

1. INTRODUCTION

What is OS?

- OS is a system software which controls the automatic operation of a computer.
- OS has 3 measure functions
 1. It acts as resource manager.
 2. Is acts as a traffic manager.
 3. It acts as user interface.

1.1 Objectives/Purpose of Operating System

1. Program execution

- When the user wants to execute his program, the system must be able to load the program into memory and run it.
- The program must be able to end its execution either normally or abnormally.

2. I/O Operation

- A running may require I/O, this I.O may involve a file or an I/O device since a user program can or execute I/O operation directly, the OS must provide the means to do so.

3. File manipulation

- The O/S provides Its user the facility to read or write (r/w) files and create or delete (c/d) files.

4. Error detection

- The O/S constantly needs to be aware of possible error. Error may occur in the CPU, in memory hardware, in I/O devices or in user program. For each type of error, the OS should take the appropriate action to ensure and consistent computing.

5. Recourse Allocation

- When these are multiple users/jobs running at the same time resource must be allocated to each of them.
- Many different types of resources are managed by the OS.

6. Accounting

- Keeps track which user use how much and what kind of computer resources.

7. Protection

- In a multi user environment when several jobs are executed simultaneously the OS prevents the interface of one job with the other.

1.1 The Function of Operating System

- I. Memory management function
- II. Processor management function
- III. Device management function
- IV. File management function
- V. Security
- VI. Command interpretation

I.Memory management

- Keep track of resource(memory), what part of the memory is use and by when, which part of memory is not in use.
- If multi programming divides which process gets memory, when it get it and how much.
- Allocate the resource(memory) to the process
- Reclaim the resource(memory) from the memory when the process no longer needed or the process has been executed.

II.Processor Management function

- Keep track of the resource(processor).
- Multiprogramming, which and how much, this is called the process scheduler.
- Allocate the resource(processor) to a process by setting of the necessary hardware register.
- Deallocate the resource(processor) to a process when process terminate or exceeds allowed amount of uses.

III.Device management function

- Keep track of resource (Devices and channel controls) this is called I/O traffic control.
- Decide what is an efficient way to allocate the resource (Device). If it is to be shared, then decide who get it and how much he used to get.
- Allocate the resource (Devices) and initiate and I/O operation.
- Deallocate resource.

IV.Information /file management function

- Keep track of the resource(file) divide who get the resource.
- Allocate the resource(file).
- Ex- Create a file.
- Deallocate, resource(file). Ex- Choose the file

V.Security

- The security model of an OS protects the resource and information of a computer system against unauthorised access.

VI.Command interpretation

- This model of an OS takes care of user command directing the system resource to handle the request.

1.2 Evolution of Operating System

I. Batch Processing

- In batch processing same type of Jobs batch together and execute at a time.
- The OS was simple its major task was to transfer control from one job to next job.
- The job was submitted to the computer. At some later tome the output appear.

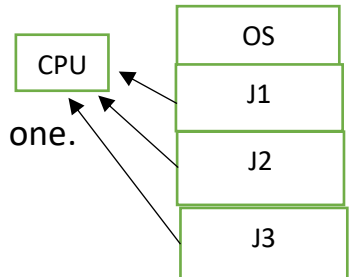
- The OS was always resident in memory

II. Multiprogramming

- This is technique to execute number of programs simultaneously by a single processor.
- A multiprogramming number of process resides in main memory at a time.
- This OS picks and begins to execute one of the jobs in the main memory.

Advantages: -

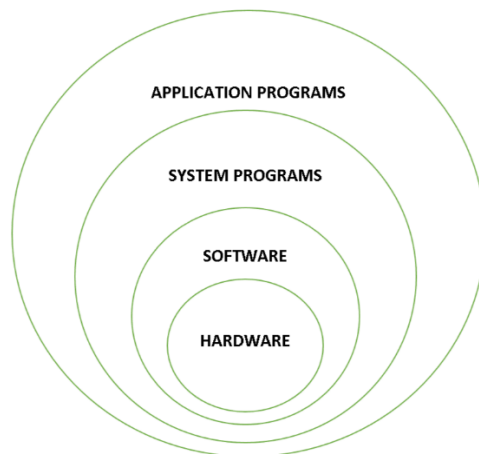
- Efficient memory utilization
- Through put increases
- CPU is never ideal, so performance increases



EX: - The main memory contains of 3 jobs, the CPU execute one by one.

1.3 structure of Operating System

The structure of OS consists of 4 layers



I. Hardware: - It is consisting of CPU, memory, I/O etc.

II. Software: - It is included process management routine, memory management routines, memory management routine, I/O, management routine.

III. System program: - This layer consists of compiler, linker, loader, assembler etc.

IV. Application Software: - This is depending on user need.

Ex: - Railway form, AutoCAD

Chapter-2

Instructions & Instruction Sequencing

Instruction: - An instruction is single operation of a processor defined by an instruction set architecture. An instruction may be any representation of an element of an executable program.

The instruction includes:

1. OP-code
2. Operand

OP-code: -

Specifying the operation to be perform such as add, sub, mul, load etc.

Operand: -

An instruction normally has one or more specifiers for operands (i.e., data) on which the operation should be act. Depending on architecture the operands may be register values, maybe values in the stack or the memory value etc.

Number of Operands: -

Instruction sets may be categorized by the maximum of operand explicitly specified in the instruction.

0(Zero) Operand: -

All arithmetic operation take place using the top one or two positions on the stack. One operand push and pop instruction are used to access to memory: Push a, push b, add pop c.

1(One)Operand: -

Most instruction specified a single write operation (e.g a register, a memory location and a constant) with the accumulative and left operand: load a, add b, store c.

2(Two) Operand: -

Many CISC (Complex Instruction Set Computer) and RISC (Reduced Instruction Set Computer) machine fall under these categories.

a) CISC

Load a, reg1; add reg1, b; store reg1, c

b) RISC

Reducing memory loads the instruction would be load a, reg1; load b, reg2; add reg1, reg2; store reg2, c

3(Three) Operand: -

Allowing better reuse of data.

CISC

It becomes either a single instruction add a, b, c or more typically: Move a, reg1, add reg1, b, c.

As most machines are limited two, two memory operands.

RISC

Due to large number of nits needed to encode the register. Arithmetic instruction uses registers only two operands load/store instructions are needed load a, reg1; load b, reg2; add reg1, reg2 → reg3, c

Instruction format: -

Instruction format is consisting of bits and bits appear in memory words. The bits of the instruction divided into groups called field.

Instruction format consists of 3 fields. The format of an instruction is usually in rectangular box.

- a) **Mode Field:** - It is a one-bit field, symbolized as I in format. It specifies the way the operand or, the effective address is determined.
- b) **Operation Code Field:** - It is 4 bits field. It specifies the processor operation to be performed such as add, subtraction, division, modulus etc.
- c) **Address Field:** - It is 11 bits field. It is designated to memory address or processor register.

15 14 11 0

Mode	OP	Address
Field	Code	Field

Address Instruction: -

Computer with 3 address instruction formats can use each address field to specify either a processor each address field to specify either a processor register or memory operand.

The 3-address instruction can be represented symbolically as ADD (operation to be performed) Z x Y (designation) source operand

Its meaning ADD the content of memory location x and y and place the result in location z i.e., $z \leftarrow x + y$

EX: $X = (A+B) * (C+D)$

ADD R1, A, B $R1 \leftarrow M[A] + M[B]$

ADD R2, C, D $R2 \leftarrow M[C] + M[D]$

MUL R1, R2 $M[X] = R1 + R2$

It has a2 processor register R1 and R2. M[A] is the operand of memory address symbolized by A.

Address Instruction: -

Two address instruction are common in commercial computer each address field can specify either a processor register or a memory word.

EX $X = (A+B) * (C+D)$

MOV R1 A $R1 \leftarrow M[A]$
 ADD R1 B $R1 \leftarrow R1 + M[B]$
 MOV R2 C $R2 \leftarrow M[C]$
 ADD R2 D $R2 \leftarrow R2 + M[D]$
 MUL R1 R2 $R1 \leftarrow R1 * R2$
 MOV X R1 $M[X] \leftarrow R1$

The MOV instruction move or transfer the operands to and from memory 4 processor register.

