CHAP-1   BASIC STRUCTURE OF COMPUTER HARDWARE

1) Define functional units of computer. (7/2015,2016) (5/2017) (6/2018(w))

Ans- Functional units of computer.

Ans-FUNCTIONAL UNITS OF COMPUTER ?

• Input Unit

• Output Unit

• Central processing Unit (ALU and Control Units)

• Memory

• Bus Structure

A Functional Unit is defined as a collection of computer systems and network infrastructure components which, when abstracted, can be more easily and obviously linked to the goals and objectives of the enterprise, ultimately supporting the success of the enterprise's mission. From a technological perspective, a Functional Unit is an entity that consists of computer systems and network infrastructure components that deliver critical information assets,1 through network-based services, to constituencies that are authenticated to that Functional Unit.

Input device : An input device is usually a keyboard or mouse, the input device is the conduit through which data and instructions enter a computer. A personal computer would be useless if you could not interact with it because the machine could not receive instructions or deliver the results of its work. Input devices accept data and instructions from the user or from another computer system (such as a computer on the Internet).

Output devices return processed data to the user or to another computer system.

common input device is the **keyboard**, which accepts letters, numbers, and commands from the user. **mouse,** which lets you select options from on-screen menus. You use a mouse by moving it across a flat surface and pressing its buttons. A variety of other input devices work with personal computers, too**: The trackball** and touchpad are variations of the mouse and enable you to draw or point on the screen. The **joystick** is a swiveling lever mounted on a stationary base that is well suited for playing video games.

Central processing unit (CPU) — The part of the computer that executes program instructions is known as the processor or central processing unit (CPU). In a microcomputer, the CPU is on a single electronic component, the microprocessor chip, within the system unit or system cabinet.

The system unit also includes circuit boards, memory chips, ports and other components. A microcomputer system cabinet will also house disk drives, hard disks, etc., but these are considered separate from the CPU. This is principal part of any digital computer system, generally composed of control unit, and arithmetic-logic unit the _heart‖ of the computer. It constitutes the physical heart of the entire computer system; to it is linked various peripheral equipment, including input/output devices and auxiliary storage units o

Control Unit is the part of a CPU or other device that directs its operation. The control unit tells the rest of the computer system how to carry out a program's instructions. It directs the movement of electronic signals between memory—which temporarily holds data, instructions and processed information—and the ALU. It also directs these control signals between the CPU and input/output devices. The control unit is the circuitry that controls the flow of information through the processor, and coordinates the activities of the other units within it. In a way, it is the "brain", as it controls what happens inside the processor, which in turn controls the rest of the PC.

Arithmetic-Logic Unit usually called the ALU is a digital circuit that performs two types of operations— arithmetic and logical. Arithmetic operations are the fundamental mathematical operations consisting of addition, subtraction, multiplication and division. Logical operations consist of comparisons. That is, two pieces of data are compared to see whether one is equal to, less than, or greater than the other. The ALU is a fundamental building block of the central processing unit of a computer.

Memory— Memory enables a computer to store, at least known as the primary storage or main memory—is a part of the microcomputer that holds data for processing, instructions for processing the data (the program) and information (processed data). Part of the contents of the memory is held only temporarily, that is, it is stored only as long as the microcomputer is turned on. When you turn the machine off, the contents are lost. The capacity of the memory to hold data and program instructions varies in different computers. The original IBM PC could hold approximately 6,40,000 characters of data or instructions only. But modern microcomputers can hold millions, even billions of characters in their memory.

2) What is bus structure , Explain the basic parameter of Bus design. Write function of each type of bus.   (2019(s) Q4)

Ans-
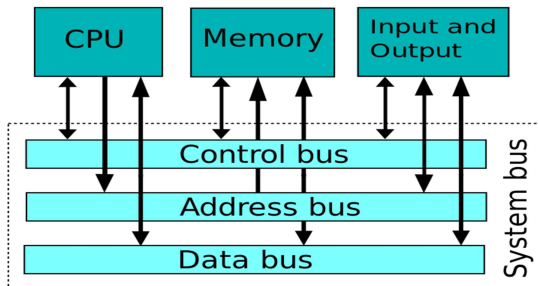
Bus Structure

A *bus* is a collection of wires that connect several devices within a computer system. When a word of data is transferred between units, all its bits are transferred in parallel. A computer must have some lines for addressing and control purposes.

Three main groupings of lines:

The CPU moves data around the computer on pathways that interconnect it to all the other components on the motherboard. These pathways are called 'buses'.

The internal bus carries data within the motherboard.

External buses carry data to peripherals and other devices attached to the motherboard.

The lines or pins of a bus are of three types:

Address - the components pass memory addresses to one another over the address bus.

Control - used to send out signals to coordinate and manage the activities of the motherboard components.

Data - transferred between peripherals, memory and the CPU. Obviously, the data bus can be a very busy pathway.

ELEMENTS /PARAMETERS OF BUS DESIGNS

1) Bus Types

2) Method of Arbitration

3) Timing

4) Bus Width

5) Data Transfer Type

6) Block Data Transfer

1) <u>Bus Types</u>

A) **Dedicated**

- A line is permanently assigned either to one function.
- An example of functional dedication is the use of separate dedicated address and data line.

B) **Multiplexed**
- Using the same lines for multiple purpose.
- Eg:- Address and data information may be transmitted over the same set of lines.
- At the beginning of the data transfer the address is placed   on the bus and the address valid line is activated.
- The address is then remove from the same bus line is used for data transfer.

C) **Physical Dedication**
- The use of multiple buses,each of which connects to only a subset of modules.

2) <u>Method of Arbitration</u>
- Determining who can use the bus at a particular time.

A) **Centralized**
- A single hardware device called the bus controller or arbiter allocate time on the bus.
- The device may be a separate or a part of a processor.

B) **Distributed**
- There is no centralized controllers.
- Each module contains assess control logic and the modules act together.

3) <u>Timing</u>

A) **Synchronous Timing**

- Bus includes a clock line upon which a clock transmits a regular sequence of alternating 1's and 0's
- A single 1-0 transition is referred to as a clock cycle or bus cycle.
- All other devices on the bus can read the clock line.
- All events start at the beginning of a clock cycle

B) **Asynchronous Timing**

- The occurrence of one event on abus follows and depends on the occurrence of a previous event.
- Harder to implement and text than synchronous timing.

4) **Bus Width**

- The width of data bus has an impact on the databus has an impact on the system performance.
- The wider data bus, the greater number of bit transferred at one time.
- The wider address bus, the greater range of location that can be referenced.

5) **Data Transfer Type**
- **Read-Modify-Write :** A read followed immediately by a write to the same address.
- **Read-After-Write :** Consisting of a write followed immediately by a read from the same address (for error checking purposes).

6) **Block Data Transfer**

- One address cycle followed by n data cycles.
- First data item to or from specified address.
- Remaining data items to or from subsequent addresses.

3)Performance measures ?   (3.5/2016,2017)

Computer performance is the amount of work accomplished by a computer system. Depending on the context, high computer performance may involve one or more of the following:

- Short response time for a given piece of work
- High throughput (rate of processing work)
- Low utilization of computing resource(s)
- High availability of the computing system or application
- Fast (or highly compact) data compression and decompression
- High bandwidth
- Short data transmission time

  Response time (execution time, latency): how long the computer responds to user input or finishes a

   • **Response time, execution time and throughput utilized for evaluate an entire computing work –**

- program; the sooner the better • A user sees the result in 5 second after inputting the keyword to a library database system • A simulation program finishes in one hour COMPUTER ARCHITECTURE 5
- • A simulation program finishes in one hour – Throughput: Amount of work done in a given time.
- **Throughput** = how much work a computer can finish for a given time; the higher the better • A web server serves up to 5 million requests per second

• Bandwidth and latency for memory performance

• when a program runs in less time (will say a computer user) – the computer user is interested in reducing response time also referred to as execution time or latency.

• when computer completes more jobs in an hour (will say computer centre manager) – the computer centre manager is interested in increasing throughput - the total amount of work done in a given time - sometimes called bandwidth.

3) **What do you mean by Bus . Explain peripheral components to interconnect bus.(7/2017,5/2018)**
   Ans- A collection of wires through which data is transmitted from one part of a computer to another.
    A collection of wires through which data is transmitted from one part of a computer to another.

Computer Hardware Peripherals

Introduction

A peripheral is a piece of computer hardware that is added to a computer in order to expand its abilities. The term peripheral is used to describe those devices that are optional in nature, as opposed to hardware that is either demanded or always required in principle. There are all different kinds of peripherals you can add your computer. The main disctinction among peripherals is the way they are connected to your computer. They can be connected internally or externally.

Buses

A bus is a subsystem that transfers data between computer components inside a computer or between computers. Unlike a point–to–point connection, a bus can logically connect several peripherals over the same set of wires. Each bus defines its set of connectors to physically plug devices, cards or cables together. There are two types of buses: internal and external. Internal buses are connections to various internal components. External buses are

connections to various external components. There are different kinds of slots that internal and external devices can connect to.

Internal

Types of Slots

There are many different kinds of internal buses, but only a handful of popular ones. Different computers come with different kinds and number of slots. It is important to know what kind and number of slots you have on your computer before you go out and by a card that matches up to a slot you don't have.

PCI

PCI (Peripheral Component Interconnect) is common in modern PCs. This kind of bus is being succeeded by PCI Express. Typical PCI cards used in PCs include: network cards, sound cards, modems, extra ports such as USB or serial, TV tuner cards and disk controllers. Video cards have outgrown the capabilities of PCI because of their higher bandwidth requirements.

PCI Slots

PCI Express

PCI Express was introduced by Intel in 2004. It was designed to replace the general-purpose PCI expansion bus and the AGP graphics card interface. PCI express is not a bus but instead a point-to-point conection of serial links called lanes. PCI Express cards have faster bandwidth then PCI cards which make them more ideal for high-end video cards.

PCMCIA

PCMCIA (also referred to as PC Card) is the type of bus used for laptop computers. The name PCMCIA comes from the group who developed the

standard: Personal Computer Memory Card International Association. PCMCIA was originally designed for computer memory expansion, but the existence of a usable general standard for notbeook peripherals led to many kinds of devices being made available in this form. Typical devices include network cards, modems, and hard disks.

