

## ❖ **LEARNING OBJECTIVES**

After completion of this lecture, you should be able to:

- Describe the main computer parts.
- Recognize the different computer parts.

## 4. **MAIN COMPUTER PARTS**

A computer is a complex machine that is capable of performing huge computations at an extraordinary speed. Its processing power is often compared to that of a human brain. Although, human intellect is the undoubted winner in this competition, the capabilities of a computer cannot be underestimated. This complex machine, influenced from the design of a human brain. The basic parts of a desktop computer are the computer case, monitor, keyboard, mouse, and power cord. Each part plays an important role whenever you use a computer. So, let's take a look at the important parts of a computer.



FIGURE 7.MAIN COMPUTER PARTS

# PDF Eraser Free

## 4.1. COMPUTER CASE (SYSTEM UNIT)

The computer case is the metal and plastic box that contains the main components of the computer. It houses the *motherboard, central processing unit (CPU), the power supply,* and more.

Computer cases come in different shapes and sizes. A desktop case lies flat on a desk, and the monitor usually sits on top of it. A tower case is tall and sits next to the monitor or on the floor. The front of the case usually has an on/off switch and one or more optical drives.

Most of the personal computers you can purchase today include tower cases, rather than desktop cases; however, some computers are being made with all of the internal components built into the monitor, which completely eliminates the tower.



FIGURE 8. AN ALL-IN-ONE IMAC



FIGURE 9. TOWER COMPUTER CASE

### 4.1.1. CENTRAL PROCESSING UNIT (CPU)

Also known as the computer processor, it is the 'brain' of the computer. The CPU is an electronic circuit that executes computer programs. The primary responsibility of a computer processor is to execute a sequential set of instructions that constitute a program. CPU operation can be divided into four basic steps, namely, *fetch, decode, execute and writeback*. During the 'fetch' step, the processor retrieves program instructions from memory. In the decode step, the instruction is broken down into parts.

## PDF Eraser Free

The instruction set architecture of the CPU defines the way in which an instruction is decoded. In the 'execute' step, CPU performs the operation implied by the program instruction. During the 'writeback' step, the CPU writes back the results of execution, to the computer's memory.

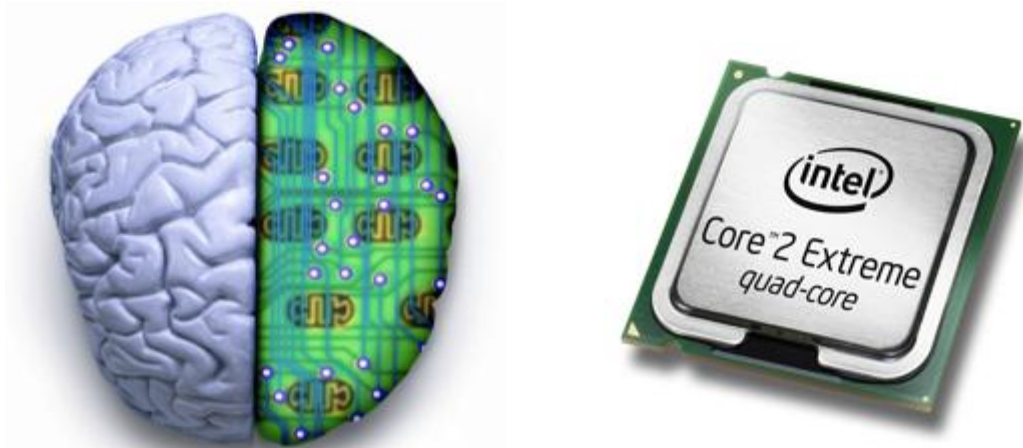


FIGURE 10. CENTRAL PROCESSING UNIT (CPU)

### 4.1.2. MOTHERBOARD

A computer motherboard consists of sockets in which microprocessors are installed, memory slots, a chipset that acts as an interface between the CPU bus and the peripheral buses, non-volatile memory chips housing the system's firmware and a clock generator, which helps in the synchronization of various system components. Some motherboards also include logic and connectors to support input devices like PS/2 connectors for a mouse and keyboard.



FIGURE 11. MOTHERBOARD

# PDF Eraser Free

## 4.1.3. **HARD DISK**

A hard disk is described as a part of the computer disk drive, which stores data and provides computer users with quick access to large amounts of data. A hard disk is an electromagnetically charged surface or set of disks that record data in concentric circles known as tracks. It is a non-volatile storage device that stores digitally encoded data. A head, resembling a phonograph arm, is used to read and write data onto a hard disk. The hard disks of desktops are generally capable of storing 120 GB to 2 TB of data. Laptop hard disk drives are smaller and have lower data storage capacities.