One Address Instruction: - One address instruction use an implied accumulator [AC] register or all data manipulation.

One address instruction has the command

ADD A

(On specified operand as assumed to be stored in fixed location, commonly in a processor register called the accumulator.)

Meaning: - Add the content of memory location A to the content of accumulator register and place the sum back into the accumulator.

$[AC] \leftarrow [AC] + R1$

$AC \leftarrow AC + R$

Load: - The load instruction copies of content of memory location A into the accumulator.

Store: - Store instruction copies the content of accumulation into the memory location A.

$EX = (A+B) * (C+D)$

Load A $AC \leftarrow M[A]$
 ADD B $AC \leftarrow AC + M[B]$
 Store T $M[T] \leftarrow AC$
 Load C $AC \leftarrow M[C]$
 ADD D $AC \leftarrow AC + M[D]$
 MUL T $AC \leftarrow AC * M[T]$
 Store X $M[X] \leftarrow AC$

T is temporary memory location for storing intermediate result.

Zero (0) Address Instruction: -

The instruction does not require address field for the instruction ADD and MUL. The push, pop instruction needs an address field to specify the operand.

EX $X = (A+B) * (C+D)$

PUSH A	$TOS \leftarrow A$
PUSH B	$TOS \leftarrow B$
ADD	$TOS \leftarrow A+B$
PUSH C	$TOS \leftarrow C$
PUSH D	$TOS \leftarrow D$
ADD	$TOS \leftarrow C+D$
MUL	$TOS \leftarrow (C+D) * (A+B)$
POP	$M[X] \leftarrow TOS$

TOS means 'TOP of the stack'.

Addressing Modes: -

The operation field of an instruction specifies the operation to be performed. The way the operands are chosen during program execution is dependent on the addressing mode of the instruction. The mode field is used to locate the operands need for operation.

Register Mode: -

In this mode the operand is specified indirectly simple in the definition of the instruction. Instructions of this type are 1- byte instruction. The instruction "complement accumulator". It is an implied mode instruction because the operand in the accumulator register is implied in the definition of the instruction. All register reference instruction that uses an accumulator are implied mode instruction.

Immediate Mode: -

An immediate mode instruction has an operand field rather than an address field. The operand field contains the actual operand field to be used in conjunction the actual operation specified in the instruction. Immediate mode instructions are useful for initializing registers to a constant value.

Direct Address Mode: -

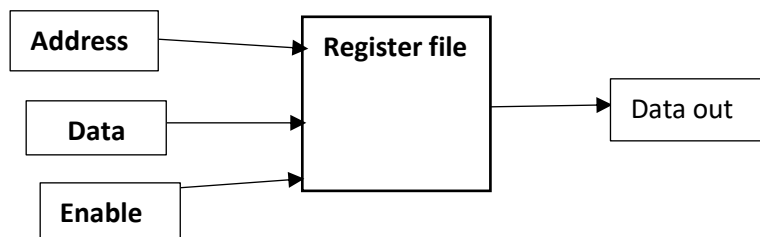
In this mode the effective address is equal to the address part of the instruction. The operand resides in memory and its address is given directly by the address field of the instruction.

PROCESSOR SYSTEM

Registers and temporary storage location inside the CPU that hold data & addresses. The register file is the component that contains all the general-purpose registers of the microprocessor. A few CPUs, also place special registers such as the PC & the status register file. Other CPUs keep them separate.

Register file: -

A simple register file is a set of registers & a decoder. The register file requires an address & a data input.



Consider the following equation.

$$C=A+B$$

To perform this operation. Read two values from the register file, A&B. One result to write back to the register file. When the operation has completed. For cases we do not want to write any value to the register file, we add a control signal called Read/write. When the control signal is high, the data is written to a register & when the control signal is low, no new values are written.

Basic memory operation.

All data & instruction are stored in the memory before & after they are used. These data should be transferred back & forth using read & write operation in memory.

Read operation.

The information to be fetched from memory may represent an instruction. The processor has to specify the address of memory location where this information is stored & request a 'Read' operation. The processor transfers the address to the Memory Address Register (MAR). The requested data are received from the memory & are stored in the Memory Data Register (MDR). They can be transferred to other registers in the processor.

- A read control signal is activated a MAR is loaded from the address lines & this will send a read command 'MR' on the bus. The data received are loaded into the MDR at the end of the clock cycle in which the Memory Clock (MFC) signal is received.

- The memory read operation can be done in these steps with the signal being activated as follows-
 - i. R_{3out} , MAR_{in} , Read
 - ii. MDR_{in} , WMFC
 - iii. MDR_{out} , R_{4in}

MFC – Memory function complete .
WMFC -Wait MFC

To execute an instruction , the control unit of the CPU generate the required control signal in proper sequence . there are two approaches used for generating the control signal in proper sequence .

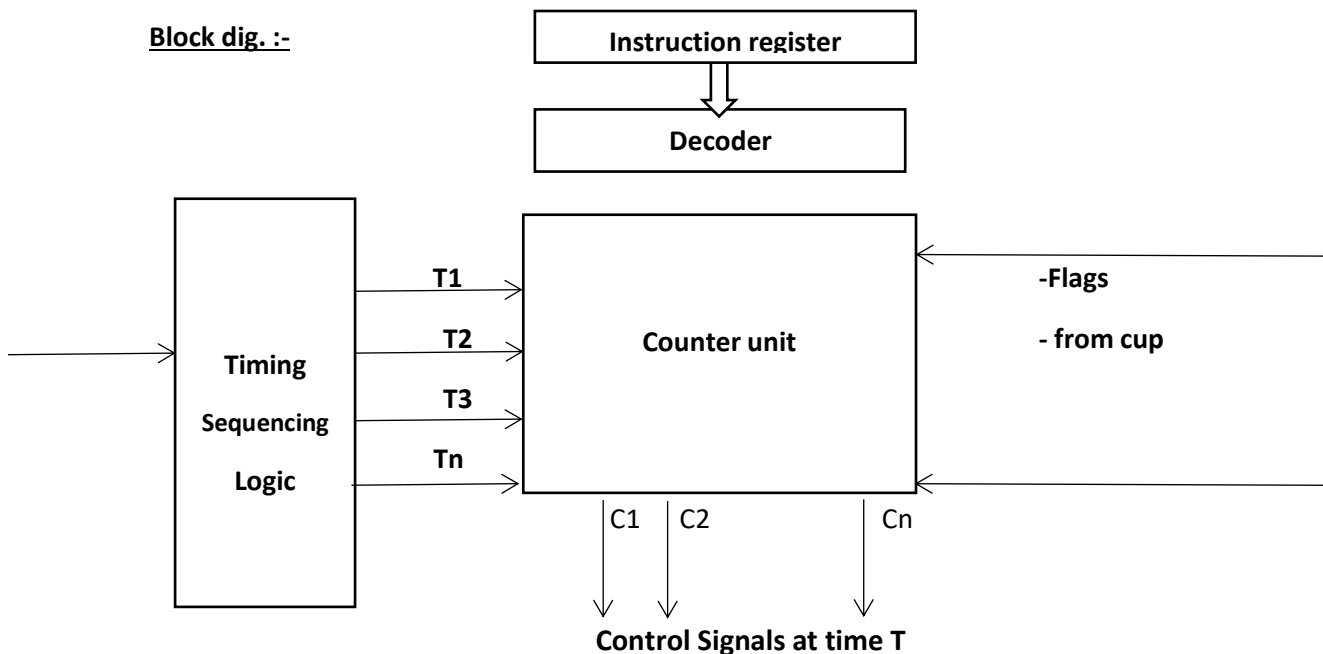
1. Hardware control unit
2. Microprogrammed control unit .

1. Hardware control unit :-

Hardware control can be defined as sequential logic circuit that generates specific sequences of control signal in response to externally supplied instructions.

- Hardware control consist of two decoders ,a sequence counter & a number of logical gates.

