Overview of computer system
* Review historical development of computer systems
* Identify design levels for computer system development
* Discuss descriptive and design tools for each design level
* Compare and contrast various performance metrics for computer systems

**1.2 Computer system**

The hierarchical nature of complex systems is essential to both their design and their description. The designer need only deal with a particular level of the system at a time. At each level, the system consists of a set of components and their interrelationships. The behavior at each level depends only on a simplified, abstracted characterization of the system at the next lower level. At each level, the designer is concerned with structure and function:

• **Structure**: The way in which the components are interrelated

• **Function**: The operation of each individual component as part of the structure



**Figure 1.1 A Functional View of the Computer**

**Function**

Both the structure and functioning of a computer are, in essence, simple. In general terms, there are only four:

• Data processing

• Data storage

• Data movement

• Control

The computer, of course, must be able to ***process data***. The data may take a wide variety of forms, and the range of processing requirements is broad. However, we shall see that there are only a few fundamental methods or types of data processing. It is also essential that a computer ***store data***. Even if the computer is processing data on the fly (i.e., data come in and get processed, and the results go out immediately), the computer must ***temporarily store*** at least those pieces of data that are being worked on at any given moment. Thus, there is at least a short-term data storage function. Equally important, the computer performs a long-term data storage function.

The computer must be able to ***move data*** between itself and the outside world. The computer’s operating environment consists of devices that serve as either sources or destinations of data. When data are received from or delivered to a device that is directly connected to the computer, the process is known as input–output (I/O), and the device is referred to as a peripheral. When data are moved over longer distances, to or from a remote device, the process is known as data communications.

Finally, there must be ***control*** of these three functions. Ultimately, this control is exercised by the individual(s) who provides the computer with instructions. Within the computer, a control unit manages the computer’s resources and orchestrates the performance of its functional parts in response to those instructions.

**1.2.1 Computer structure**

There are four main structural components:

**Central processing unit (CPU**): Controls the operation of the computer and performs its data processing functions; often simply referred to as processor.

**Main memory**: Stores data.

**I/O**: Moves data between the computer and its external environment.

**System interconnection**: Some mechanism that provides for communication among CPU, main memory, and I/O. A common example of system interconnection is by means of a system bus, consisting of a number of conducting wires to which all the other components attach.

There may be one or more of each of the aforementioned components. Traditionally, there has been just a single processor. In recent years, there has been increasing use of multiple processors in a single computer.

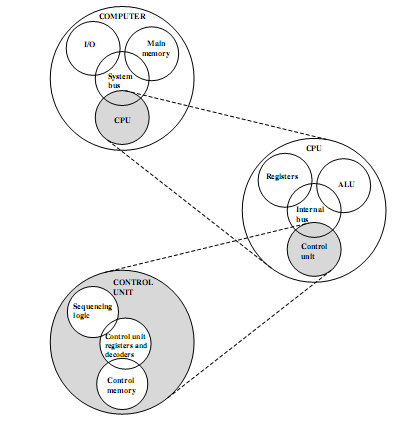
The most interesting and in some ways the most complex component is the CPU. Its major structural components are as follows:

**• Control unit**: Controls the operation of the CPU and hence the computer

• **Arithmetic and logic unit (ALU):** Performs the computer’s data processing functions

• **Registers**: Provides storage internal to the CPU

• **CPU interconnection**: Some mechanism that provides for communication among the control unit ,ALU, and registers



**Figure 1.2 The Computer: Top-Level Structure**

**1.3 A Brief History of computers**

**1.3.1 The First Generation**

***Vacuum Tubes***

The project was a response to U.S. needs during World War II. The Army’s Ballistics Research Laboratory (BRL)

***The Von Neumann Machine***

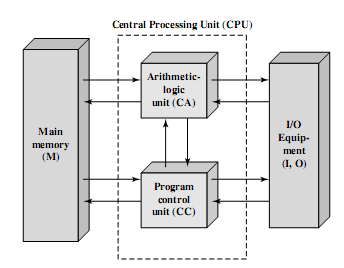
In 1946, von Neumann and his colleagues began the design of a new stored program computer, referred to as the IAS computer, at the Princeton Institute for Advanced Studies. The IAS computer, although not completed until 1952, is the prototype of all subsequent general-purpose computers. Figure 1.3 shows the general structure of the IAS computer. It consists of

• A main memory, which stores both data and instructions

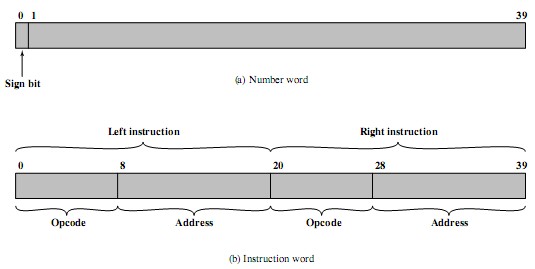
• An arithmetic and logic unit (ALU) capable of operating on binary data

• A control unit, which interprets the instructions in memory and causes them to be executed

• Input and output (I/O) equipment operated by the control unit



**Figure 1.3 Structure of the IAS Computer**



**Figure 1.4 IAS Memory Formats**

The memory of the IAS consists of 1000 storage locations, called words, of 40 binary digits (bits) each. Both data and instructions are stored there. Numbers are represented in binary form, and each instruction is a binary code. Figure 1.4 illustrates these formats. Each number is represented by a sign bit and a 39-bit value. A word may also contain two 20-bit instructions, with each instruction consisting of an 8-bit operation code (opcode) specifying the operation to be performed and a 12-bit address designating one of the words in memory (numbered from 0 to 999).