AGP

AGP (Accelerated Graphics Port) is a high-speed point-to-point channel for attaching a graphics card to a computer's motherboard, primarily to assist in the acceleration of 3D computer graphics. AGP has been replaced over the past couple years by PCI Express. AGP cards and motherboards are still available to buy, but they are becoming less common.

AGP Slot

Types Of Cards

Video Card

A video card (also known as graphics card) is an expansion card whose function is to generate and output images to a display. Some video cards offer added functions, such as video capture, TV tuner adapter, ability to connect multiple monitors, and others. Most video cards all share similar components. They include a graphics processing unit (GPU) which is a dedicated microprocessor optimized for 3D graphics rendering. It also includes a video BIOS that contains the basic program that governs the video card's operations and provides the instructions that allow the computer and software to interface with the card. If the video card is integrated in the motherboard, it may use the computer RAM memory. If it is not it will have its own video memory called Video RAM. This kind of memory can range from 128MB to 2GB. A video card also has a RAMDAC (Random Access Memory Digital-to-Analog Converter) which takes responsibility for turning the digital signals produced by the computer processor into an analog signal which can be understood by the computer display. Lastly, they all have outputs such as an HD-15 connector (standard monitor cable), DVI connector, S-Video, composite video or component video.

Graphics Card

Sound Card

A sound card is an expansion card that facilitates the input and output of audio signals to/from a computer under control of computer programs. Typical uses for sound cards include providing the audio component for multimedia applications such as music composition, editing video or audio, presentation/education, and entertainment. Many computers have sound capabilities built in,, while others require additional expansion cards to provide for audio capability.

Network Card

A network card is an expansion card that allows computers to communicate over a computer network. It allows users to connect to each other either by using cables or wirelessly. Although other network technologies exist, Ethernet has achieved near-ubiquity for a while now. Every Ethernet network card has a unique 48-bit serial number called a MAC address, which is stored in ROM carried on the card. You can learn more about networking in the introduction to networking lesson.


Network Card

External

Types of Connections

USB

USB (Universal Serial Bus) is a serial bus standard to interface devices. USB was designed to allow many peripherals to be connected using a single standardized interface socket and to improve the plug-and-play capabilities by allowing devices to be connected and disconnected without rebooting the computer. Other convient features include providing power to low-consumption devices without the need for an external power supply and allowing many devices to be used without requiring manufacturer specific, individual device drivers to be installed. USB is by far the dominating bus for connecting external devices to your computer.
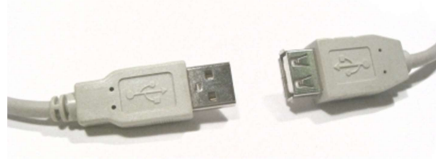
USB Connectors

Firewire

Firewire (technically known as IEEE 1394 and also known as i.LINK for Sony) is a serial bus interface standard for high-speed communications and isochronous real-time data transfer, frequently used in a personal computer. Firewire has replaced Parallel ports in many applications. It has been adopted as the High Definition Audio-Video Network Alliance (HANA) standard connection interface for A/V (audio/visual) component communication and control. Almost all modern digital camcorders have included this connection.


Firewire Cable

PS/2

The PS/2 connector is used for connecting some keyboards and mice to a PC compatible computer system. The keyboard and mouse interfaces are electrically similar with the main difference being that open collector outputs

are required on both ends of the keyboard interface to allow bidirectional communication. If a PS/2 mouse is connected to a PS/2 keyboard port, the mouse may not be recognized by the computer depending on configuration.
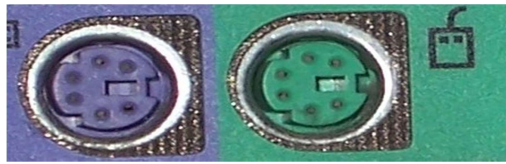
PS/2 Ports

Devices

Removable Storage

The same kinds of CD and DVD drives that could come built-in on your computer can also be attached externally. You might only have a CD-ROM drive built-in to your computer but you need a CD writer to burn CDs. You can buy an external CD writer that connects to your USB port and acts the same way as if it was built-in to your computer. The same is true for DVD writers, Blu-ray drives, and floppy drives. Flash drives have become very popular forms of removable storage especially as the price of flash drives decreases and the possible size for them increases. Flash drives are usually USB ones either in the form USB sticks or very small, portable devices. USB flash drives are small, fast, removable, rewritable, and long-lasting. Storage capacities range from 64MB to 32GB or more. A flash drive does not have any mechanically driven parts so as opposed to a hard drive which makes it more durable and smaller usually.

USB Flash Drive

Non-removable Storage

Non-removable storage can be a hard drive that is connected externally. External hard drives have become very popular for backups, shared drives among many computers, and simply expaning the amount of hard drive space you have from your internal hard drive. External hard drives come in many

shapes and sizes like flash drives do. An external hard drive is usually connected by USB but you can also have a networked hardrive which will connect to your network which allows all computers on that network to access that hard drive.

Input

Input devices are absolutely crucial to computers. The most common input devices are mice and keyboards which barely every computer has. A new popular pointing device that may eventually replace the mouse is touch screen which you can get on some tablet notebooks. Other popular input devices include microphones, webcams, and fingerprint readers which can also be built in to modern laptops and desktops. A scanner is another popular input device that might be built-in to your printer.

Webcam

Output

There are lots of different kinds of output devices that you can get for your computer. The absolute most common external output device is a monitor. Other very popular output devices are printers and speakers. There are lots of different kinds of printers and different sizes of speakers for your computer. Monitors are connected usually through the HD-15 connector on your video card. Printers are usually connected through a USB port. Speakers have their own audio out port built-in to the sound card.

4) **Name 4-peripheral devices used in computer. (2/2016)**

Ans- There are many different peripheral devices, but they fall into three general categories:

Input devices, such as a **mouse** and a **keyboard**.

Output devices, such as a **monitor** and a **printer**.

Storage devices, such as a **hard drive** or **flash drive**.

CHAP-2   INSTRUCTION AND INSTRUCTION SEQUENCING

1) Define operand and opcode . (5/2016,3/2017,2(v)-2018(w))

Ans- an **operand** is the part of a **computer** instruction which specifies what data is to be manipulated or operated on, while at the same time representing the data itself.

an **opcode** (operation code), is the portion of a machine language instruction that specifies the operation to be performed.

Example[edit]

The following arithmetic expression shows an example of operators and operands:

In the above example, '+' is the symbol for the operation called addition.

The **operand** '3' is one of the inputs (quantities) followed by the addition operator, and the operand '6' is the other input necessary for the operation.

The result of the operation is 9. (The number '9' is also called the sum of the augend 3 and the addend 6.)

An operand, then, is also referred to as "one of the inputs (quantities) for an operation".

2) Explain Instruction formats.   (5/2017,2018(6),2019(s)-2(iii))
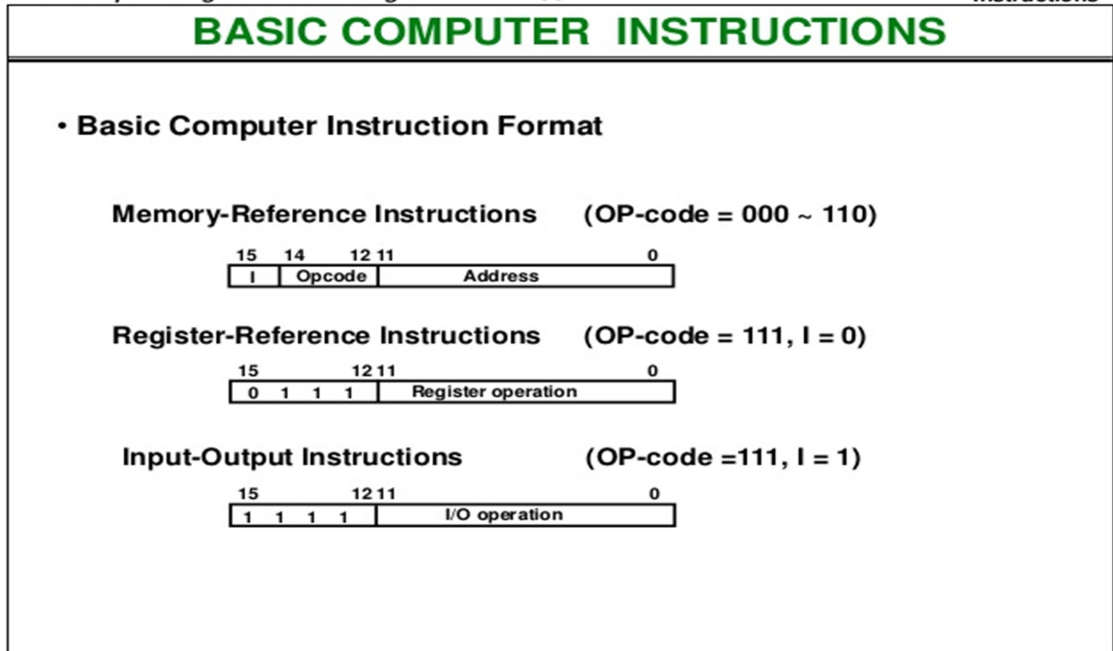
Ans-Format of Instruction

a program must be written, translated into machine language, and loaded into memory; that such a program consists of machine instructions, and that a machine instruction is a binary number. In this chapter we look at the details of machine instructions, and in particular the fact that a machine instruction is rarely a single binary number. It normally consists of several numbers that are assembled (placed together) by the assembler, to become the fields of the complete instruction. As a result, any machine instruction has a certain format.

The instruction is divided into fields, each a binary number. A machine instruction must contain at least one field, namely the operation code (opcode), that tells the control unit what the instruction is supposed to do. Most instructions contain other fields specifying registers, memory addresses, immediate quantities, addressing modes, and other parameters.