Block dig. :-



1. I use flags , decoder , logic gates & other digital circuits .
2. As name implies it is a hardware control unit
3. On the basis of input signal output is generated .

4. Difficult to design, test & implement .
5. Inflexible to modify.
6. Faster mode of operation .
7. Expensive & high error .
8. Used in RISC processor .

Advantages of hardware control unit:

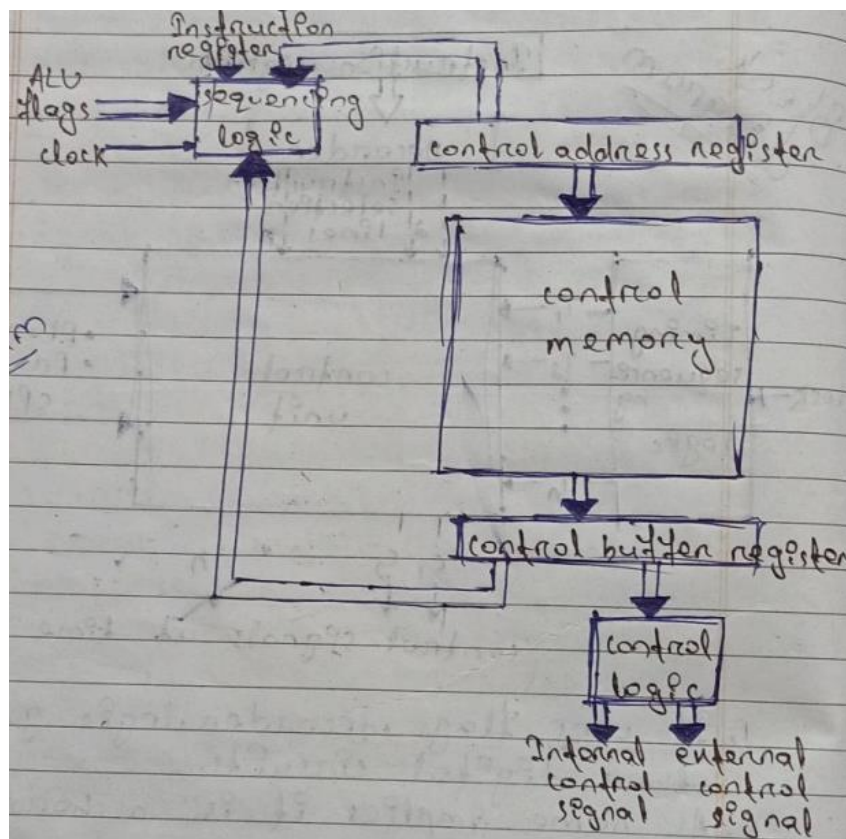
- i. Faster than microprogrammed control unit .
- ii. Can be optimized to produce fast mode of operation .

Disadvantages of hardware control unit:

- i. Instruction set control logic are directly .
- ii. Require change in wiring of designed has to be controlled.

2. Micro-programmed control unit

- A programmed control unit is built around a storage unit called a control store where all the control signals are stored in a program like format.



1. I use sequence of microinstruction in micro programming language
2. It is mid-way between hardware & software .
3. It general asset of control signal on the basis of control line.

4. Easy to design ,text & implement.
5. Flexible to modify .
6. Slower mode of operation
7. Cheaper & less error .
8. Used in CISC processor .

Advantages of microprogrammed control unit:

- i. Simplifies design of CU.
- ii. Cheaper.
- iii. Less error prone to implement.
- iv.

Disadvantages of microprogrammed control unit:

- i. Slower compared to hardwired control unit.

Memory system

Characteristics of storage.

- Storage technology at all levels of the storage hierarchy can be differentiating by evaluating certain core characteristics.
- These core characteristics are volatility mutability, accessibility & addressability the characteristics worth measuring are capacity & performance.

Volatility.

Non-volatile memory:- Will retain the stored information even if it is not constantly supplied with electric power. It is suitable long-term storage of information.

Volatile memory:- Requires constant power to maintain the stored information. The faster memory technologies of today. Primary storage is required to be very fast.

Differentiation.

Dynamic random access memory:- A form of volatile memory while also requires the stored information to be periodically re-read & re-written or refreshed, otherwise it would vanish.

Static memory:- A form of volatile memory similar to DRAM with the expectation that it never needs to be refreshed as long as power is applied (It loses its content if power is removed).

Mutability.

Read write storage or mutable storage:- Allows information to be overwritten at any time.

Read only storage:- Retains the information stored at the time of manufacture and write once storage (write once read many). These are called immutable storage. Immutable storage is used for off-line storage. Example includes CD-ROM & CO-R.

Slow write, fast read storage:- Read/Write storage allows information to be overwritten multiple times. Write operation must be slower than the read operation. CD-RW & flash memory.

Accessibility:-

Random access:- Any location in storage can be accessed at any moment in such a characteristic is well suited for primary & secondary storage.

Sequential access:- The accessing of pieces of information will be in a serial order. One after the other, such a characteristic is typical of off-line storage.

Addressability:-

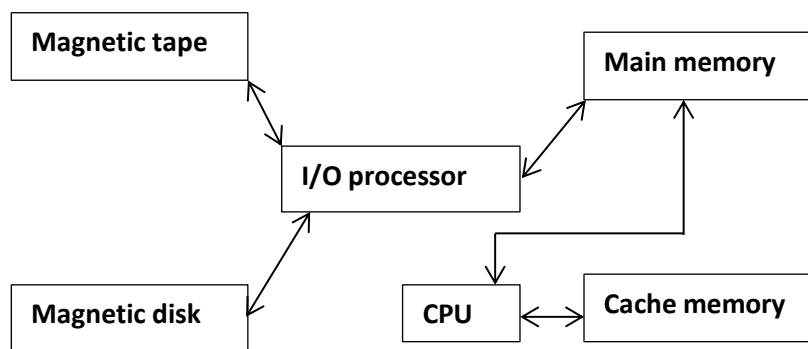
Location addressable:- In modern computers location addressable storage usually limits to primary storage accessed internally by computer programs.

File addressable:-Information is divided into files of variables length & a particular file is selected with human readable directory and file names.

Capacity:-

Law capacity:- The total amount of stored information that a storage device or medium can hold. It is expressed as a quantity of bits or byte.

Memory hierarchy:- The memory hierarchy system consist of all storage devices employed in a computer system from the slow but high capacity auxiliary memory to relative faster main memory to an even smaller and faster cache memory access unite to the high speed processing.



At the bottom of hierarchy are the relatively slow magnetic tapes are used to removal files. Next are magnetic disk used as backup storage. The main memory communicates directly with the CPU & with auxiliary memory devices through an I/O processor. When program not residing in main memory are needed by the CPU, they are brought in from auxiliary memory. Programs not currently needed transferred into auxiliary memory.

CACHE MEMORY:-

It is very high speed memory. Its stands between main memory and CPU. Cache holds the most heavily used data and program. It is increases the performance rate of the computer. Cache transfer the information between main memory and CPU. CPU directly access this memory.

Main memory:-

The main memory occupies a central position by being able to communicate directly with the CPU. When programs not residing in main memory are needed by CPU. They are brought in from auxiliary memory. Programs not needed memory CPU directly access the main memory.

Auxiliary memory :-

Auxiliary memory is slow and high capacity memory. This memory holds the data and program which is presently not used by CPU.

31. Access time in 1000 times of main memory. Transferring black size range in 266 to 2048 words. Magnetic taps relative slow used for removal files. Magnetic disk used as backup storage.

Semiconductor RAM memory:-

Semiconductor memory or RAM is used in all types of computers. It is also known as read/write memory. Semiconductor RAM use either a read cycle or a write cycle depending on the type of request. This memory is normally destructive & volatile memory. It is very fast memory & expensive. There are two main types of semiconductor RAM, S RAM, D RAM. The semiconductor RAM made up of chips. Each chip contains large number of cell. Each all has at least two stable stages that are 0 & 1.

S RAM:-

- Fast memory technology that requires power to hold its content.
- It used high speed register. Transistor.
- S RAM used as cache memory.
- Its access time is 10 to 30 Nano seconds range.