FIGURE 12.HARD DISK

## 4.1.4. **MEMORY**

Computer Memory It refers to those components of a computer, which retain digital data. It forms the core of a computer and makes up the basic computer model in collaboration with the CPU. Magnetic drums and delay lines used as primary storage by computers of the early days, have metamorphosed into a miniature silicon chip, which can achieve efficient storage of large volumes of data. Random Access Memory, popularly known as RAM, is a small-sized light and volatile form of computer memory. It is capable of temporary storage of data. Registers located in a computer processor are the fastest forms of computer storage. The most frequently used information is duplicated in the processor cache



FIGURE 13.COMPUTER MEMORY

## PDF Eraser Free

of a computer, thereby improving its performance. Computers require a non-volatile primary storage to read large programs. This non-volatile memory is known as ROM or Read-only memory. It also contains the startup programs used for bootstrapping a computer. Secondary storage media such as flash memory, magnetic tape, punch cards and zip drives and tertiary storage media like tape libraries are also a part of computer memory.

### 4.2. MONITER

The monitor works with a video card, located inside the computer case, to display images and text on the screen. Newer monitors usually have LCD (liquid crystal display) or LED (light-emitting diode) displays. These can be made very thin, and they are often called flat panel displays. Older monitors use CRT (cathode ray tube) displays. CRT monitors are much bigger and heavier, and they take up more desk space.



FIGURE 14.COMPUTER MONITOR

Most monitors have control buttons that allow you to change your monitor's display settings, and some monitors also have built-in speakers.

LED displays are actually LCD displays that are backlit with light-emitting diodes. This allows for greater contrast than a traditional LCD display.

### 4.3. POWER CORD

The power cord is the link between the power outlet and the power supply unit in the computer casing. If the power cord is not plugged in, the computer will not power on. To protect your computer from voltage spikes, you can plug the power cord into a



FIGURE 15.THE POWER CORD



## PDF Eraser Free

surge protector. You can also use an Uninterruptable Power Supply (UPS), which acts as a surge protector and also provides temporary power if there is a blackout.

### 4.4. **KEYBOARD**

The keyboard is one of the primary ways we communicate with the computer and enter data. There are many different types of computer keyboards such as wired, wireless, ergonomic, multimedia, and more. Although there may be differences in the location of some keys or features, keyboards are very similar and allow you to accomplish basically the same tasks.



FIGURE 16. THE KEYBOARD

### 4.5. **MOUSE**

The mouse is a peripheral that is known as a pointing device. It lets you point to objects on the screen, click on them, and move them.



FIGURE 17. THE MOUSE

There are two main types of mice: optical and mechanical.

*The optical mouse* uses an electronic eye to detect movement and is easier to clean.

*The mechanical mouse* uses a rolling ball to detect movement. Generally, a mechanical mouse is cheaper, although it may require regular cleaning to keep it working properly.

Traditionally, a mouse connects to the computer using a USB or PS/2 connection. However, you can also buy a wireless mouse, which can reduce clutter on your desktop.

### 4.6. **MOUSE ALTERNATIVE**

There are other devices that can do the same thing a mouse can do, but with a different look and feel. Many



## PDF Eraser Free

people find them to be easier to use, and they also require less desk space than a mouse. The most common mouse alternatives include: A touchpad on a laptop.

- **Trackball:** A trackball has a ball on top that can rotate freely. Instead of moving the device like a mouse, you can simply roll the ball with your fingers to move the pointer. Some mobile devices have miniature trackballs that can be controlled with your thumb.
- **Touchpad:** A touchpad (also called a trackpad) is a touch-sensitive pad that lets you control the pointer by making a "drawing" motion with your finger. Touchpads are very common on laptop computers.

### 4.7. *SPEAKERS*

The speakers are the audio component of the computer.



FIGURE 19.SPEAKERS

### 4.8. *PRINTERS*

The printer is used to develop a hard copy or paper copy of a document or picture. The printer connects to the computer and requires the use of its own power cord.



FIGURE 20.PRINTERS

### 4.9. *COMPACT DISK (CD)*

This device is used for storage. Some CDs hold information to be loaded into a computer. Some are used to hold information from the computer. It depends on what type of CD it is.