The control unit operates the IAS by fetching instructions from memory and executing them one at a time. To explain this, a more detailed structure diagram is needed, as indicated in Figure 1.5.This figure reveals that both the control unit and the ALU contain storage locations, called registers, defined as follows:

• **Memory buffer register (MBR)**: Contains a word to be stored in memory or sent to the I/O unit, or is used to receive a word from memory or from the I/O unit.

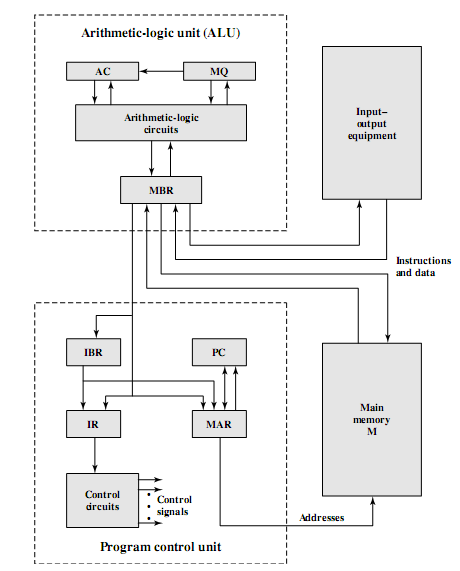
• **Memory address register (MAR):** Specifies the address in memory of the word to be written from or read into the MBR.

• **Instruction register (IR):** Contains the 8-bit opcode instruction being executed.

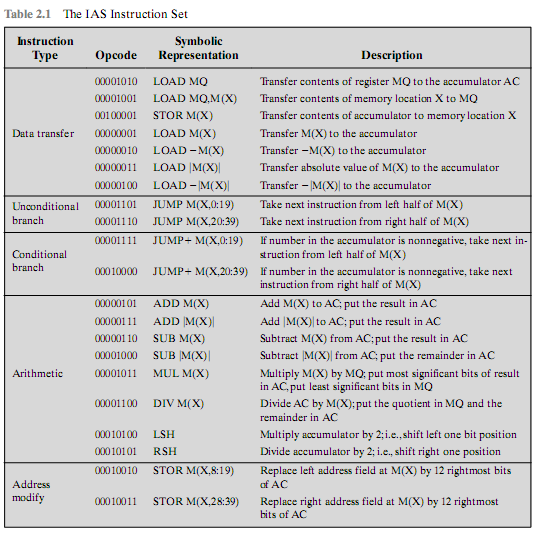
• **Instruction buffer register (IBR):** Employed to hold temporarily the right hand instruction from a word in memory.

• **Program counter (PC):** Contains the address of the next instruction-pair to be fetched from memory.

• **Accumulator (AC) and multiplier quotient (MQ)**: Employed to hold temporarily operands and results of ALU operations. For example, the result of multiplying two 40-bit numbers is an 80-bit number; the most significant 40 bits are stored in the AC and the least significant in the MQ.



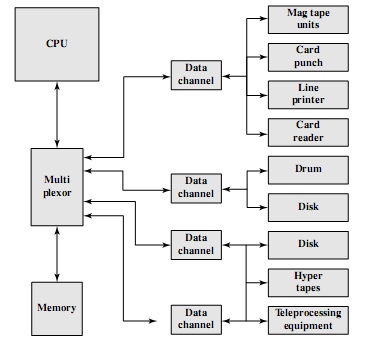
**Figure 1.5 Expanded Structure of IAS Computer**