Figure 2.1: Various Instruction Formats

Figure 2.1 illustrates typical instruction formats. They range from the simplest to the very complex and they demonstrate the following properties of machine instructions:

- **Instructions** can have different sizes. The size depends on the number and nature of the individual fields.
- The **opcode** can have different sizes. The opcode can also be broken up into a number of separate fields.
- A **field** containing an address is much larger than a register field. This is because number of registers is small, while size of memory is large.
- Fields containing immediate **operands** (numbers used as data) can have different sizes. Experience shows that most data items used by programs are small. Thus, a well-designed instruction set should allow for both small and large data items, resulting in short instructions whenever possible.
-

BASIC COMPUTER INSTRUCTIONS

• Basic Computer Instruction Format

Memory-Reference Instructions   (OP-code = 000 ~ 110)

15	14	12 11		0
I	Opcode		Address	

Register-Reference Instructions   (OP-code = 111, I = 0)

15		12 11		0
0	1 1 1		Register operation	

Input-Output Instructions   (OP-code =111, I = 1)

15		12 11		0
1	1 1 1		I/O operation	

Computer Organization   *Computer Architectures Lab*

-

3)Types of instruction formats. (7/2017,2019(s)-2(ii))

Ans-**TYPES OF INSTRUCTION CODE FORMAT USED IN COMPUTER**

Instruction Format:

An instruction format defines the layout of bits of an instruction in terms of constituent parts. There are various instruction format's depending upon the architecture of the computer. The types of commonly used instructions are:

- Three-Address Instruction.
- Two-Address Instruction.
- One-Address Instruction.
- Zero-Address Instruction.

Three-Address Instruction:

A three address instruction consists of the following parts:

- Operation code.
- Addresses of two operands, called address 1 and address 2.
- Address of the memory location where the result of the operation is to be stored i.e., address of the destination.

The number of bits (field length) allocated to each of the three parts depend upon the computer. A typical three address instruction is shown below.

OP code	Address 1	Address 2	Address of Destination

A register called program counter (PC) is used to computer the address of the next instruction and the process of the execution of instruction continues.

Two Address Instruction:

In this type of instruction one operand is placed in a specified register such as an accumulator and the address of the next instruction is obtained from another register called program counter (PC). This implies that such an instruction should have the following parts.

- Operation code.
- Address of one of the operands, say address 1.
- Address of the storage location where the result is to be stored. This address is denoted by address 2.

The general form of a two address instruction is

OP code   Address 1   Address 2

One Address Instruction:
As the name suggests, this instruction has address of one operand only, the other operand is stored in accumulator. The result of operation are left in the accumulator it self, from where these can be moved to main memory by another instruction. The address of the next instruction is obtained from the program counter. The general form of a one-address instruction is

OP code   Address 1

No doubt with one address instruction more bits of the instruction word could be allocated to OP code and the address of the operand.

Zero Address Instruction:
The zero address instructions are also called stack instructions and consist of OP code only. The address of operand and destination are implied. The general form of zero address instruction is.

OP code

4)Define types of Addressing Modes (7/2017,5/2018)

Immediate Mode

In this mode, the operand is specified in the instruction itself. An immediate mode instruction has an operand field rather than the address field.

For example: `ADD 7`, which says Add 7 to contents of accumulator. 7 is the operand here.

Register Mode

In this mode the operand is stored in the register and this register is present in CPU. The instruction has the address of the Register where the operand is stored.

Advantages of this mode:

- Shorter instructions and faster instruction fetch.

- Faster memory access to the operand(s)

Disadvantages of this mode:

- Very limited address space
- Using multiple registers helps performance but it complicates the instructions.

Register Indirect Mode

In this mode, the instruction specifies the register whose contents give us the address of operand which is in memory. Thus, the register contains the address of operand rather than the operand itself.

Auto Increment/Decrement Mode

In this the register is incremented or decremented after or before its value is used.

Direct Addressing Mode

In this mode, effective address of operand is present in instruction itself.

- Single memory reference to access data.
- No additional calculations to find the effective address of the operand.

For Example: `ADD R1, 4000` - In this the 4000 is effective address of operand.

NOTE: Effective Address is the location where operand is present.

Indirect Addressing Mode

In this, the address field of instruction gives the address where the effective address is stored in memory. This slows down the execution, as this includes multiple memory lookups to find the operand.

Displacement Addressing Mode

In this the contents of the indexed register is added to the Address part of the instruction, to obtain the effective address of operand.

`EA = A + (R)`, In this the address field holds two values, A(which is the base value) and R(that holds the displacement), or vice versa.

Relative Addressing Mode

It is a version of Displacement addressing mode.

In this the contents of PC(Program Counter) is added to address part of instruction to obtain the effective address.

`EA = A + (PC)`, where EA is effective address and PC is program counter.

The operand is A cells away from the current cell(the one pointed to by PC)

Base Register Addressing Mode

It is again a version of Displacement addressing mode. This can be defined as `EA = A + (R)`, where A is displacement and R holds pointer to base address.

Stack Addressing Mode

In this mode, operand is at the top of the stack. For example: `ADD`, this instruction will *POP* top two items from the stack, add them, and will then *PUSH* the result to the top of the stack.

5) What is Interrupt. Explain interrupt handling methods

(2/2015 , 5/2016, 2/2017,2019(s)-2-vii)

Definition of Interrupt

Ans-An interrupt is a function of an operating system that provides multi-process multi-tasking. The interrupt is a signal that prompts the operating system to stop work on one process and start work on another. Many different kinds of interrupts enhance the capability of an operating system to provide on-demand services to users while handling more and more additional computing tasks in the background.

Types of Interrupts:

Although interrupts have highest priority than other signals, there are many type of interrupts but basic type of interrupts are

1. **Hardware Interrupts:** If the signal for the processor is from external device or hardware is called hardware interrupts. Example: from keyboard we will press the key to do some action this pressing of key in keyboard will generate a signal which is given to the processor to do action, such interrupts are called hardware interrupts. Hardware interrupts can be classified into two types they are
   - **Maskable Interrupt:** The hardware interrupts which can be delayed when a much highest priority interrupt has occurred to the processor.
   - **Non Maskable Interrupt:** The hardware which cannot be delayed and should process by the processor immediately.

   Software Interrupts: Software interrupt can also divided in to two types. They are
   - **Normal Interrupts:** the interrupts which are caused by the software instructions are called software instructions.
   - **Exception:** unplanned interrupts while executing a program is called Exception. For example: while executing a program if we got a value which should be divided by zero is called a exception.

Classification of Interrupts According to Periodicity of Occurrence:

1. **Periodic Interrupt:** If the interrupts occurred at fixed interval in timeline then that interrupts are called periodic interrupts
2. **Aperiodic Interrupt:** If the occurrence of interrupt cannot be predicted then that interrupt is called aperiodic interrupt.

Classification of Interrupts According to the Temporal Relationship with System Clock:

1. **Synchronous Interrupt:** The source of interrupt is in phase to the system clock is called synchronous interrupt. In other words interrupts which are

dependent on the system clock. Example: timer service that uses the system clock.

2. **Asynchronous Interrupts:** If the interrupts are independent or not in phase to the system clock is called asynchronous interrupt.

6) How Instructions is executed Explain steps of Fetch and Execution cycle. (5/2017)(5/2017),(2019(s)-2-iv)

Ans-An **instruction** cycle (sometimes called a fetch–decode–**execute** cycle) is the basic operational process of a computer. It is the process by which a computer retrieves a program **instruction** from its memory, determines what actions the **instruction** dictates, and carries out those actions.

The fetch-decode-execute cycle

A standard process describes the steps needed for processing to take place. It is called the Fetch - Decode - Execute cycle or sometimes simply called the Fetch-Execute Cycle.

First of all, both the data and the program that acts upon that data are loaded into main memory (RAM) by the operating system. The CPU is now ready to do some work.

1. FETCH

The first step the CPU carries out is to fetch some data and instructions (program) from main memory then store them in its own internal temporary memory areas. These memory areas are called 'registers'.

STEPS OF The Fetch Execute Cycle

There are several steps in the Fetch Execute cycle that are executed in order. The CPU goes through this cycle once for each instruction that is executed.

1. Fetch the instruction. Each CPU has a register which contains the current address of execution. This register is usually called the PC for Program Counter. This address is given to the memory with the fetch command to get the instruction to be executed.

2. Decode the instruction. Each instruction contains an operation code, which says what type of instruction it is. This operation code also determines what operands are needed. For instance, an add instruction needs some operands to sum, while a halt instruction needs no operands.

3. Increment the PC. Once the type of the instruction is known, the length of the instruction can be calculated. This length is then added to the Program Counter so

that at the beginning of the next instruction, the current memory location will be fetched.

4. Fetch the operands. Operands of an instruction may be in memory or CPU registers. If they are in memory, the memory needs to fetch them at this time, since all work like arithmetic occurs in the CPU. This step is not needed for instructions with no memory operands.

5. Execute the instruction. Once the operands are present in the CPU, the computation may proceed. This is often an arithmetic operation, but could also be a comparison or many other things.

6. Store the results. If any values need to be saved in memory, this is now done. The next step is to go back to step one and do it all over again.

Various processors may have numerous variations on how and when each step is done, but all computers follow the same basic pattern.
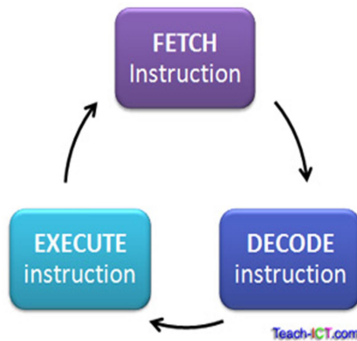
This is called the 'fetch' part of the cycle.

For this to happen, the CPU makes use of a vital hardware path called the 'address bus'.

The CPU places the address of the next item to be fetched on to the address bus.

Data from this address then moves from main memory into the CPU by travelling along another hardware path called the 'data bus'.

You could imagine that it is a bit like a boat attendant at a lake calling in customers when their time is up -- "Boat number 3, time to come in!" The 'address' of the boat is 3 and the 'data' is its content. The boat is parked at a pier, which is like the internal register of the CPU.

2. DECODE

The next step is for the CPU to make sense of the instruction it has just fetched.

This process is called 'decode'.

The CPU is designed to understand a specific set of commands. These are called the 'instruction set' of the CPU. Each make of CPU has a different instruction set.