FIGURE 21.COMPACT DISK

# PDF Eraser Free

## 4.10. THUMB DRIVE

The thumb drive is a portable storage device. You use it store information from a computer and use it on another system. They are also known as flash drives.



FIGURE 22. THUMB DRIVE (FLASH MEMORY)

### Note

Not all hardware components look the same. In fact, a laptop has most of the same hardware components but is more compact and not as easy to recognize. The laptop is a portable computing device. Your task is to realize these parts exist, how they work for you, and how you can use them.

### Challenge!

- Think about the **desktop computers** you've seen at work, school, the library, a store, or a friend's house. What did they look like? Were they **all-in-one**, or did they have a separate **tower**?
- Review the **Parts of the Keyboard** interactive on this le. Are there any keys that you haven't used before?
- If you're using a **mouse**, flip it over to see whether it's **optical** or **mechanical**.
- Is your monitor **LCD**, **LED**, or **CRT**?
- If your monitor has **control buttons**, try adjusting the **brightness** and **contrast**.



## ❖ LEARNING OBJECTIVES

After completion of this lecture, you should be able to:

- Describe the data representation
- Describe the number systems

## 5. DATA REPRESENTATION

The basic building block of personal computers is the transistor. A *transistor* is an electronic device for controlling the flow of electrons in an electrical circuit. If electrons are allowed to flow, the circuit is on; conversely, if electrons are not allowed to flow, the circuit is off. Thinking of a transistorized circuit as a switch like a light switch at a home. The switch is either **on** or **off** and stays that way until it is flipped again. When a circuit is **on**, we say it is in the marking state and assign a **1** to it. Conversely, when it is **off**, we assign a **0** to it.

A modern digital computer is often said to be a binary computer because its most basic circuits can remember either one of *two states: 0 and 1*. The binary digits 0 and 1 are called **bits**.

## 6. NUMBERS SYSTEMS

The number systems that we discuss here are based on positional number systems. The decimal number system that we are already familiar with is an example of a positional number system. In contrast, the Roman numeral system is not a positional number system. Every positional number system has a *base*, and an *alphabet*. The base is a positive number. For example, the decimal system is a base-10 system. The number of symbols in the alphabet is equal to the base of the number system. The alphabet of the decimal system is 0 through 9, a total of 10 symbols or digits. There are four number systems that are relevant in the context of computer systems and programming. These are the *decimal* (base-10), *binary* (base-2), *octal* (base-8), and *hexadecimal* (base-16) number systems.

## PDF Eraser Free

System	Base	Possible digits (alphabet)
Binary	2	0 1
Octal	8	0 1 2 3 4 5 6 7
Decimal	10	0 1 2 3 4 5 6 7 8 9
Hexadecimal	16	0 1 2 3 4 5 6 7 8 9 A B C D E F

Digits	Binary
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

### Note

Computers internally use the binary system. The remaining two number systems—octal and hexadecimal—are used mainly for convenience to write a binary number even though they are number systems on their own. We would have ended up using these number systems if we had 8 or 16 fingers instead of 10.

In a positional number system, a sequence of digits is used to represent a number. Each digit in this sequence should be a symbol in the alphabet. There is a weight associated with each position. If we count position numbers from right to left starting with zero, the weight of position  $n$  in a base  $b$  number system is  $b^n$ . For example, the number 579 in the decimal system is actually interpreted as

$$5 \times (10^2) + 7 \times (10^1) + 9 \times (10^0) .$$

(Of course,  $10^0=1$ ). In other words, 9 is in unit's place, 7 in 10's place, and 5 in 100's place.

### 6.1. BINARY SYSTEM

The key to understanding computers is the binary number system. The binary number system has only two digits: **0 to 1**. Just as decimal notation is based on places that represent powers of ten,

## PDF Eraser Free

binary notation is based on power of two. For example, the decimal number **537** is really the sum of powers of ten:

$$\begin{aligned}537 &= (5 \times 10^2) + (3 \times 10^1) + (7 \times 10^0) \\ &= (5 \times 100) + (3 \times 10) + (7 \times 1) \\ &= 500 + 30 + 7 = 537\end{aligned}$$

Binary numbers are sums of powers of two in the same way that decimal numbers are sums of powers of ten. The following table shows the decimal numbers represented by some of more important powers of two in personal computing:

$2^{-2} = 0.25$	$2^{-1} = 0.5$	$2^0 = 1$	$2^1 = 2$	$2^2 = 4$
$2^3 = 8$	$2^4 = 16$	$2^5 = 32$	$2^6 = 64$	$2^7 = 128$
$2^8 = 256$	$2^{16} = 65.536$	$2^{20} = 1.048.576$	$2^{24} = 16.777.216$	

**A BINARY NUMBER** is a string of 1s and 0s, each indicating the presence or absence of a power of two. For example, consider the binary number **101**. This number is converted to its decimal equivalents as follows:

**Binary number**    0 0 0 0 0 1 0 1

**Power of two**    7 6 5 4 3 2 1 0

**Decimal number** 0 0 0 0 0 4 0 1

$$101 \text{ binary} = (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0)$$

$$= (1 \times 4) + (0 \times 2) + (1 \times 1)$$

$$= 4 + 0 + 1 = 5 \text{ decimal}$$

Thus, **101** is the binary number representation of the decimal number **5**. In a binary number such **101.1** the fractional number part is a sum of negative powers of **2**. For example, **101.1** binary is converted to its decimal equivalent as follows:

## PDF Eraser Free

$$101.1 \text{ binary} = (1 \times 2^2) + (0 \times 2^1) + (1 \times 2^0) + (1 \times 2^{-1})$$

$$= (1 \times 4) + (0 \times 2) + (1 \times 1) + (1 \times (1/2))$$

$$= 4 + 0 + 1 + 1/2 = 5.5 \text{ decimal}$$

Keep in mind that computers work **exclusively with binary numbers** because they can store only either a 1 or a 0. To do arithmetic and word processing they must convert from binary to decimal and back again.

In the binary system, using  $n$  bits, we can represent numbers from 0 through  $(2^n - 1)$  for a total of  $2^n$  different values. We need  $m$  bits to represent  $X$  different values, where

$$m = \lceil \log_2 X \rceil .$$

for example, 150 different values can be represented by using

$$\lceil \log_2 150 \rceil = \lceil 7.229 \rceil = 8 \text{ bits} .$$

In fact, using 8 bits, we can represent  $2^8 = 256$  different values (i.e., from 0 through 255).

### 6.2. BASIC TERMS AND NOTATION

The alphabet of computers, more precisely digital computers, consists of 0 and 1. Each is called a **bit**, which stands for the binary digit. The term **byte** is used to represent a group of 8 bits. The term **word** is used to refer to a group of bytes that is processed simultaneously. The exact number of bytes that constitute a word depends on the system. For example, in the Pentium, a word refers to four bytes or 32 bits. We use the abbreviation “**b**” for bits, “**B**” for bytes, and “**W**” for words. Sometimes we also use **doubleword** and **quadword**. A **doubleword** has twice the number of bits as the word and the **quadword** has four times the number of bits in a word.

**2-bit**  $\rightarrow 2^2 = 4$       possible states      **(00, 01, 10, 11)**

**3-bit**  $\rightarrow 2^3 = 8$       possible states      **(000, - - -, 111)**

**8-bit**  $\rightarrow 2^8 = 256$       possible states      **(00000000, - - -, 11111111)**

## PDF Eraser Free

Bits in a word are usually ordered from right to left, as you would write digits in a decimal number. The rightmost bit is called the *least significant bit (LSB)*, and the leftmost bit is called the *most significant bit (MSB)*.

We use standard terms such as *kilo* (K), *mega* (M), *giga* (G), and so on to represent large integers. Unfortunately, we use two different versions of each, depending on the number system, decimal or binary. Table 1 summarizes the differences between the two systems. Typically, computer-related attributes use the binary version. For example, when we say 128 megabyte (MB) memory, we mean  $128 \times 2^{20}$  bytes. Usually, communication-related quantities and time units are expressed using the decimal system. For example, when we say that the data transfer rate is 100 megabits/second (Mb/s), we mean  $100 \times 10^6$  Mb/s.

**Table 1** Terms to represent large integer values

Term	Decimal (base 10)	Binary (base 2)
K (kilo)	$10^3$	$2^{10}$
M (mega)	$10^6$	$2^{20}$
G (giga)	$10^9$	$2^{30}$
T (tera)	$10^{12}$	$2^{40}$
P (peta)	$10^{15}$	$2^{50}$

## 7. UNSIGNED INTEGER REPRESENTATION

Now that you are familiar with different number systems, let us turn our attention to how integers (numbers with no fractional part) are represented internally in computers. Of course, we know that the binary number system is used internally. Still, there are a number of other details that need to be sorted out before we have a workable internal number representation scheme.