D-RAM:-

- D-RAM chips are very dense because-

32. They use only one transistor & one capacitor. The capacitor stresses the electrical pulse. Due to capacitor constantly counts one teak even when power is on. So it requires approximately 15 times per seconds refreshment is needed to maintain the change in the capacitor and retain the information.

- It is used as main memory.
- It's access time usually above 30 Nano seconds.

Read only memory:-

- Read only memory (ROM) is a class of storage media used in computers & other electronic devices. Data stored in RAM cannot be modified or can be modified only slowly or with difficulty.
- ROM refers only to mask ROM permanently stored in it and thus can never modified.

Type:-

- ROM chips are integrated circuits that physically encode the data to be stored & thus it is impossible to change their contents after fabrication
 - Programmable Read only memory (PROM) or one time programmed ROM (OTP) can be written to or programmed via a special device called a PROM programmer. Typically this device uses high voltage to permanently destroy. A PROM can only be programmed once.
- Erasable programmable read-only memory (F PROM) can be erased by exposure to strong ultra violet light, then rewritten with a Process that again needs higher than usual voltage applied. EPROM chip can be identified by the "window" which allows UV light enters after programming the window is typically covered with a label to prevent accidental erasure.

- Electrically erasable programmable read only memory, (EEPROM) is based similar semiconductor structure to EPROM, but allows its entire contents electrically erased then re-written electrically so that they need not be removed from the computer writing or Flashing an EEPROM is much slower.
- Flash memory type of be a modern type EEPROM. Flash memory erased & rewritten. Faster than ordinary EEPROM. Flash memory is sometimes called flash ROM or flash EPROM.

Cache memory:-

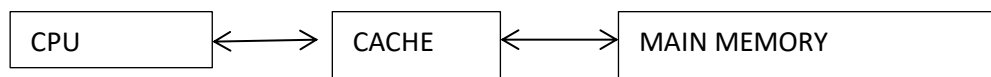
It is a special very high speed memory used to increase speed of processing. Its function is making current programs & data available to the CPU at a rapid rate it is logical placed between the cache & main memory. The cache is the fastest component of memory hierarchy. The performance of cache memory to measure by hit ratio.

Hit: when the by memory & finds hit ratio. CPU meters to the world in cache, it is said hit.

Miss: if the word is not found in cache, it is in the main memory, it is said miss.

Hit ratio:-

The ratio of the number of hits by total CPU reference to memory hits + misses is the hit ratio.



Mapping:-

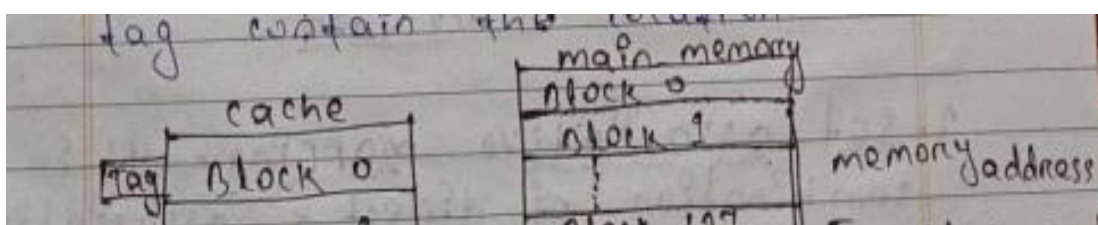
The transformation of data from main memory to cache memory is called as “mapping process”. e.g. - consider a cache consisting of 128 blocks of 116 words each, for a total of 2048 (2k) words & assume.

36. That the main memory has 64k words which is viewed as 4k blocks of 16 words each.

- Three types of mapping function are there
 1. Direct mapping
 2. Associative mapping
 3. Set associative mapping.

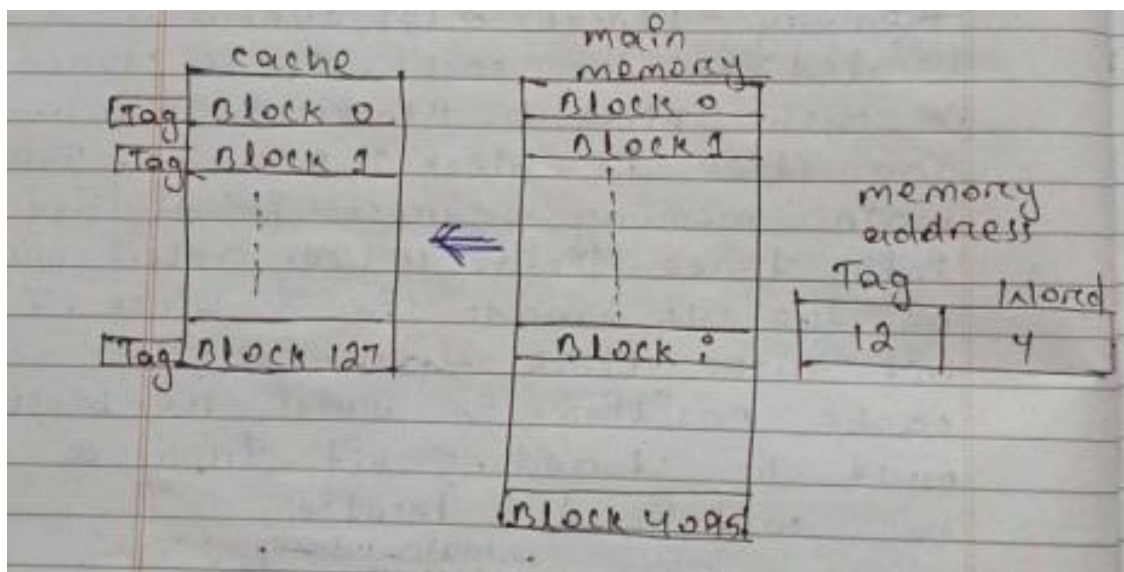
1. Direct mapping function :-

Simplest way to determine cache location. In this technique block J module 128 of the cache. Whenever one of the main memory blocks 0,128,256..... is located in the cache. It is stored in cache block 0. Blocks 1,129,257.... Are store in block 1 & 50 00. The main memory address is divided into three field. 4 bits select one of the 16 words in a block .7 cache position in which the block must be stored.5 bit form a tag contain the location.



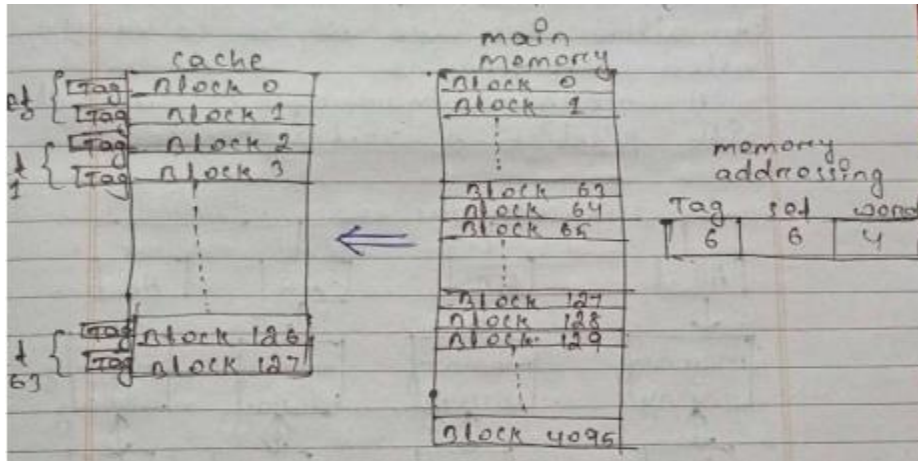
2. Associative mapping:-

It is a much more flexible method of mapping. In this case a main memory block can be placed into any cache block position. 12 tag bits are required to identify a memory block when it is resident in the cache. The tag bit of an address received from the CPU are compared to the tag bit of each block of the cache to see if present. This is called associative mapping technique;



3. Set associative mapping :-

It is the combination of direct & associative mapping techniques. In this technique a set consist of two blocks. Memory blocks 0,64,128....4032 map into cache set 0 & .they can occupy either of the two block position within this set. The tag field of the address must be compared to the tags of the two blocks of the set to check if the desired block is present.