The CPU decodes the instruction and prepares various areas within the chip in readiness of the next step.

3. EXECUTE

This is the part of the cycle when data processing actually takes place. The instruction is carried out upon the data (executed). The result of this processing is stored in yet another register.

Once the execute stage is complete, the CPU sets itself up to begin another cycle once more.

CHAP-4   PROCESSOR SYSTEM

1) What is the function of control unit.   (2/2015) (5/2017)(2018(w)-2-vi)

Ans- Control Unit:

Control unit co-ordinates the transfer of data between _registers of CPU_ or microprocessor and _ALU_. Control unit serves the instructions for _ALU_. Along with this, control unit, as its name implies, controls every other parts of the machine, their co-ordinations, traffic etc.

Functions of Control Unit
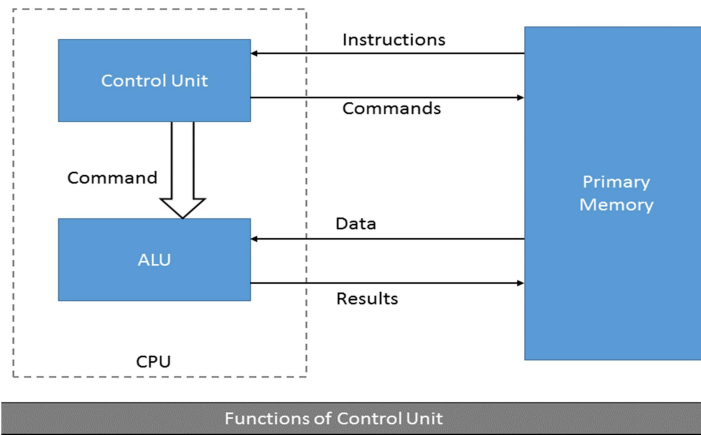
Functions of Control Unit:

- Fetching instructions one by one from primary memory and gather required data and operands to perform those instructions.
- Sending instructions to ALU to perform additions, multiplication etc.
- Receiving and sending results of operations of ALU to primary memory
- Fetching programs from input and secondary memory and bringing them to primary memory
- Sending results from ALU stored in primary memory to output

2) **What is the function of register . Write different type of registers available along with their function. (5/2015),(2017(s)-6)(2018(w)-2-viii)**

Ans- A processor *register* (CPU *register*) is one of a small set of data holding places that are part of the computer processor. A *register* may hold an instruction, a storage address, or any kind of data (such as a bit sequence or individual characters). Some instructions specify *registers* as part of the instruction.

FUNCTION OF REFGISTER

A Special *Function Register* (or Special *Purpose Register*, or simply Special *Register*) is a *register* within a microprocessor, which controls or monitors various aspects of the microprocessor's *function*. Depending on the processor architecture, this can include, but is not limited to: ... subroutine return address

Types of Registers are as Followings

MAR stand for Memory Address Register

This register holds the memory addresses of data and instructions. This register is used to access data and instructions from memory during the execution phase of an instruction. Suppose CPU wants to store some data in the memory or to read the data from the memory. It places the address of the-required memory location in the MAR.

Program Counter

The program counter (PC), commonly called the instruction pointer (IP) in Intel x86 microprocessors, and sometimes called the instruction address register, or just part of the instruction sequencer in some computers, is a processor register

It is a 16 bit special function register in the 8085 microprocessor. It keeps track of the the next memory address of the instruction that is to be executed once the execution of the current instruction is completed. In other words, it holds the address of the memory location of the next instruction when the current instruction is executed by the microprocessor.

Accumulator Register

This Register is used for storing the Results those are produced by the System. When the CPU will generate Some Results after the Processing then all the Results will be Stored into the AC Register.

Memory Data Register (MDR)

MDR is the register of a computer's control unit that contains the data to be stored in the computer storage (e.g. RAM), or the data after a fetch from the computer storage. It acts like a buffer and holds anything that is copied from the memory ready for the processor to use it. MDR hold the information before it goes to the decoder.

MDR which contains the data to be written into or readout of the addressed location. For example, to retrieve the contents of cell 123, we would load the value 123 (in binary, of course) into the MAR and perform a fetch operation. When the operation is done, a copy of the contents of cell 123 would be in the MDR. To store the value 98 into cell 4, we load a 4 into the MAR and a 98 into the MDR and perform a store. When the operation is completed the contents of cell 4 will have been set to 98, by discarding whatever was there previously.

The MDR is a two-way register. When data is fetched from memory and placed into the MDR, it is written to in one direction. When there is a write instruction, the data to be written is placed into the MDR from another CPU register, which then puts the data into memory.

The Memory Data Register is half of a minimal interface between a micro program and computer storage, the other half is a memory address register.

Index Register

A hardware element which holds a number that can be added to (or, in some cases, subtracted from) the address portion of a computer instruction to form an effective address. Also known as base register. An index register in a computer's CPU is a processor register used for modifying operand addresses during the run of a program.

Memory Buffer Register

MBR stand for *Memory Buffer Register*. This register holds the contents of data or instruction read from, or written in memory. It means that this register is used to store data/instruction coming from the memory or going to the memory.

Data Register

A register used in microcomputers to temporarily store data being transmitted to or from a peripheral device.

**3) What do you mean by microprogram control. Explain in detail.
.(7/2015) (5/2017),(2019(s)-2-v)**
      Ans-

Microprograms were organized as a sequence of microinstructions and stored in special control memory. The algorithm for the *microprogram control unit* is usually specified by flowchart description. The main advantage of the *microprogram control unit* is the simplicity of its structure.

Micro-programmed Control Unit –
- The control signals associated with operations are stored in special memory units inaccessible by the programmer as Control Words.
- Control signals are generated by a program are similar to machine language programs.
- Micro-programmed control unit is slower in speed because of the time it takes to fetch microinstructions from the control memory.

Some Important Terms –
1. **Control Word :** A control word is a word whose individual bits represent various control signals.
2. **Micro-routine :** A sequence of control words corresponding to the control sequence of a machine instruction constitutes the micro-routine for that instruction.
3. **Micro-instruction :** Individual control words in this micro-routine are referred to as microinstructions.
4. **Micro-program :** A sequence of micro-instructions is called a micro-program, which is stored in a ROM or RAM called a Control Memory (CM).
5. **Control Store :** the micro-routines for all instructions in the instruction set of a computer are stored in a special memory called the Control Store.

Types of Micro-programmed Control Unit – Based on the type of Control Word stored in the Control Memory (CM), it is classified into two types :
1. Horizontal Micro-programmed control Unit :
The control signals are represented in the decoded binary format that is 1 bit/CS. Example: If 53 Control signals are present in the processor than 53 bits are required. More than 1 control signal can be enabled at a time.
- It supports longer control word.
- It is used in parallel processing applications.
- It allows higher degree of parallelism. If degree is n, n CS are enabled at a time.
- It requires no additional hardware(decoders). It means it is faster than Verical Microprogrammed.

**2. Vertical Micro-programmed control Unit :**
The control signals re represented in the encoded binary format. For N control signals- Log2(N) bits are required.

- It supports shorter control words.
- It supports easy implementation of new conrol signals therefore it is more flexible.
- It allows low degree of parallelism i.e., degree of parallelism is either 0 or 1.
- Requires an additional hardware (decoders) to generate control signals, it implies it is slower than horizontal microprogrammed.

4) **What is the function of ALU. Explain the design of of ALu with diagram (5/2016) (5/2018(w)),(2019(s)-2-ii)**

Ans- **Design of ALU :**

ALU or Arithmatic Logical Unit is a digital circuit to do arithmatic operations like addition, subtraction,division, multiplication and logical oparations like and, or, xor, nand, nor etc. A simple block diagram of a 4 bit ALU for operations and,or,xor and Add is shown here :
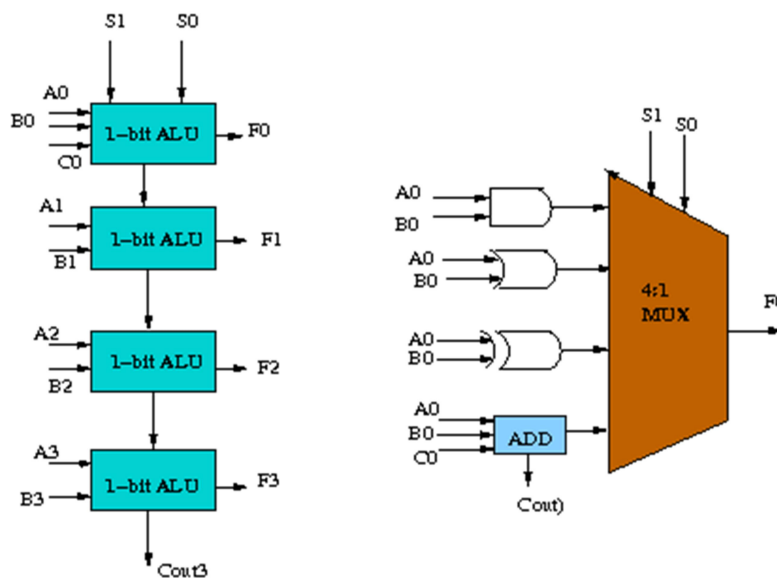
The 4-bit ALU block is combined using 4 1-bit ALU block

Function of ALU

The characteristics of the ALU are as follows:

- The ALU is responsible for performing all logical and arithmetic operations.

- Some of the arithmetic operations are as follows: addition, subtraction, ultiplication and division.

- Some of the logical operations are as follows: comparison between numbers, letter and or special characters.

- The ALU is also responsible for the following conditions: Equal-to conditions, Less-than condition and greater than condition.

5) What do you mean by multiprocessor system . (2/2017)

Ans- *Multiprocessing* is the use of two or more central processing units (CPUs) within a single computer *system.* The key objective of using a multiprocessor is to boost the system's execution speed, with other objectives being fault tolerance and application matching.

A good illustration of a multiprocessor is a single central tower attached to two computer systems. A multiprocessor is regarded as a means to improve computing speeds, performance and cost-effectiveness, as well as to provide enhanced availability and reliability.