## PDF Eraser Free

The most natural way to represent unsigned (i.e., nonnegative) numbers is to use the equivalent binary representation. A binary number with  $n$  bits can represent  $2^n$  different values, and the range of the numbers is from 0 to  $2^n - 1$ . Padding of 0s on the left can be used to make the binary conversion of a decimal number equal exactly  $N$  bits. For example, to represent 16D we need  $\lceil \log_2 16 \rceil = 5$  bits. Therefore,  $16D = 10000B$ . However, this can be extended to a byte (i.e.,  $N=8$ ) as

00010000B

A problem arises if the number of bits required to represent an integer in binary is more than the  $N$  bits we have. Clearly, such numbers are outside the range of numbers that can be represented using  $N$  bits. Recall that using  $N$  bits, we can represent any integer  $X$  such that

$$0 \leq X \leq 2^n - 1.$$

## 8. SIGNED NUMBER REPRESENTATION

One common way of handling negative numbers is to add one bit to the binary code of the number called sign bit. This is the frequently the most left bit, with a **0** indicating a positive number and a **1** a negative number.

There are three widely used techniques for representing both positive and negative numbers:-

1. **Sign and magnitude.**
2. **1's complement.**
3. **2's complement.**

*In sign and magnitude* method the first bit from the left used for a sign and the remaining for the magnitude of the number. For example, the numbers between **-127** and **+127** could be represented in **8 bits**, the *first* being used for a sign and the *remaining 7* for the magnitude.

For example, the negative number (-18) is represented using 6 bits, base 2 in the sign-magnitude format, as follows (110010), while a (+18) is represented as (010010). Although simple, the sign-magnitude representation is complicated when performing arithmetic operations. In particular, the sign bit has to be dealt with separately from the magnitude bits. Consider, for example, the addition of the two numbers +18 (010010) and -19 (110011) using the sign-magnitude representation. Since



## PDF Eraser Free

the two numbers carry different signs, then the result should carry the sign of the larger number in magnitude, in this case the (-19). The remaining 5-bit numbers are subtracted (10011-10010) to produce (00001), that is, (-1).



The positive values have the same representation systems, while variation occurs in the representation of negative values.

*In the 1's complement* representation system, negative values are obtained by complementing each bit of the representation of the corresponding positive value.

*According to the 2's complement* system, a positive number is represented the same way as in the sign-magnitude. However, a negative number is obtained by adding 1 to the 1's complement of value. Consider, for example, the representation of the number (-19) using 2's complement. In this case, the number 19 is first represented as (010011). Then each digit is complemented, hence the number will be (101100). Finally a "1" is added at the least significant bit position to result in (101101). Now, consider the 2's complement representation of the number (+18). Since the number is positive, then it is represented as (010010), the same as in the sign-magnitude case. Now, consider the addition of these two numbers. In this case, we add the corresponding bits without giving special treatment to the sign bit. The results of adding the two numbers produces (111111). This is the 2's complement representation of a (-1), as expected. The main advantage of the 2's complement representation is that no special treatment is needed for the sign of the numbers. Another characteristic of the 2's complement is the fact that a carry coming out of the most significant bit while performing arithmetic operations is ignored without affecting the correctness of the result. Consider, for example, adding -19 (101101) and +26 (011010). The result will be (1)(000111), which is correct (+7) if the carry bit is ignored.

**EX: Represent the number +9 using the three methods**

+9	→	0,1001	signed – magnitude
	→	0,1001	1's complement
	→	0,1001	2's complement

## PDF Eraser Free

**EX: what is the 1's complement, 2's complement and signed – magnitude of -9.**

-9	→	1,1001	signed – magnitude
	→	1,0110	1's complement
	→	1,0111	2's complement

**Note:-** Special case in 2's complement representation is be that whenever a signed number has a 1 in the sign bit and a 0 for all the magnitude bits, its decimal equivalent is (  $-2^{N-1}$  ) where  $N$  is the total number of bits including sign bit. For example:

$$1,0000 \longrightarrow 2^{5-1} = -2^4 = -16$$

The advantage of the sign 2's complement representation over the 1's complement and sign/magnitude is that **it contains only one type of zero** while the other representations have both a +0 and -0. And the other is that **addition of 2' complement numbers can be performed without regard for the sign.**

### ***STUDENT-ACTIVITY***

1. What is the basic building block of personal computers?
2. List various number systems with examples.
3. Define the binary number system.