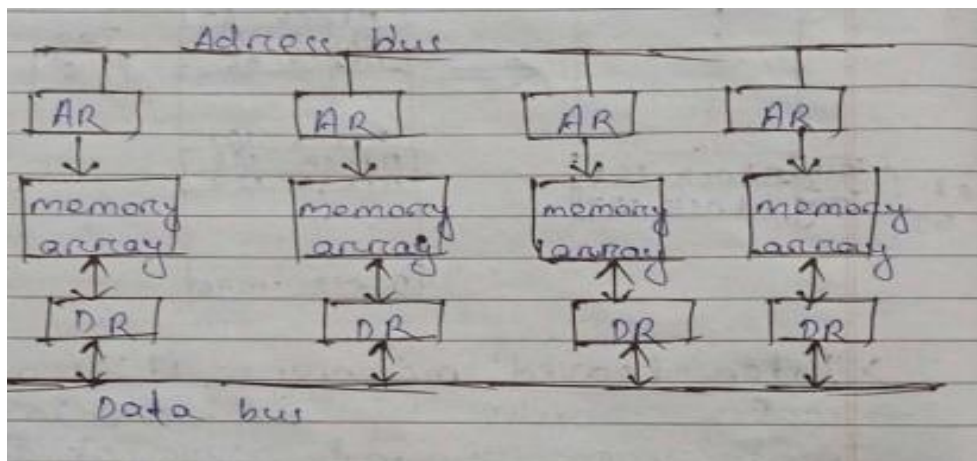


4. Inter leaved memory :-

-which works by partitioning the main memory into several independent memory module & distributing It allows the concurrent access to more than one module the interleaving of address of 'M' modules of memory is called 'M-way interleaving'.

A memory module is a memory array together with its own address & data register figure shows a memory unit four modules. Each module has its own AR & VR. AR receivers information from communicate with common bidirecti onal data bus the advantage of a module memory is that it allows the use of a technique called interleaving.

A modular memory is useful with pipeline & vector processor.



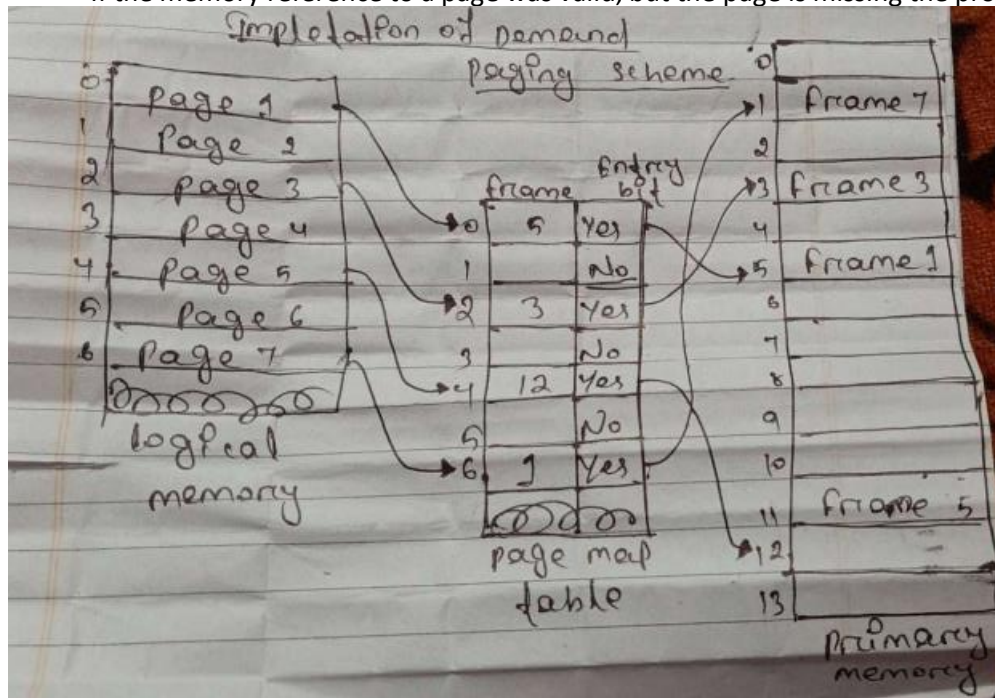
Array in the series of memory location or boxes

Demand Paging-

- In demand paging pages are loaded only on demand, not in advance.
- To implement paging it is necessary for the operating system to keep track of which Pages are currently in use ,The page map table contains an entry bit for each virtual page of the related process. Each page actually swapped in memory. Page map table points in actual location that contains the corresponding page frame & marked as No if a particular page is not in the memory.

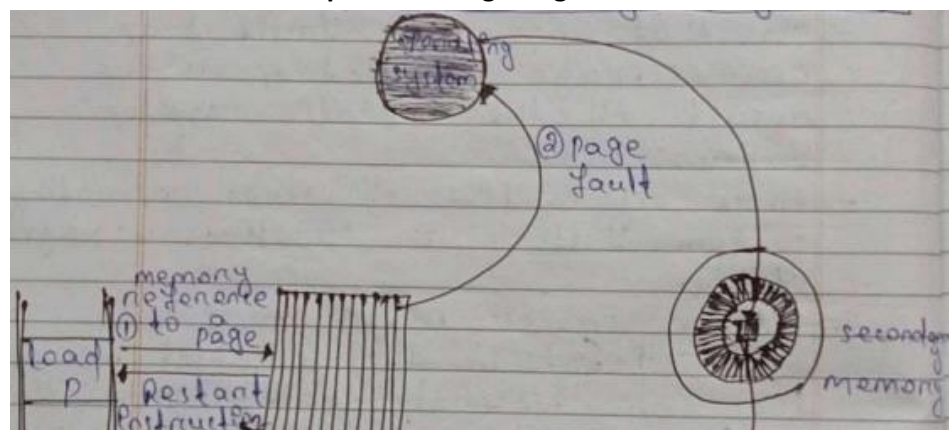
If the program tries to access a page that was not swapped in memory?

- In that case page fault trap occurs, it is the result of the operation system's failure.
- Here is a list of steps operating system follows in handling a page fault-
 - If a process refers to a page which is not in the physical memory then an internal table checked to verify whether a memory reference to a page was valid or invalid.
 - If the memory reference to a page was valid, but the page is missing the process



- By reading a disk, the desired page is brought back into the free memory location.
- Once the page is in the physical memory, the internal table kept with the process & page map table is updated to indicate that the page is now in memory.
- Restart the instruction that was interrupted due to the missing page.

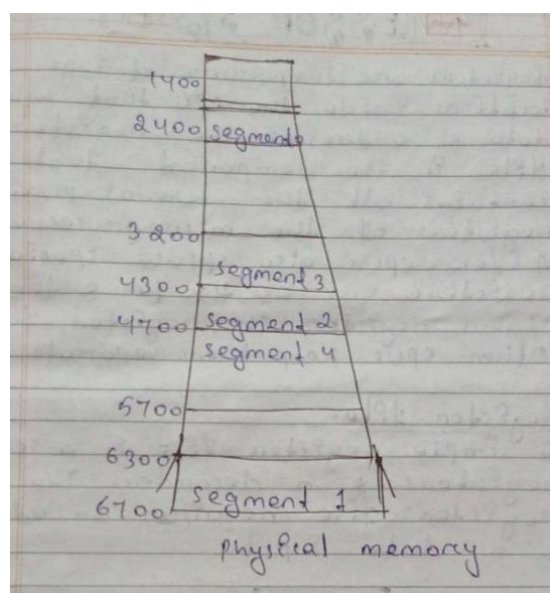
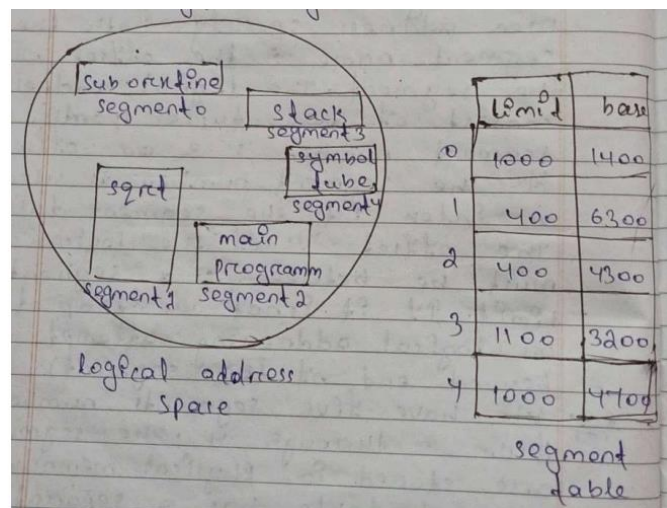
Steps in handling a Page Fault