6) Distinghish between hardwired control and microprogrammed control .(5/2016)

Ans- To execute an instruction, the control unit of the CPU must generate the required control signal in the proper sequence. There are two approaches used for generating the control signals in proper sequence as Hardwired Control unit and Micro-programmed control unit.

Hardwired Control Unit –
The control hardware can be viewed as a state machine that changes from one state to another in every clock cycle, depending on the contents of the instruction register, the condition codes and the external inputs. The outputs of the state machine are the control signals. The sequence of the operation carried out by this machine is determined by the wiring of the logic elements and hence named as "hardwired".
- Fixed logic circuits that correspond directly to the Boolean expressions are used to generate the control signals.
- Hardwired control is faster than micro-programmed control.
- A controller that uses this approach can operate at high speed.

Micro-programmed Control Unit –
- The control signals associated with operations are stored in special memory units inaccessible by the programmer as Control Words.
- Control signals are generated by a program are similar to machine language programs.
- Micro-programmed control unit is slower in speed because of the time it takes to fetch microinstructions from the control memory.

### 7) Define I/O processor. (3.5/2016) (5/2017),(2019(s)-1-v)

Ans- Input/Output Processor
An input-output processor (IOP) is a processor with direct memory access capability. In this, the computer system is divided into a memory unit and number of processors. Each IOP controls and manage the input-output tasks. The IOP is

similar to CPU except that it handles only the details of I/O processing. The IOP can fetch and execute its own instructions. These IOP instructions are designed to manage I/O transfers only.

Block Diagram Of IOP
Below is a block diagram of a computer along with various I/O Processors. The memory unit occupies the central position and can communicate with each processor. The CPU processes the data required for solving the computational tasks. The IOP provides a path for transfer of data between peripherals and memory. The CPU assigns the task of initiating the I/O program. The IOP operates independent from CPU and transfer data between peripherals and memory.
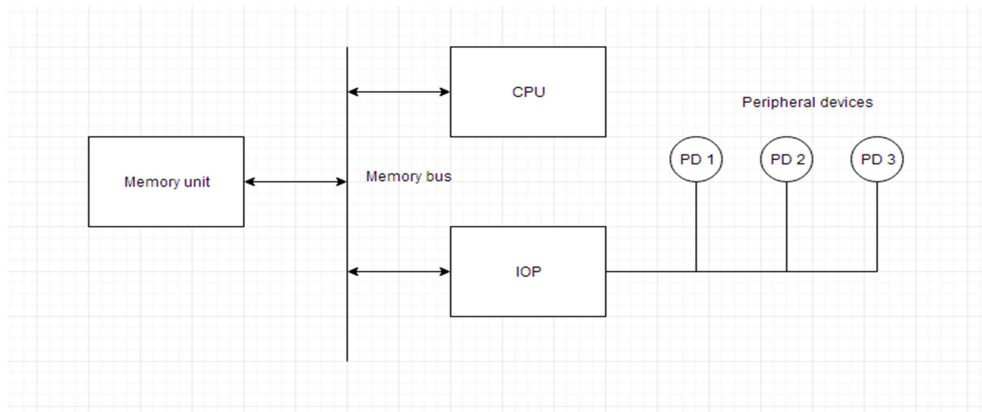
The communication between the IOP and the devices is similar to the program control method of transfer. And the communication with the memory is similar to the direct memory access method.

In large scale computers, each processor is independent of other processors and any processor can initiate the operation.

The CPU can act as master and the IOP act as slave processor. The CPU assigns the task of initiating operations but it is the IOP, who executes the instructions, and not the CPU. CPU instructions provide operations to start an I/O transfer. The IOP asks for CPU through interrupt.

Instructions that are read from memory by an IOP are also called *commands* to distinguish them from instructions that are read by CPU. Commands are prepared by programmers and are stored in memory. Command words make the program for IOP. CPU informs the IOP where to find the commands in memory.

8) **What is throughput .(2/2017)**

Ans- Throughput refers to the performance of tasks by a computing service or device over a specific period. It measures the amount of completed work against time consumed and may be used to measure the performance of a processor, memory and/or network communications.

Chap-5   MEMORY SYSTEM

1) What is the function of main memory (2/2017)

Ans- Primary storage, also known as **main** storage or **memory**, is the area in a computer in which data is stored for quick access by the computer's processor. The terms random access **memory** (RAM) and **memory** are often as synonyms for primary or **main** storage.

2) Define memory characteristics.   (3.5/2017)(2019(s)-1-v)

Ans- The key characteristics of memory devices or memory system are as follows:

1. Location
2. Capacity
3. Unit of Transfer
4. Access Method
5. Performance
6. Physical type
7. Physical characteristics
8. Organization

1. Location:

It deals with the location of the memory device in the computer system. There are three possible locations:

- CPU : This is often in the form of CPU registers and small amount of cache
- Internal or main: This is the main memory like RAM or ROM. The CPU can directly access the main memory.
- External or secondary: It comprises of secondary storage devices like hard disks, magnetic tapes. The CPU doesn't access these devices directly. It uses device controllers to access secondary storage devices.

2. Capacity:

The capacity of any memory device is expressed in terms of: i)word size ii)Number of words

- **Word size:** Words are expressed in bytes (8 bits). A word can however mean any number of bytes. Commonly used word sizes are 1 byte (8 bits), 2bytes (16 bits) and 4 bytes (32 bits).

- **Number of words:** This specifies the number of words available in the particular memory device. For example, if a memory device is given as 4K x 16.This means the device has a word size of 16 bits and a total of 4096(4K) words in memory.

3. Unit of Transfer:

It is the maximum number of bits that can be read or written into the memory at a time. In case of main memory, it is mostly equal to word size. In case of external memory, unit of transfer is not limited to the word size; it is often larger and is referred to as blocks.

4. Access Methods:

It is a fundamental characteristic of memory devices. It is the sequence or order in which memory can be accessed. There are three types of access methods:

- **Random Access:** If storage locations in a particular memory device can be accessed in any order and access time is independent of the memory location being accessed. Such memory devices are said to have a random access mechanism. RAM (Random Access Memory) IC's use this access method.

- **Serial Access:** If memory locations can be accessed only in a certain predetermined sequence, this access method is called serial access. Magnetic Tapes, CD-ROMs employ serial access methods.

- **Semi random Access:** Memory devices such as Magnetic Hard disks use this access method. Here each track has a read/write head thus each track can be accessed randomly but access within each track is restricted to a serial access.

5. Performance:
The performance of the memory system is determined using three parameters

- **Access Time:** In random access memories, it is the time taken by memory to complete the read/write operation from the instant that an address is sent to the memory. For non-random access memories, it is the time taken to position the read write head at the desired location. Access time is widely used to measure performance of memory devices.

- **Memory cycle time:** It is defined only for Random Access Memories and is the sum of the access time and the additional time required before the second access can commence.

- **Transfer rate:** It is defined as the rate at which data can be transferred into or out of a memory unit.

6. Physical type:
Memory devices can be either semiconductor memory (like RAM) or magnetic surface memory (like Hard disks).

7.Physical Characteristics:

- **Volatile/Non- Volatile:** If a memory devices continues hold data even if power is turned off. The memory device is non-volatile else it is volatile.

8. Organization:

- **Erasable/Non-erasable:** The memories in which data once programmed cannot be erased are called Non-erasable memories. Memory devices in which data in the memory can be erased is called erasable memory.

  E.g. RAM(erasable), ROM(non-erasable).

3) Define semiconductor memory. (3.5/2017)(2019(s)-2-vi)

Ans- Semiconductor memory technology is an essential element of today's electronics. Normally based around semiconductor technology, memory is used in any equipment that uses a processor of one form or another.

With the rapid growth in the requirement for semiconductor memories there have been a number of technologies and types of memory that have emerged. Names such as ROM, RAM, EPROM, EEPROM, Flash memory, DRAM, SRAM, SDRAM .

Types of semiconductor memory

Electronic semiconductor memory technology can be split into two main types or categories, according to the way in which the memory operates:

- *RAM - Random Access Memory:* As the names suggest, the RAM or random access memory is a form of semiconductor memory technology that is used for reading and writing data in any order as required. It is used for such applications as the computer or processor memory where variables and other stored and are required on a random basis. Data is stored and read many times to and from this type of memory.

- *ROM - Read Only Memory:* A ROM is a form of semiconductor memory technology used where the data is written once and then not changed. In view of this it is used where data needs to be stored permanently, even when the power is removed - many memory technologies lose the data once the power is removed.

  As a result, this type of semiconductor memory technology is widely used for storing programs and data that must survive when a computer or processor is powered down. For example the BIOS of a computer will be stored in ROM. As the name implies, data cannot be easily written to ROM. Depending on the technology used in the ROM, writing the data into the ROM initially may require special hardware. Although it is often possible to change the data, this gain requires special hardware to erase the data ready for new data to be written in.

  Both of these categories of semiconductor technology are widely used. Each type being used in different areas and applications within microprocessor based electronics systems.

3) State and Explain interleaved memory organisation .(5/2017),2018(w)-2-vii

Ans-Main memory refers to physical memory that is internal to the computer. The word *main* is used to distinguish it from external mass storage devices such as disk drives. Other terms used to mean main memory include *RAM* and primary storage.

The computer can manipulate only data that is in main memory. Therefore, every program you execute and every file you accessmust be copied from a storage device into main memory. The amount of main memory on a computer is crucial because it determines how many programs can be executed at one time and how much data can be readily available to a program.

Interleaved Memory

It is a technique for compensating the relatively slow speed of DRAM(Dynamic RAM). In this technique, the main memory is divided into memory banks which can be accessed individually without any dependency on the other.

For example: If we have 4 memory banks(4-way Interleaved memory), with each containing 256 bytes, then, the Block Oriented scheme(no interleaving), will assign virtual address 0 to 255 to the first bank, 256 to 511 to the second bank. But in Interleaved memory, virtual address 0 will be with the first bank, 1 with the second memory bank, 2 with the third bank and 3 with the fourt, and then 4 with the first memory bank again.

Hence, CPU can access alternate sections immediately without waiting for memory to be cached. There are multiple memory banks which take turns for supply of data.

