BIOS (basic input/output system) is the program a personal **computer's** microprocessor uses to get the **computer** system started after you turn it on. It also manages data flow between the **computer's** operating system and attached devices such as the hard disk, video adapter, keyboard, mouse and printer.
What is the purpose of BIOS?
BIOS enables computers to perform certain operations as soon as they are turned on. The principal job of a computer's **BIOS** is to govern the early stages of the startup process, ensuring that the operating system is correctly loaded into memory.
The four main functions of a PC BIOS

- POST - Test the computer hardware and make sure no errors exist before loading the operating system. ...
- Bootstrap Loader - Locate the operating system. ...
- BIOS drivers - Low-level drivers that give the computer basic operational control over your computer's hardware.

The Motherboard is the **heart of the computer**. The **BIOS** is like your mother kicking you out of bed and explaining what you will wear, what you will do and what will you do it with.

here are two different types of BIOS:

- UEFI (Unified Extensible Firmware Interface) BIOS - Any modern PC has a UEFI BIOS. ...
- Legacy BIOS (Basic Input/Output System) - Older motherboards have legacy BIOS firmware for turning on the PC.

The **Booting Process**. **Booting** (also known as **booting** up) is the initial set of operations that a computer system performs when electrical power is switched on. The **process** begins when a computer that has been turned off is re-energized, and ends when the computer is ready to perform its normal operations.

Motherboard: Definition

A motherboard is one of the most essential parts of a computer system. It holds together many of the crucial components of a computer, including the central processing unit (CPU), memory and connectors for input and output devices. The base of a motherboard consists of a very firm sheet of non-conductive material, typically some sort of rigid plastic. Thin layers of copper or aluminum foil, referred to as *traces*, are printed onto this sheet. These traces are very narrow and form the circuits between the various components. In addition to circuits, a motherboard contains a number of sockets and slots to connect the other components.

Parts of a Motherboard

If you were to open up your computer and take out the motherboard, you would probably get pretty confused about all the different parts. Depending on the make and model of your computer, it might look something like this.

Photograph of a typical motherboard of a desktop computer

To understand how computers work, you don't need to know every single part of the motherboard. However, it is good to know some of the more important parts and how the motherboard connects the various parts of a computer system together. Here are some of the typical parts:

- A CPU socket - the actual CPU is directly soldered onto the socket. Since high speed CPUs generate a lot of heat, there are heat sinks and mounting points for fans right next to the CPU socket.
- A power connector to distribute power to the CPU and other components.
- Slots for the system's main memory, typically in the form of DRAM chips.
- A chip forms an interface between the CPU, the main memory and other components. On many types of motherboards, this is referred to as the Northbridge. This chip also contains a large heat sink.
- A second chip controls the input and output (I/O) functions. It is not connected directly to the CPU but to the Northbridge. This I/O controller is referred to as the Southbridge. The Northbridge and Southbridge combined are referred to as the *chipset*.
- Several connectors, which provide the physical interface between input and output devices and the motherboard. The Southbridge handles these connections.

- Slots for one or more hard drives to store files. The most common types of connections are Integrated Drive Electronics (IDE) and Serial Advanced Technology Attachment (SATA).
- A read-only memory (ROM) chip, which contains the firmware, or startup instructions for the computer system. This is also called the BIOS.
- A slot for a video or graphics card. There are a number of different types of slots, including the Accelerated Graphics Port (AGP) and Peripheral Component Interconnect Express (PCIe).
- Additional slots to connect hardware in the form of Peripheral Component Interconnect (PCI) slots.

Mode of Transfer:

The binary information that is received from an external device is usually stored in the memory unit. The information that is transferred from the CPU to the external device is originated from the memory unit. CPU merely processes the information but the source and target is always the memory unit. Data transfer between CPU and the I/O devices may be done in different modes.

Data transfer to and from the peripherals may be done in any of the three possible ways

1. rogrammed I/O.
2. Interrupt- initiated I/O.
3. Direct memory access(DMA).

Now let's discuss each mode one by one.

1. **Programmed I/O:** It is due to the result of the I/O instructions that are written in the computer program. Each data item transfer is initiated by an instruction in the program. Usually the transfer is from a CPU register and memory. In this case it requires constant monitoring by the CPU of the peripheral devices.
   Example of Programmed I/O: In this case, the I/O device does not have direct access to the memory unit. A transfer from I/O device to memory requires the execution of several instructions by the CPU, including an input instruction to transfer the data from device to the CPU and store instruction to transfer the data from CPU to memory. In programmed I/O, the CPU stays in the program loop until the I/O unit indicates that it is ready for data transfer. This is a time consuming process since it needlessly keeps the CPU busy. This situation can be avoided by using an interrupt facility. This is discussed below.

2. **Interrupt- initiated I/O:** Since in the above case we saw the CPU is kept busy unnecessarily. This situation can very well be avoided by using an interrupt driven
3. method for data transfer. By using interrupt facility and special commands to inform the interface to issue an interrupt request signal whenever data is available from any device. In the meantime the CPU can proceed for any other program execution. The interface meanwhile keeps monitoring the device. Whenever it is determined that the device is ready for data transfer it initiates an interrupt request signal to the computer. Upon detection of an external interrupt

signal the CPU stops momentarily the task that it was already performing, branches to the service program to process the I/O transfer, and then return to the task it was originally performing.

Note: Both the methods programmed I/O and Interrupt-driven I/O require the active intervention of the
processor to transfer data between memory and the I/O module, and any data transfer must transverse
a path through the processor. Thus both these forms of I/O suffer from two inherent drawbacks.

Definition - What does Direct Memory Access (DMA) mean?

Direct memory access (DMA) is a method that allows an input/output (I/O) device to send or receive data directly to or from the main memory, bypassing the CPU to speed up memory operations. The process is managed by a chip known as a DMA controller (DMAC).

A computer's system resource tools are used for communication between hardware and software. The four types of system resources are:

- I/O addresses
- Memory addresses
- Interrupt request numbers (IRQ)
- Direct memory access (DMA) channels

DMA channels are used to communicate data between the peripheral device and the system memory. All four system resources rely on certain lines on a bus. Some lines on the bus are used for IRQs, some for addresses (the I/O addresses and the memory address) and some for DMA channels.

A DMA channel enables a device to transfer data without exposing the CPU to a work overload. Without the DMA channels, the CPU copies every piece of data using a peripheral bus from the I/O device. Using a peripheral bus occupies the CPU during the read/write process and does not allow other work to be performed until the operation is complete