Segmentation – It is a memory management scheme view of memory .Each segment has a name a length. User address specify both the segment name & the offset within the segment .The logical address consists of two parts a segment number 's' & an offset 'd'. The segment number is used as an index into the segment table. The offset 'd' of the logical address must be between 0 segment limit. If it is not trap to the a.s (logical addressing attempt beyond end of the segment) .

Example – we have five segment number from 0 through 4. The segments are stored in physical memory. The segment table has a separate entry foe each segment, giving the beginning address of the segment in the physical memory (or base) & the length of that segment (or limit).If it is not logical addressing attempt beyond end of segment.

Example - segment 2 is 400 byte long & begins at location 4300.A reference to segment 3 .byte 853 is mapped to $3200+853= 4053$.A reference to bye 1222 of segment 0 would results in a trap to the o,s as this segment is only 1000 bytes long.



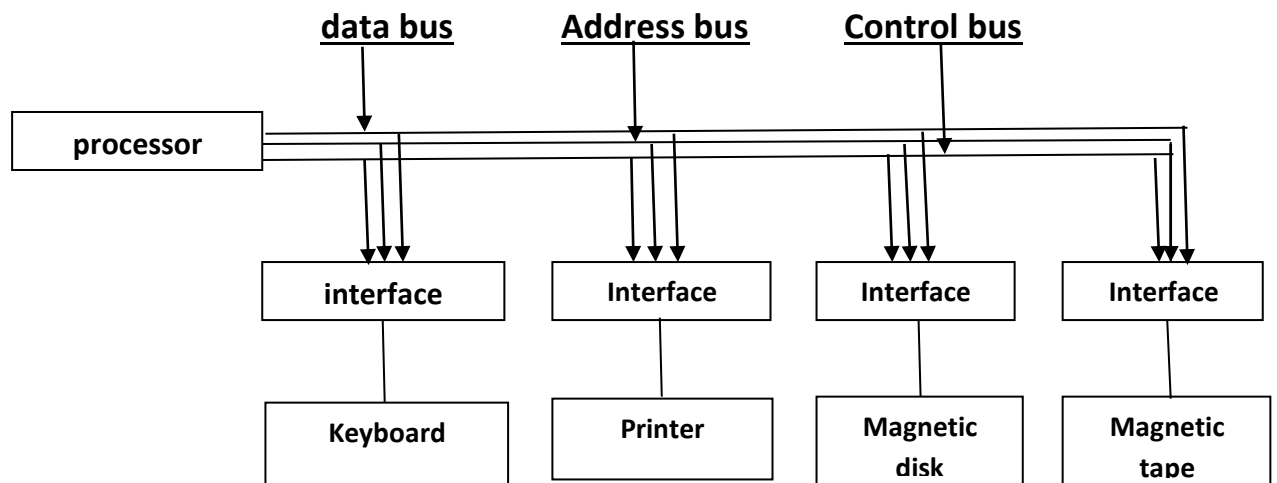
Unit-5

Input- Output :-

An interface is a physical connection between two devices. It provides method for transferring information between internal storage such as memory & CPU register & external i/o devices. Peripherals connected to a computer need special communication links for interfacing them with the CPU. The purpose of the communication link is to resolve the differences that exist between the central computer & each peripheral. The differences are:-

1. Peripherals are electro mechanical & electromagnetic devices & their manner of operation is different from the operation of the CPU & memory which are electronic devices. So a conversion of signal values may be required.
2. The i/o devices are normally slower than memory & CPU.
3. The operation of the peripherals must be synchronized (connected) with the operation of the CPU & memory.
4. Data formats & word length in peripheral different from the word length & data format in the CPU.
5. The operation of each peripheral must be controlled so as not to disturb the operation of the CPU & the other peripheral connected to the CPU.

i/o bus and interface:-



The i/o bus consists of data lines, address lines & control lines. Each peripheral device has associated with an interface. Each interface decodes the address & received from the i/o bus, interprets them for the peripherals & provides signals from the peripheral controller. It also synchronizes the data flow & supervises the transfer between peripheral & data. Each peripheral has its own controller that operates the particular electro mechanical device. The i/o bus from the processor attached to all peripheral interfaces to communicate to all peripheral interfaces to communicate with a particular device. The processor places a device address on the address line. When the interface detects its own address it activates the path between the bus line & the device that it controls. All peripherals whose address does not correspond to the address in the bus are disabled by their interface.

Modes of data transform or Transmission:-

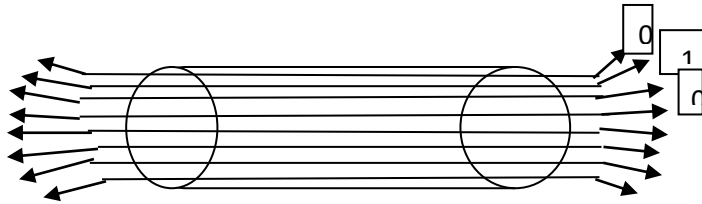
What is data transmission ?

It refers to movement of bits over some physical medium connecting two or more digital devices.

There are two option of transmission of bits. I.E-

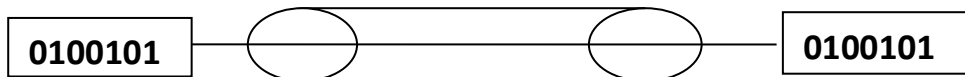
- Parallel transmission
- Serial Transmission

Parallel Transmission



When data transmitted a byte or a word through many wires with one wire carrying each bit this parallel transmission or communication.

1. Serial Transmission:-



Communication is in which the bit send one after another in a series. The measure differentials in serial data transfer is to detect the beginning of each new character in the bit Stean . If it is unable to active this it will not be able to interoperate the incoming bit steam correctly. Timing refer to how the receiving system knows the it receives the start of bits 4 the end of the bits.

Stream:- Moved in a continuous flow

Synchronised:- control

Synchronous:- Occurring at the same time

- These are two measure timing schemes are used.
 1. Asynchronous
 2. Synchronous

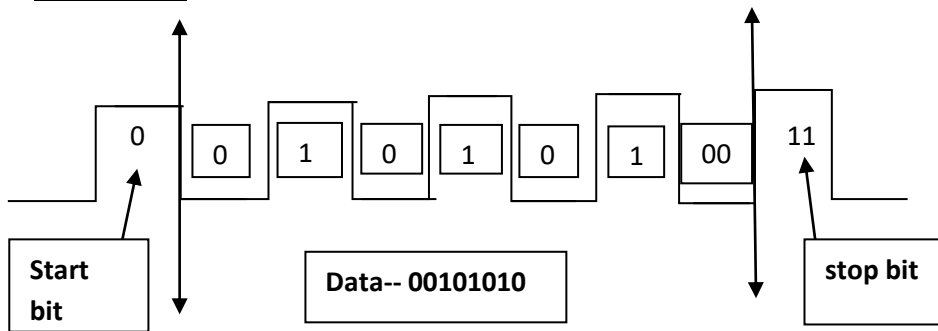
Asynchronous Transmission:-

1. It sends only one character at a time. Where a character is either a letter of the alphabets or numbers.
2. Each character has a start bit 4 ending each character has one or more stop bits.

