

Computer Applications and Fundamentals



Computer Applications and Fundamentals Study Notes: Definition of Computer:

A computer is an electronic device that manipulates information or data. It has the ability to store, retrieve, and process data. You may already know that you can use a computer to type documents, send an email, play games, and browse the Web. You can also use it to edit or create spreadsheets, presentations, and even videos.

Computer System

Computer System is divided into two functional entities: Hardware and Software

- Hardware is any part of your computer that has a physical structure, such as the keyboard or mouse. It also includes all of the computer's internal parts, which you can see in the image below.
- Software is any set of instructions that tells the hardware what to do and how to do it. Examples of software include web browsers, games, and word processors.

BBA Important Exams

SET BBA

IPMAT

GGSIU CET

JIPMAT

DU JAT

UGAT

NPAT

NCHMCT JEE

What are the different types of computers?

When most people hear the word computer, they think of a personal computer such as a desktop or laptop. However, computers come in many shapes and sizes, and they perform many different functions in our daily lives. When you withdraw cash from an ATM, scan groceries at the store, or use a calculator, you're using a type of computer.

- Desktop computers
- Laptop computers
- Tablet computers

Von Neumann Architecture

Von Neumann architecture was first published by John von Neumann in 1945. His computer architecture design consists of a Control Unit, Arithmetic and Logic Unit (ALU), Memory Unit, Registers and Inputs/Outputs. Von Neumann architecture is based on the stored-program computer concept, where instruction data and program data are stored in the same memory. This design is still used in most computers produced today.

Harvard Architecture

In a normal computer that follows von Neumann architecture, instructions and data both are stored in the same memory. So same buses are used to fetch instructions and data. This means the CPU cannot do both things together (read the instruction and read/write data). Harvard Architecture is the computer architecture that contains separate storage and separate buses (signal path) for instruction and data. It was basically developed to overcome the bottleneck of Von Neumann Architecture. The main advantage of having separate buses for instruction and data is that the CPU can access instructions and read/write data at the same time.



Prepare for BBA & IPM
Entrance Exams

START FREE TRIAL

Evolution of Digital Computer

1. First Generation - 1940-1956: Vacuum Tubes:

The first computers used vacuum tubes for circuitry and magnetic drums for memory, and were often enormous, taking up entire rooms. They were very expensive to operate and in addition to using a great deal of electricity, generated a lot of heat, which was often the cause of malfunctions. First-generation computers relied on machine language to perform operations, and they could only solve one problem at a time. The input was based on punched cards and paper tape, and the output was displayed on printouts.

2. Second Generation - 1956-1963: Transistors:

Transistors replaced vacuum tubes and ushered in the second generation of computers. The transistor was invented in 1947 but did not see widespread use in computers until the late 50s. The transistor was a vast improvement over the vacuum tube, allowing computers to become smaller, faster, cheaper, more energy-efficient and more reliable than their first-generation predecessors. Second-generation computers still relied on punched cards for input and printouts for output. Second-generation computers moved from cryptic binary machine language to symbolic, or assembly, languages, which allowed programmers to specify instructions in words. High-level programming languages were also being developed at this time, such as early versions of COBOL and FORTRAN. These were also the first computers that stored their instructions in their memory, which moved from a magnetic drum to magnetic core technology. The first computers of this generation were developed for the atomic energy industry.

3. Third Generation - 1964-1971: Integrated Circuits:

The development of the integrated circuit was the hallmark of the third generation of computers. Transistors were miniaturized and placed on silicon chips, called semiconductors, which drastically increased the speed and efficiency of computers. Instead of punched cards and printouts, users interacted with third generation computers through keyboards and monitors and interfaced with an operating system, which allowed the device to run many different applications at one time with a central program that monitored the memory. Computers for the first time became accessible to a mass audience because they were smaller and cheaper than their predecessors.

4. Fourth Generation - 1971-Present: Microprocessors:

The period 1972 to 2010 is roughly considered the fourth generation of computers. The fourth-generation computers were developed by using microprocessor technology. By coming to the fourth generation, computers became very small in size, it became portable. The machine of the fourth generation started generating a very low amount of heat. It is much faster and accuracy became more reliable. The production cost reduced to very low in comparison to the previous generation.

5. Fifth Generation - Present and Beyond: Artificial Intelligence:

Fifth-generation computing devices, based on artificial intelligence, are still in development, though there are some applications, such as voice recognition, that are being used today. The use of parallel processing and superconductors is helping to make artificial intelligence a reality. Quantum computation and molecular and nanotechnology will radically change the face of computers in years to come. The goal of fifth-generation computing is to develop devices that respond to natural language input and are capable of learning and self-organization.