**Table 1.1 The IAS Instruction set**

**1.3.2 The Second Generation: Transistors**

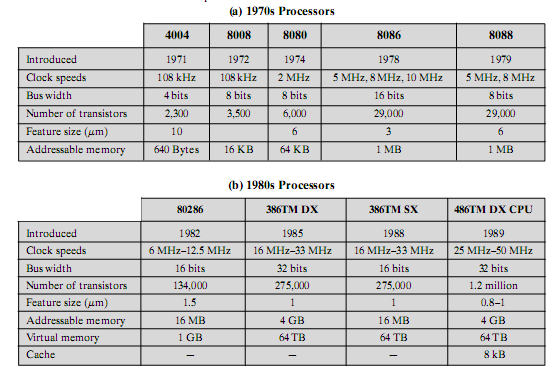
Figure 1.6 shows a large (many peripherals) configuration for an IBM 7094, which is representative of second-generation computers [BELL71]. Several differences from the IAS computer are worth noting. The most important of these is the use of data channels. A data channel is an independent I/O module with its own processor and its own instruction set. In a computer system with such devices, the CPU does not execute detailed I/O instructions. Such instructions are stored in a main memory to be executed by a special-purpose processor in the data channel itself. The CPU initiates an I/O transfer by sending a control signal to the data channel, instructing it to execute a sequence of instructions in memory. The data channel performs its task independently of the CPU and signals the CPU when the operation is complete. This arrangement relieves the CPU of a considerable processing burden. Another new feature is the multiplexor, which is the central termination point for data channels, the CPU, and memory. The multiplexor schedules access to the memory from the CPU and data channels, allowing these devices to act independently.

****

**Figure 1.6 An IBM 7094 Configuration**

**1.3.3 The Third Generation: Integrated Circuits**

**Table 1.2 Evolution of Intel Microprocessors**



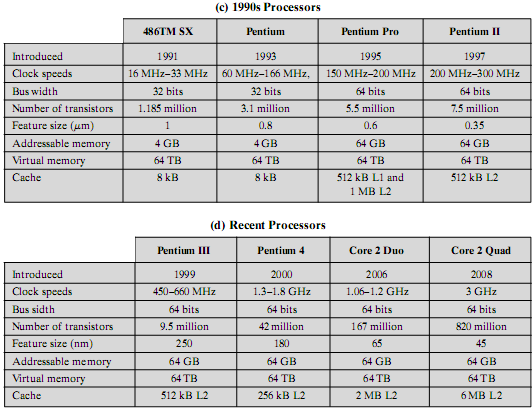
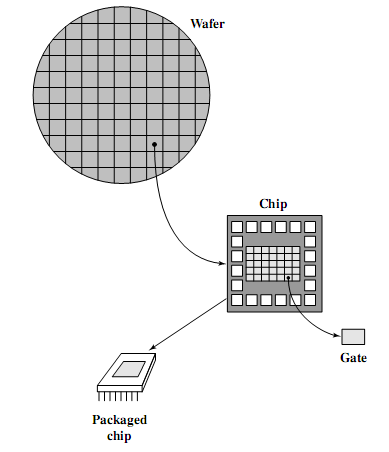


Figure 1.7 depicts the key concepts in an integrated circuit. A thin wafer of silicon is divided into a matrix of small areas, each a few millimeters square. The identical circuit pattern is fabricated in each area, and the wafer is broken up into chips. Each chip consists of many gates and/or memory cells plus a number of input and output attachment points. This chip is then packaged in housing that protects it and provides pins for attachment to devices beyond the chip. A number of these packages can then be interconnected on a printed circuit board to produce larger and more complex circuits.



**Figure 1.7 Relationship among Wafer, Chip, and Gate**

The concept of a family of compatible computers was both novel and extremely successful. A customer with modest requirements and a budget to match could start with the relatively inexpensive Model 30. Later, if the customer’s needs grew, it was possible to upgrade to a faster machine with more memory without sacrificing the investment in already-developed software. The characteristics of a family are as follows:

• ***Similar or identical instruction set***: In many cases, the exact same set of machine instructions is supported on all members of the family. Thus, a program that executes on one machine will also execute on any other. In some cases, the lower end of the family has an instruction set that is a subset of that of the top end of the family. This means that programs can move up but not down.

• ***Similar or identical operating system***: The same basic operating system is available for all family members. In some cases, additional features are added to the higher-end members.

• ***Increasing speed:*** The rate of instruction execution increases in going from lower to higher family members.

• ***Increasing number of I/O ports***: The number of I/O ports increases in going from lower to higher family members.

• ***Increasing memory size:*** The size of main memory increases in going from lower to higher family members.

• ***Increasing cost***: At a given point in time, the cost of a system increases in going from lower to higher family members.

**Assignment of Lesson 1:**

**Assignment 1**  
1.1What, in general terms, is the distinction between computer  
organization and computer architecture?  
1.2What, in general terms, is the distinction between computer structure  
and computer function?  
1.3 What are the four main functions of a computer?  
1.4 List and briefly define the main structural components of a computer.  
1.5 List and briefly define the main structural components of a processor.  
1.6 Construct a 5-to-32-line decoder with four 3-to-8-line decoders with  
enable and one 2-to-4-line decoder. Use block diagrams.  
1.7 Construct a 16-to-l-line multiplexer with two 8-to-l-line multiplexers  
and one 2-to-l-line multiplexer. Use block diagrams for the three  
multiplexers.  
1.8 Given a 32 x 8 ROM chip with an enable input, show the external  
connections necessary to construct a 128 x 8 ROM with four chips and a  
decoder.  
1.9 The following memory units are specified by the number of words  
times the number of bits per word. How many address lines and inputoutput data lines are needed in each case? (a) 2K x 16; (b) 64K x 8; (c)16M x 32; (d) 4G x 64.