This transmission is simple & expensive to implement. It is used mainly with serial ports i, e keyboard & mouse . It requires start & stop bits. For example: Every byte of data add one start & two stop bits. 11 bits are require to send 8 bits.

- Asynchronous transmission is normally only use for speed up to 3000 bit per second.

Diagram:-



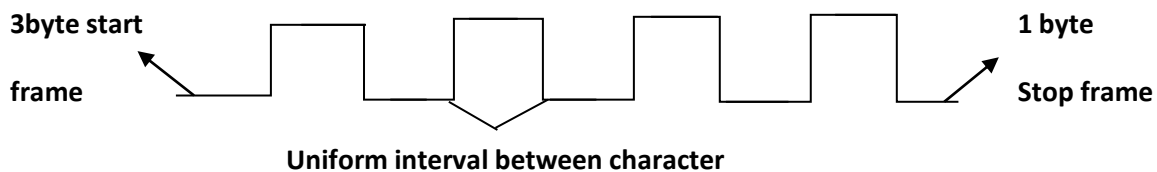
- When a character is not being sent, the state in-1.
- The start bit always 0.
- All bits follow the start bit.
- After the last bit of the character in transmitted the state will be 1 state.
- 1 or 2 bits are stop bits.
- Stop bit always 1.

2. Synchronous Transmission :

It sends one packet of character at a time. A start packet is used to tell receiving oration that a new packet of character is arriving & to synchronise, the receiving station internal clock proceeds each station. The packets also have end frames or packet to indicate the end of the packet. The packet can contain up to 6400 bits. Both starts frames have a special bit sequence that receiving station recognises to indicate the start & end of the packet. The start & end frames may be only 2 bite each. This transmission is more efficient us a little us only for frames (3 start & 1 stop) are required transmit up to 64 kilo bits

Synchronous transmission is more difficult & expensive

Diagram:-



In digital electronics both synchronous and Asynchronous transmission are the serial data transmission in which data transmitted between sender and receiver based on the clock pulie used for synchronization.

<u>Synchronous Transmission</u>	<u>Asynchronous Transmission</u>
<p><u>Deference:-</u>It in the type of transmission in Which a common clock pulse in shared between Transmitter and receiver to synchronized Communication.</p> <p><u>Data unit:-</u> Data sent in the form of frames or blocks so the data unite is frame/block.</p> <p><u>Performance:-</u> Due to common clock transmission is faster as compared to asynchrony.</p> <p><u>Cost:-</u> costlier</p> <p><u>Complexity:-</u> Easy to design</p>	<p><u>Deference:-</u> If it is the type of transmission in which the sender and receiver have their own external clocks.</p> <p><u>Data unit:-</u> Data sent in the form of bytes or characters so the data unite in byte/character</p> <p><u>Performance:-</u> Transmission is slower because sender, receiver has their input clocks.</p> <p><u>Cost:-</u>Cheaper</p> <p><u>Complexity:-</u> complex to design.</p>

In synchronous transmission data can be transmitted between two points in three different modes.

1. Simplex: It carries information in one direction only. Receiver cannot communicate with sender.

EX: Radio and TV.

I/O Operation :- An I/O operation is inputting data to a device from memory where CPU wrote it.

I/O operation can be done using three techniques.

1. Programmed I/O
2. interrupt drive I/O
3. direct memory access (DMA)

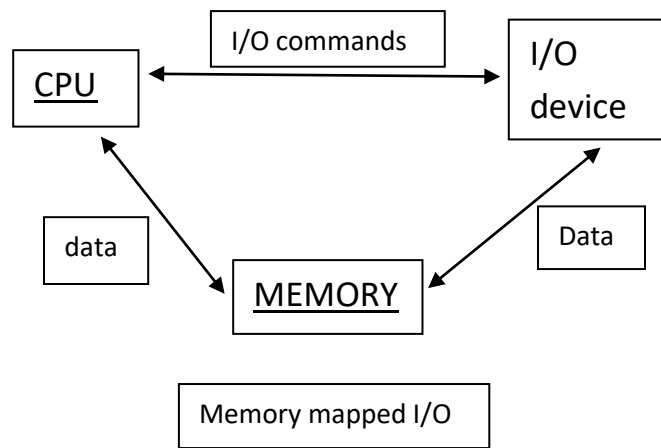
Programmed I/O:

In programmed I/O data exchanged between processor and I/O devices (peripherals). Each data transfer is involving CPU for every transaction.

The processor directly controls the I/O operation, sending a r/w command and transferring the data. When the processor issues a command to the I/O module, a lot of processor's time is wasted.

In programmed I/O method, the CPU stays in a program until the I/O unit indicates that it is ready for data transfer. This is a time-consuming process.

It can be avoided by using an interrupt facility.



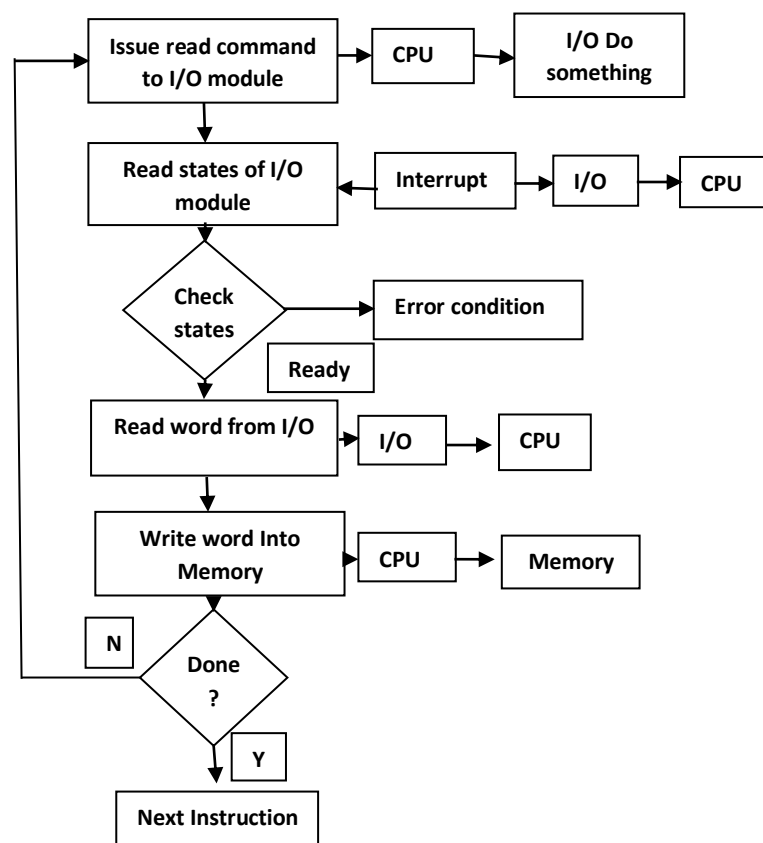
Interrupt driven I/O :

Interrupt driven I/O issues the I/O command to the I/O interface and then immediately .

Control of the CPU over to an other program while the I/O es being performed. This technique is used to overcame the limitation of programmed I/O.

Basic operation of interrupt driven I/O

- CPU issues read command.
- I/O module gets data from peripherals CPU does other work.
- I/O module interrupt CPU.
- CPU requests data.
- I/O model transfer data.