CPU [Central Processing Unit]

The Central Processing Unit (CPU) is the electronic circuit responsible for executing the instructions of a computer program.

It is sometimes referred to as the microprocessor or processor

The CPU contains the ALU, CU and a variety of registers.

Registers:

Registers are high-speed storage areas in the CPU. All data must be stored in a register before it can be processed.

MAR Memory Address Register	Holds the memory location of data that needs to be accessed
MDR Memory Data Register	Holds data that is being transferred to or from memory
AC Accumulator	Where intermediate arithmetic and logic results are stored
PC Program Counter	Contains the address of the next instruction to be executed
CIR Current Instruction Register	Contains the current instruction during processing

Arithmetic and Logic Unit (ALU)

The ALU allows arithmetic (add, subtract etc) and logic (AND, OR, NOT etc) operations to be carried out.

Control Unit (CU)

The control unit controls the operation of the computer's ALU, memory and input/output devices, telling them how to respond to the program instructions it has just read and interpreted from the memory unit.

The control unit also provides the timing and control signals required by other computer components.

Buses

Buses are the means by which data is transmitted from one part of a computer to another, connecting all major internal components to the CPU and memory. A standard CPU system bus is comprised of a control bus, data bus and address bus.

Address Bus	Carries the addresses of data (but not the data) between the processor and memory
Data Bus	Carries data between the processor, the memory unit and the input/output devices
Control Bus	Carries control signals/commands from the CPU (and status signals from other devices) in order to control and coordinate all the activities within the computer



Memory Unit

The memory unit consists of RAM, sometimes referred to as primary or main memory. Unlike a hard drive (secondary memory), this memory is fast and also directly accessible by the CPU.

RAM is split into partitions. Each partition consists of an address and its contents (both in binary form).

The address will uniquely identify every location in the memory.

Loading data from permanent memory (hard drive), into the faster and directly accessible temporary memory (RAM), allows the CPU to operate much quicker.

I/O interface:

- The input/output processor (I/O processor) is a processor that is specially designed to handle only input/output processes for the computer or a computer.
- The IOP can fetch and execute its instructions. These IOP instructions are designed to manage only I/O transfers.
- Input-output devices are very slow devices; therefore, they are not directly connected with the system bus because input-output devices are electromagnetic devices. The CPU is an electronic device, so there is a difference in the operating mode, data transfer rate, and word format.
- I/O Module is used to synchronize the input-output devices with the processor.
- The CPU only needs to initiate the I/O processor by specifying what activity is to be performed. Once the required actions are performed, then the I/O processor provides the results to the CPU. Doing these actions allow the I/O processor to act as a bus to the CPU, carrying out activities by directly interacting with memory and other devices in the computer.
- The CPU can act as a master and the IOP act as a slave processor. The processor assigns the task of initiating operations, but the instructions are executed by IOP, not by the CPU. CPU instructions provide operations to begin an I/O transfer. The IOP asks for CPU via an interrupt.

I/O Transfer Modes:

Programmed I/O:

- The CPU directly communicates with the I/O device.
- The processor waits until the completion of the I/O operation.
- Instruction in the program initiates each data item transfer.
- Usually, the transfer is between a CPU register and memory.
- The processor is continuously busy executing the program related to devising functions and waits for the device status to be ready most of the time or function completions (Busy-wait state).

Process:

- CPU requests I/O operation.
- I/O module performs operations.
- I/O module sets status bits.
- CPU checks status bits periodically.
- I/O module does not inform the CPU directly.
- I/O module does not interrupt the CPU.
- CPU may wait or come back later.
- Under programmed I/O, data transfer is very like memory access (CPU viewpoint).
- Each device is given a unique identifier.
- CPU commands contain an identifier (address).



Direct Memory Access (DMA)

Direct memory access (DMA) is a data transfer mode between the memory and I/O devices.

Thus, the peripherals directly communicate and transfer information with each other using the memory buses, removing the intervention of the CPU.

Such a data transfer technique is known as DMA or direct memory access.

Working of DMA

- Whenever an I/O device wants to transmit the data to or from memory, a DMA request (DRQ) is sent by the I/O device to the DMA controller.
- The DMA controller accepts this DRQ and asks the CPU to hold a few clock cycles by sending it the Hold request (HLD).
- CPU receives the Hold request (HLD) from the DMA controller, relinquishes the bus, and sends the Hold acknowledgement (HLDA) to the DMA controller.
- After receiving the Hold acknowledgement (HLDA), the DMA controller acknowledges the I/O device (DACK) that the data transmission can be performed. The DMA controller takes charge of the bus and transmits the data to or from memory.
- When the data transmission is accomplished, the DMA raises an interrupt to let know the processor know that the data transfer task is finished, and the processor can take control over the bus again and start processing where it has left.