Memory interleaving is a technique for increasing memory speed. It is a process that makes the system more efficient, fast and reliable. For example: In the above example of 4 memory banks, data with virtual address 0, 1, 2 and 3 can be accessed simultaneously as they reside in spearate memory banks, hence we do not have to wait for completion of a data fetch, to begin with the next.

An interleaved memory with n banks is said to be n-way interleaved. In an interleaved memory system, there are still two banks of DRAM but logically the system seems one bank of memory that is twice as large.

In the interleaved bank representation below with 2 memory banks, the first long word of bank 0 is floowed by that of bank 1, which is followed by the second long word of bank 0, which is followed by the second long word of bank 1 and so on.

The following figure shows the organization of two physical banks of n long words. All even long words of logical bank are located in physical bank 0 and all odd long words are located in physical bank 1.
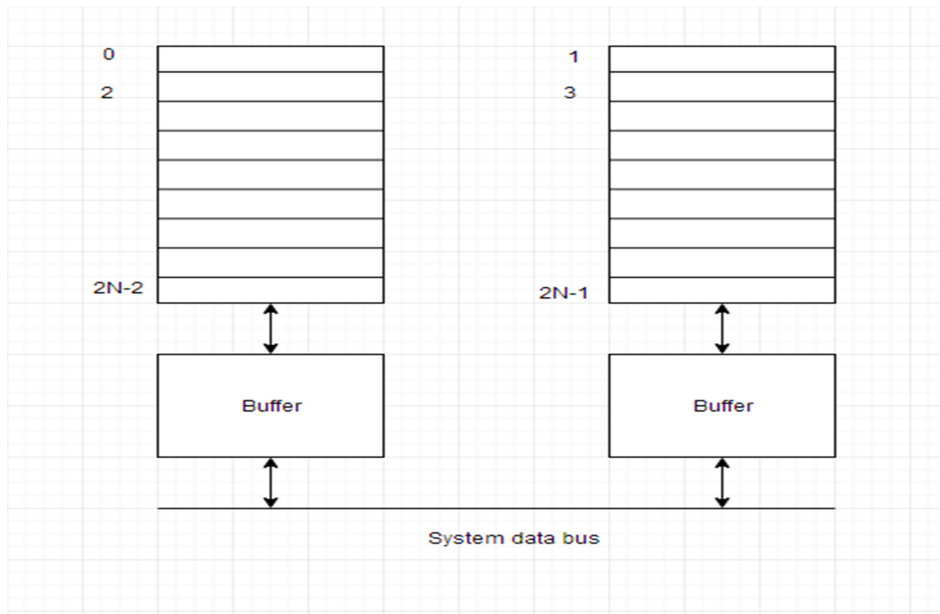
Types:
There are two methods for interleaving a memory:

2-Way Interleaved

Two memory blocks are accessed at same time for writing and reading operations.

4-Way Interleaved

Four memory blocks are accessed at the same time.

4) **Why cache memory is used .Explain mapping process of main memory to cache memory (7/2015) (7/2017)(2019(s)-6)**

Ans- A CPU **cache** is a hardware **cache used** by the central processing unit (CPU) of a computer to reduce the average cost (time or energy) to access data from the main **memory**. A **cache** is a smaller, faster **memory**, closer to a processor core, which stores copies of the data from frequently **used** main **memory** locations.

Memory cache configurations

Caching configurations continue to evolve, but memory cache traditionally works under three different configurations:

- **Direct mapping**, in which each block is mapped to exactly one cache location. Conceptually, this is like rows in a table with three columns: the data block or cache line that contains the actual data fetched and stored, a tag that contains all

or part of the address of the fetched data, and a <u>flag</u> <u>bit</u> that connotes the presence of a valid bit of data in the row entry.

- **Fully associative mapping** is similar to direct mapping in structure, but allows a block to be mapped to any cache location rather than to a pre-specified cache location (as is the case with direct mapping).

   Set associative mapping can be viewed as a compromise between direct mapping and fully associative mapping in which each block is mapped to a subset of cache locations. It is sometimes called *N-way set associative mapping*, which provides for a location in main memory to be cached to any of "N" locations in the L1 cache.

5) **What is Hit ratio. (2/2017)**
   Ans- The *hit ratio* is the fraction of accesses which are a *hit*. The miss *ratio* is the fraction of accesses which are a miss. It holds that. miss rate = 1 − *hit* rate . The (*hit*/miss) latency (AKA access time) is the time it takes to fetch the data in case of a *hit*/miss.

6) **Define Virtual memory . How memory table is used for mapping a virtual address.**(7/2015) (3.5/2016) (7/2017) (2019(s)-1-viii)

Ans- Virtual memory is a common part of most <u>operating systems</u> on <u>desktop computers</u>. It has become so common because it provides a big benefit for users at a very low cost.

In this article, you will learn exactly what virtual memory is, what your computer uses it for and how to configure it on your own machine to achieve optimal performance.

A **page table** is the <u>data structure</u> used by a <u>virtual memory</u> system in a <u>computer</u> <u>operating system</u> to store the mapping between <u>virtual addresses</u> and <u>physical addresses</u>. Virtual addresses are used by the program executed by the accessing <u>process</u>, while physical addresses are used by the hardware, or more specifically, by the <u>RAM</u> subsystem.

In operating systems that use virtual memory, every process is given the impression that it is working with large, contiguous sections of memory. Physically, the memory of each process may be dispersed across different areas of physical memory, or may have been moved (<u>paged out</u>) to another storage, typically to a hard disk drive.

When a process requests access to data in its memory, it is the responsibility of the operating system to map the virtual address provided by the process to the physical address of the actual memory where that data is stored. The page table is where the operating system stores its mappings of virtual addresses to physical addresses, with each mapping also known as a *page table*

The translation process
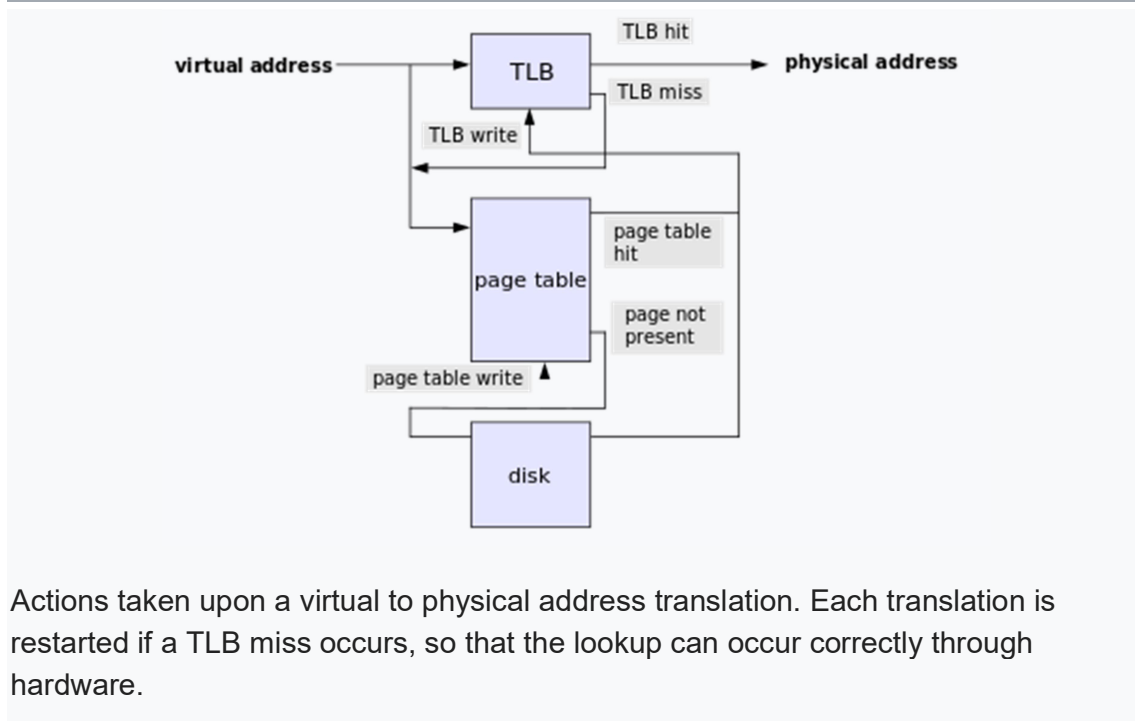
Actions taken upon a virtual to physical address translation. Each translation is restarted if a TLB miss occurs, so that the lookup can occur correctly through hardware.

The CPU's memory management unit (MMU) stores a cache of recently used mappings from the operating system's page table. This is called the translation lookaside buffer (TLB), which is an associative cache.

When a virtual address needs to be translated into a physical address, the TLB is searched first. If a match is found (a *TLB hit*), the physical address is returned and memory access can continue. However, if there is no match (called a *TLB miss*), the memory management unit, or the operating system TLB miss handler, will typically look up the address mapping in the page table to see whether a mapping exists (a *page walk*). If one exists, it is written back to the TLB (this must be done, as the hardware accesses memory through the TLB in a virtual memory system), and the faulting instruction is restarted (this may happen in parallel as well). This subsequent translation will find a TLB hit, and the memory access will continue.

7) **What are the different types of ROM. Explain them in details.**
   (5/2016,5/2017,2018(w)-6)

Ans- *ROM* stands for Read Only Memory. The memory from which we can only read but cannot write on it. This type of memory is non-volatile. The information is stored permanently in such memories during manufacture. A *ROM* stores such instructions that are required to start a computer.

Classification Of ROM
1. **Mask ROM –** In this type of ROM, the specification of the ROM (its contents and their location), is taken by the manufacturer from the customer in tabular form in a specified format and then makes corresponding masks for the paths to produce the desired output . This is costly, as the vendor charges special fee from the customer for making a particular ROM (recommended, only if large quantity of the same ROM is required).

Uses – They are used in network operating systems, server operating systems, storing of fonts for laser printers, sound data in electronic musical instruments .

2. **PROM –** It stands for Programmable Read-Only Memory . It is first prepared as blank memory, and then it is programmed to store the information . The difference between PROM and Mask ROM is that PROM is manufactured as blank memory and programmed after manufacturing, whereas a Mask ROM is programmed   during   the   manufacturing   process. To program the PROM, a PROM programmer or PROM burner is used . The process of programming the PROM is called as burning the PROM . Also, the data stored in it cannot be modified, so it is called as one – time programmable device.
   Uses – They have several different applications, including cell phones, video game consoles, RFID tags, medical devices, and other electronics.

3. **EPROM –** It stands for Erasable Programmable Read-Only Memory . It overcomes the disadvantage of PROM that once programmed, the fixed pattern is permanent and cannot be altered . If a bit pattern has been established, the PROM becomes unusable, if the bit pattern has to be changed .
   This problem has been overcome by the EPROM, as when the EPROM is placed under a special ultraviolet light for a length of time, the shortwave radiation makes the EPROM return to its initial state, which then can be programmed accordingly . Again for erasing the content, PROM programmer or PROM burner is used.

   Uses – Before the advent of EEPROMs, some micro-controllers, like some versions of Intel 8048, the Freescale 68HC11 used EPROM to store their program .

4. **EEPROM –** It stands for Electrically Erasable Programmable Read-Only Memory . It is similar to EPROM, except that in this, the EEPROM is returned to its initial state by application of an electrical signal, in place of ultraviolet light . Thus, it provides the ease of erasing, as this can be done, even if the memory is positioned in the computer. It erases or writes one byte of data at a time .

8) **Why cache memory is needed . What do you mean by mapping process in cache memory . Explain the type of mapping procedures.   (7/2016),2018-(s)-2-vii**
   Ans- Cache memory is used to increase the performance of the PC. It holds data and instructions retrieved from RAM to provide faster access to the CPU. ... It holds frequently requested data and instructions so that they are immediately available to the CPU when **needed**.

   Processor cache is fast memory that the processor can access as quickly as possible. The type of memory used in the processor cache is on the same die as the processor and far more expensive to produce than normal RAM. Processor cache is substantially faster than the system RAM and contains information that the processor will be immediately and repeatedly accessing. The

faster memory runs on at a refresh rate that's closer to the CPU's clock speed, which minimizes wasted cycles .

CHAP-6 ,   PROGRAMMED I/O

1) **Programmed I/O . (3.5/2017,2018-7)**

ans- Programmable I/O is one of the I/O technique other than the interrupt-driven I/O and direct memory access (DMA). The programmed I/O was the most simple type of I/O technique for the exchanges of data or any types of communication between the processor and the external devices. With programmed I/O, data are exchanged between the processor and the I/O module. The processor executes a program that gives it direct control of the I/O operation, including sensing device status, sending a read or write command, and transferring the data. When the processor issues a command to the I/O module, it must wait until the I/O operation is complete. If the processor is faster than the I/O module, this is wasteful of processor time. The overall operation of the programmed I/O can be summaries as follow:

1. *The processor is executing a program and encounters an instruction relating to I/O operation.*
2. *The processor then executes that instruction by issuing a command to the appropriate I/O module.*
3. *The I/O module will perform the requested action based on the I/O command issued by the processor (READ/WRITE) and set the appropriate bits in the I/O status register.*
4. *The processor will periodically check the status of the I/O module until it find that the operation is complete.*

Programmed I/O Mode Input Data Transfer
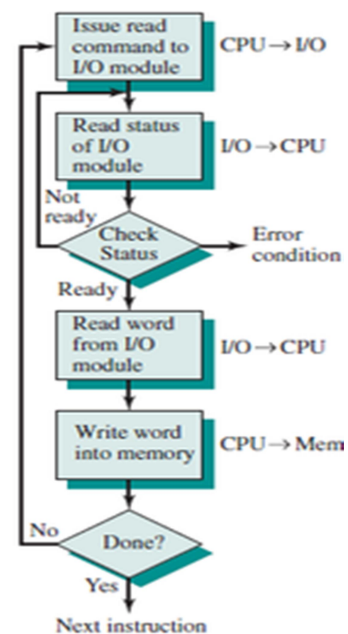
2)What is DMA.Explain DMA method of data transfer by switable diagram.   (7/2017),(2019(s)-5)

Ans- DMA is a method of transferring data from the computer's RAM to another part of the computer without processing it using the CPU.

DMA Function

• DMA module on system bus used to mimic the processor.

• DMA module only uses system bus when processor does not need it.

• DMA module may temporarily force processor to suspend
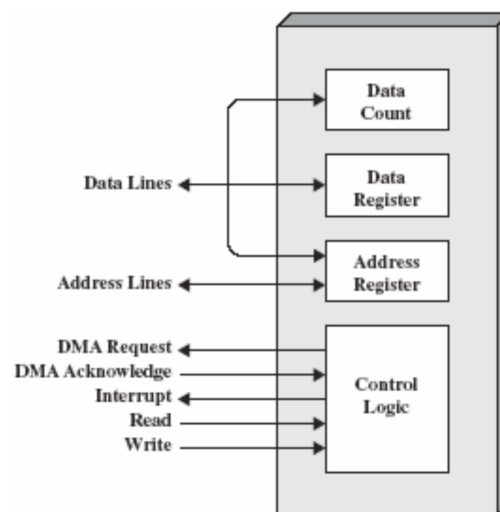
operations

DMA Operation

• The processor issues a command to DMA module o Read or write o I/O device address using data lines o Starting memory address using data lines – stored in address register o Number of words to be transferred using data lines – stored in data register

• The processor then continues with other work

• DMA module transfers the entire block of data – one word at a time – directly to or from memory without going through the processor

• DMA module sends an interrupt to the processor when complete

DMA Configurations
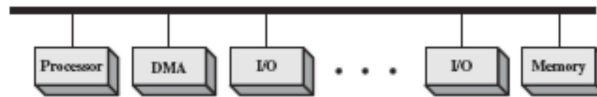
- **Single bus** – detached DMA module
  - Each transfer uses bus twice – I/O to DMA, DMA to memory
  - Processor suspended twice
- Single bus – integrated DMA module
- Module may support more than one device
- Each transfer uses bus once – DMA to memory
- Processor suspended once

2) **Identify the different modes of data transfer to and from peripherals .Explain I/O channel architecture. (5/2015)(2/2017)**

Ans- Data transfer to and from peripherals may be handled in one of three possible modes:

 A. Programmed I/O

 B. Interrupt-initiated I/O

C. Direct memory access (DMA)

 I/O Channels and Processors

The Evolution of the I/O Function

1. Processor directly controls peripheral device

2. Addition of a controller or I/O module – programmed I/O

 3. Same as 2 – interrupts added

 4. I/O module direct access to memory using DMA

 5. I/O module enhanced to become processor like – I/O channel

6. I/O module has local memory of its own – computer like – I/O processor

• More and more the I/O function is performed without processor involvement

. • The processor is increasingly relieved of I/O related tasks – improved performance.

 Characteristics of I/O Channels

• Extension of the DMA concept

• Ability to execute I/O instructions – special-purpose processor on I/O channel – complete control over I/O operations

• Processor does not execute I/O instructions itself – processor initiates I/O transfer by instructing the I/O channel to execute a program in memory

• Program specifies

o Device or devices

o Area or areas of memory

o Priority o Error condition actions

Two type of I/O channels

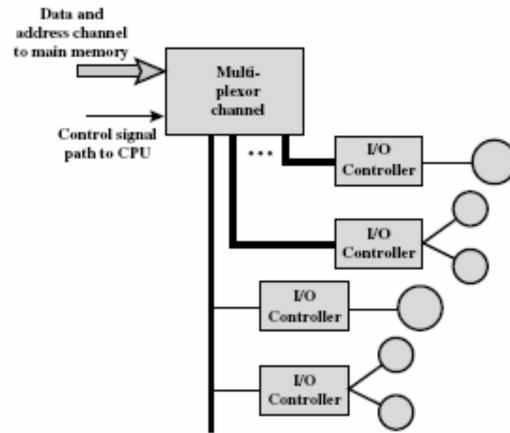**• Selector channel**
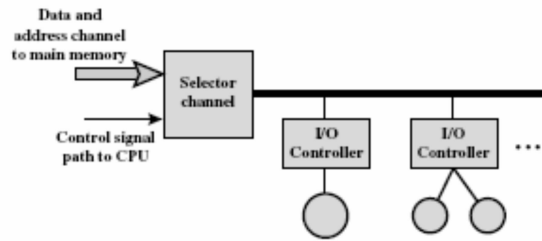
(a) Selector

o Controls multiple high-speed devices

o Dedicated to the transfer of data with one of the devices

o Each device handled by a controller, or I/O module

o I/O channel controls these I/O controllers

• Multiplexor channel

(b) Multiplexor

o Can handle multiple devices at the same time

o Byte multiplexor – used for low-speed devices

o Block multiplexor – interleaves blocks of data from several devices.

5)What is the function of I/O Interfaces . (2/2015)(2019(s)-1-vii)

Ans- The computer system's I/O architecture is its interface to the outside world. This architecture is designed to provide a systematic means of controlling interaction with the outside world and to provide the operating system with the information it needs to manage I/O activity effectively.

Two important examples of external I/O interfaces are FireWire and Infiniband.

Type of Interfaces

Parallel interface – multiple bits transferred simultaneously
 Serial interface – bits transferred one at a time I/O module dialog for a write operation
1. I/O module sends control signal – requesting permission to send data
2. Peripheral acknowledges the request
3. I/O module transfer data
4. Peripheral acknowledges receipt of data

CHAP-7   I/O   INTERFACE AND BUS ARCHITECTURE

1) Define USB.   (2/2017)

Ans- A Universal Serial Bus (*USB*) is a common interface that enables communication between devices and a host controller such as a personal computer (PC). It connects peripheral devices such as digital cameras, mice, keyboards, printers, scanners, media devices, external hard drives and flash drives.

2) Short notes on SCSI.   (3/2017)(2019(s)-1-ii)

Ans- Stands for "Small Computer System Interface," and is pronounced "scuzzy." SCSI is a computer interface used primarily for high-speed hard drives. This is because SCSI can support faster data transfer rates than the commonly used IDE storage interface. SCSI also supports daisy-chaining devices, which means several SCSI hard drives can be connected to single a SCSI interface, with little to no decrease in performance.

The different types of SCSI interfaces are listed below:

- **SCSI-1:** Uses an 8-bit bus, supports data transfer speeds of 4 MBps.

- **SCSI-2:** Uses a 50-pin connector instead of a 25-pin connector, and supports multiple devices. It is one of the most commonly used SCSI standards. Data transfer speeds are typically around 5 MBps.

- **Wide SCSI:** Uses a wider cable (168 cable lines to 68 pins) to support 16-bit data transfers.

- **Fast SCSI:** Uses an 8-bit bus, but doubles the clock rate to support data transfer speeds of 10 MBps.

- **Fast Wide SCSI:** Uses a 16-bit bus and supports data transfer speeds of 20 MBps.

- **Ultra SCSI:** Uses an 8-bit bus, supports data rates of 20 MBps.
- **SCSI-3:** Uses a 16-bit bus, supports data rates of 40 MBps. Also called Ultra Wide SCSI.
- **Ultra2 SCSI:** Uses an 8-bit bus, supports data transfer speeds of 40 MBps.
- **Wide Ultra2 SCSI:** Uses a 16-bit bus, supports data transfer speeds of 80 MBps.
- **Ultra3 SCSI:** Uses a 16-bit bus, supports data transfer rates of 160 MBps. Also known as Ultra-160.
- **Ultra-320 SCSI:** Uses a 16-bit bus, supports data transfer speeds of 320 MBps.
- **Ultra-640 SCSI:** Uses a 16-bit bus, supports data transfer speeds of 640 MBps.

While SCSI is still used for some high-performance equipment, newer interfaces have largely replaced SCSI in certain applications. For example, Firewire and USB 2.0 have become commonly used for connecting external hard drives. Serial ATA, or SATA, is now used as a fast interface for internal hard drives.

3) Peripheral component interface. (3.5/2016)

Ans- A Peripheral Component Interconnect Bus (PCI bus) connects the CPU and expansion boards such as modem cards, network cards and sound cards. These expansion boards are normally plugged into expansion slots on the motherboard.

The PCI local bus is the general standard for a PC expansion bus, having replaced the Video Electronics Standards Association (VESA) local bus and the Industry Standard Architecture (ISA) bus. PCI has largely been replaced by USB.

PCI requirements include:

- Bus timing
- Physical size (determined by the wiring and spacing of the circuit board)
- Electrical features
- Protocols

PCI specifications are standardized by the Peripheral Component Interconnect Special Interest Group.

Today, most PCs do not have expansion cards, but rather devices integrated into the motherboard. The PCI bus is still used for specific cards. However, for practical purposes, USB has replaced the PCI expansion card.

During system startup the operating system searches for all PCI buses to attain information about the resources needed for each device. The OS communicates with each device and assigns system resources, including memory, interrupt requests and allotted input/output (I/O) space.

CHAP-8 PARALLEL PROCESSING

1) Define Parallel Processing.   (2/2015)   (2/2016)

Ans-

Parallel processing is the method of evenly distributing computer processes between two or more computer processors. This requires a computer with two or more processors installed and enabled. It also requires an operating system capable of supporting two or more processors, and software programs capable of evenly distributing processes between them.

2) State and Explain Filynns classiofication.   (5/2017) (2019(s)-2-viii)

Ans- This classification was first studied and proposed by Michael Flynn in 1972. Flynn did not consider the machine architecture for classification of parallel computers; he introduced the concept of instruction and data streams for categorizing of computers. All the computers classified by Flynn are not parallel computers, but to grasp the concept of parallel computers, it is necessary to understand all types of Flynn's classification. Since, this classification is based on instruction and data streams, first we need to understand how the instruction cycle works.

Flynn's classification is based on multiplicity of instruction streams and data streams observed by the CPU during program execution. Let Is and Ds are minimum number of streams flowing at any point in the execution, then the computer organisation can be categorized as follows:

1) **Single Instruction and Single Data stream (SISD)**   In this organisation, sequential execution of instructions is performed by one CPU containing a single processing element (PE), i.e., ALU under one control unit as shown in Figure 4. Therefore, SISD machines are conventional serial computers that process only one stream of instructions and one stream of data.

2) **Single Instruction and Multiple Data stream (SIMD)**   In this organisation, multiple processing elements work under the control of a single control unit. It has one instruction and multiple data stream. All the processing elements of this organization receive the

same instruction broadcast from the CU. Main memory can also be divided into modules for generating multiple data streams acting as a distributed memory as shown in Figure 5. Therefore, all the processing elements simultaneously execute the same instruction and are said to be 'lock-stepped' together. Each processor takes the data from its own memory and hence it has on distinct data streams. (Some systems also provide a shared global memory for communications.) Every processor must be allowed to complete its instruction before the next instruction is taken for execution. Thus, the execution of instructions is synchronous. Examples of SIMD organisation are ILLIAC-IV, PEPE, BSP, STARAN, MPP, DAP and the Connection Machine (CM-1).

3) **Multiple Instruction and Single Data stream (MISD)** In this organization, multiple processing elements are organised under the control of multiple control units. Each control unit is handling one instruction stream and processed through its corresponding processing element. But each processing element is processing only a single data stream at a time. Therefore, for handling multiple instruction streams and single data stream, multiple control units and multiple processing elements are organised in this classification.

This classification is not popular in commercial machines as the concept of single data streams executing on multiple processors is rarely applied. But for the specialized applications, MISD organisation can be very helpful. For example, Real time computers need to be fault tolerant where several processors execute the same data for producing the redundant data. This is also known as N- version programming. All these redundant data are compared as results which should be same; otherwise faulty unit is replaced. Thus MISD machines can be applied to fault tolerant real time computers.

4) **Multiple Instruction and Multiple Data stream (MIMD)** In this organization, multiple processing elements and multiple control units are organized as in MISD. But the difference is that now in this organization multiple instruction streams operate on multiple data streams . Therefore, for handling multiple instruction streams, multiple control units and multiple processing elements are organized such that multiple processing elements are handling multiple data streams from the Main memory as shown in Figure 7. The processors work on their own data with their own instructions.

44

Tasks executed by different processors can start or finish at different times. They are not lock-stepped, as in SIMD computers, but run asynchronously. This classification actually recognizes the parallel computer. That means in the real sense MIMD organisation is said to be a Parallel computer. All multiprocessor systems fall under this classification. Examples include; C.mmp, Burroughs D825, Cray-2, S1, Cray X-MP, HEP, Pluribus, IBM 370/168 MP, Univac 1100/80, Tandem/16, IBM 3081/3084, C.m*, BBN Butterfly, Meiko Computing Surface (CS-1), FPS T/40000, iPSC.

5) What is Pipeline .Write an expression for speed up.. Draw a space time diagram to illustrate a 4-segment pipeline. (7/2015,5/2018) (2019(s)-3)

Ans- **Pipelining**

Pipelining is the process of accumulating instruction from the processor through a pipeline. It allows storing and executing instructions in an orderly process. It is also known as **pipeline processing**.

Pipelining is a technique where multiple instructions are overlapped during execution. Pipeline is divided into stages and these stages are connected with one another to form a pipe like structure. Instructions enter from one end and exit from another end.

Pipelining increases the overall instruction throughput.

In pipeline system, each segment consists of an input register followed by a combinational circuit. The register is used to hold data and combinational circuit performs operations on it. The output of combinational circuit is applied to the input register of the next segment.
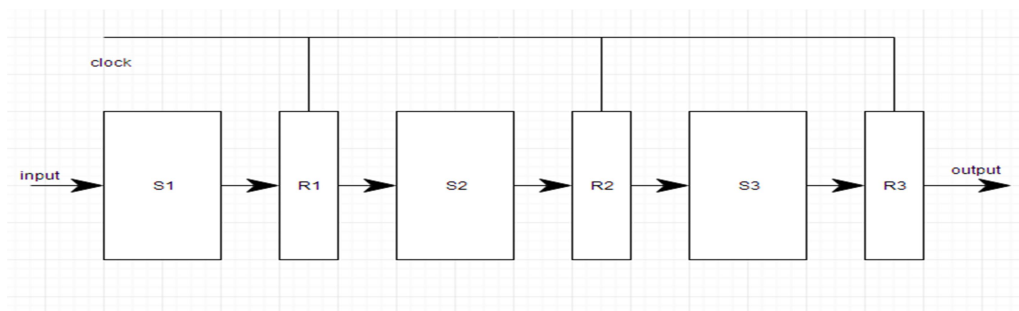
Pipeline system is like the modern day assembly line setup in factories. For example in a car manufacturing industry, huge assembly lines are setup and at each point, there are robotic arms to perform a certain task, and then the car moves on ahead to the next arm.

SPEED UP- In *computer architecture*, *speedup* is a number that measures the relative performance of two systems processing the same problem. ... The notion

of *speedup* was established by Amdahl's law, which was particularly focused on parallel processing.

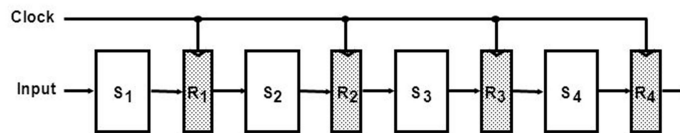*Speed up* of Machine B over Machine A can generally be defined as = Time(Machine A) / Time(Machine B);

Example: Weather simulation that does not complete in reasonable time

5)   Define instruction stream and data stream . (2/2016)

Ans- **Data streaming** is the transfer of **data** at a steady high-speed rate sufficient to support such applications as high-definition television (HDTV) or the continuous backup copying to a storage medium of the **data** flow within a computer.

instruction stream The sequence of *instructions*from memory to the control unit. "*instruction stream*." "*instruction stream*."

7) Write the Expression for speed up in pipelining. (2/2016)

Ans- **SPEED UP-** In *computer architecture*, *speedup* is a number that measures the relative performance of two systems processing the same problem. ... The notion of *speedup* was established by Amdahl's law, which was particularly focused on parallel processing.

Speed up of Machine B over Machine A can generally be defined as = Time(Machine A) / Time(Machine B);

Example: Weather simulation that does not complete in reasonable time.