Advantage: 1. Fast

2. effect

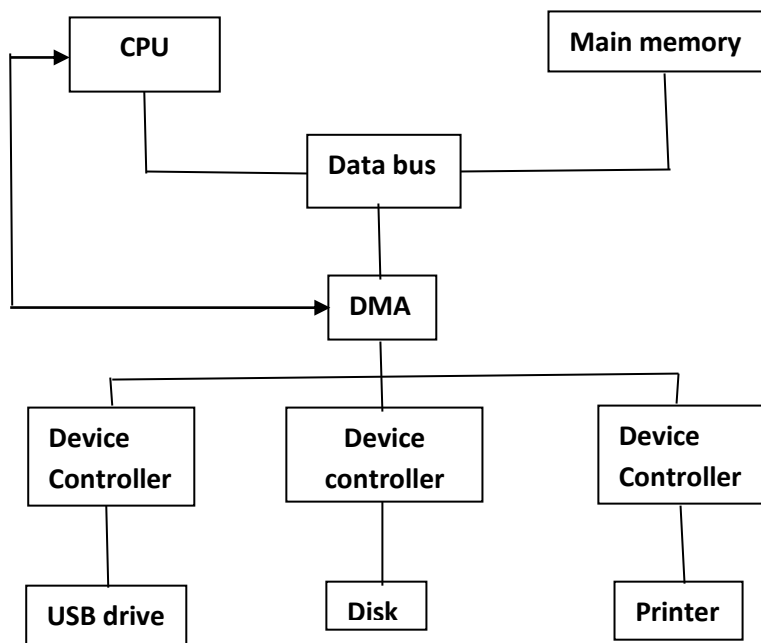
Disadvantage: Can be tricky and complicated

Programmed I/O	Interrupt driven I/O
<ol style="list-style-type: none">1. It is a time consuming process. It keeps processor busy needlessly and wastage of CPU time.2. It is treated as slow module.	<ol style="list-style-type: none">1. It overcomes the time consuming process of programmed I/O.2. It is faster than programmed I/O.

DMA(Direct Memory Access):-

DMA means CPU grants I/O module authority to read from or write to memory without involvement. DMA module itself controls exchange of data between main memory and the I/O devices. CPU is only involved at the beginning and end of the transfer and interrupted only after entire block transferred.

DMA needs special hardware called DMA controller that manages data transfer. The controller is programmed with where to read/write the data.



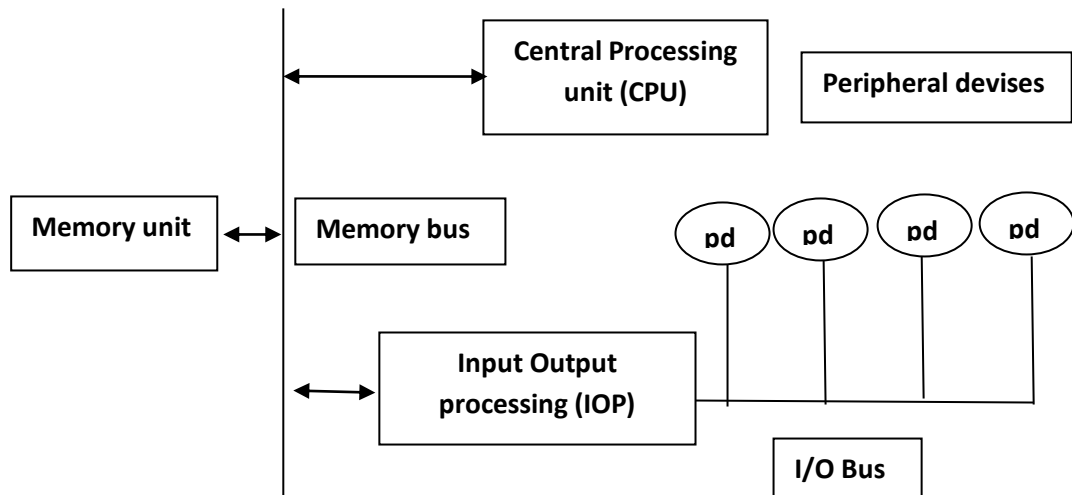
1. Device Driver is instructed to transfer disk data to a buffer address x.
2. Device driver then instructs disk controller to transfer data to buffer.
3. Disk controller states DMA transfer.
4. Disk controller sends each byte to DMA controller.

5. DMA controller transfers bytes to buffer.
6. DMA interrupts CPU to signal transfer completion.

IOP(Input/output processor)

Draw the functional block diagram of a commercially available input/output processor and explain.

- i. An input- output processor (IOP) may be classified as a processor with direct memory access capability that communicates with I/O devices. Each configuration may have two (or) more IOP's.
- ii. IOP'S take care of input and output tasks, relieving the CPU from the house keeping chores involved in I/O transfers.
- iii. A processor that communicates with remote terminals over telephone and other communication media in serial fashion is called a data communication processor(DCP).
- iv. IOP instructions are specifically designed to facilitate I/O translates. IOP can perform processing tasks like arithmetic, logic, branching and code translation.
- v. In the following diagram IOP provides a path for transfer of data between various peripheral devices and memory units.
- vi. CPU initiates the operation of IOP after that it will work independently.



- vii. After input data are assembled into memory words they are transferred from IOP.
 - a. The communication between IOP and devices connects to its similar to the program control method of transfer.
 - b. The communication with memory is similar to the DMA method.
 - c. The way in which CPU and IOP communicate depends on the levels of sophistication included in the system.

Unit – 6

I/o interface & Bus architecture

Bus interconnection:

Bus is a interconnection path way for communication between two or more device .Bus carrying signals signal representing binary '0' or '1'.Path way control of parallel lines. Ex: 8 bit data can b e transmitted over 8- lines,16 bits and so on . Buses provides path way between component at various level called system bus .

Bus structure :

Bus consist of 50 to 100 separate communication lines and each line is assigned a function , system .System bus classified into data, address and control lines.

Data bus (Refer unit – 1)

Address bus (do)

Control (do)



The control signal can be classified into following category:

1. Memory read / write
2. I/o read/ write
3. Transfer ACK
4. Bus request
5. Bus grouf
6. Interrupt request
7. Interrupt ACK
8. clock
9. reset

Multiple bus architecture:

When a bus is to loaded by no of devices , there will be a reduction in performance .Multiple bus architecture us two types.

1. Traditional bus architecture

2. High performance bus architecture

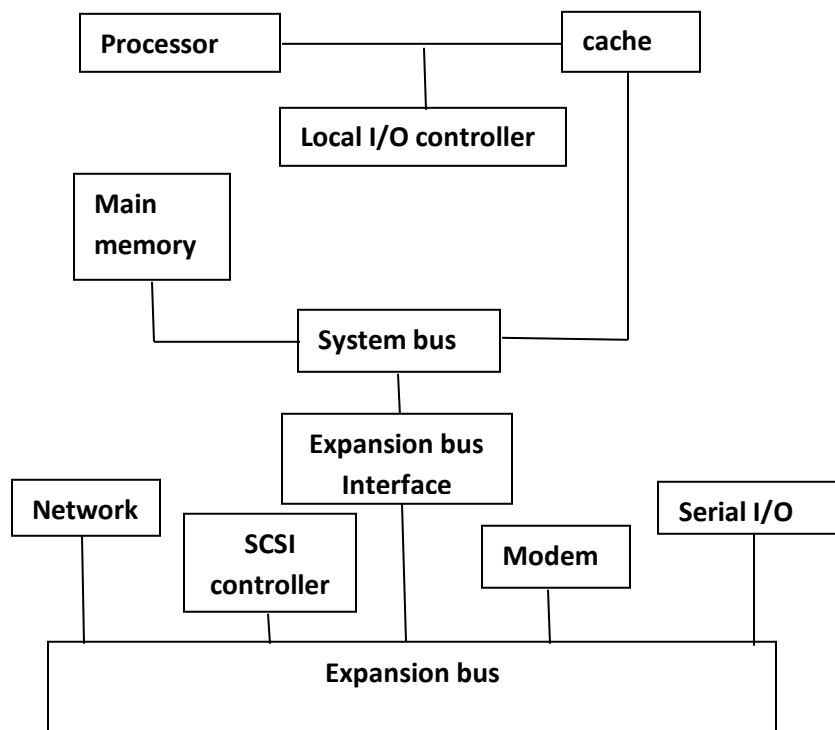
Traditional bus architecture :

This architecture control of three buses

1. Local buses

2. System buses

3. Expansion buses



The local bus connection to the processor, each memory and a local devices. The cache memory insulates the processor from accessing the main memory frequently.

The main memory is connected to the system bus. I/O transfer to and from the main memory and will not enter the processor activity. The expansion bus can be utilized to attach difficult I/O devices such as SCSI, LAN serial interface to support printer or scanners.

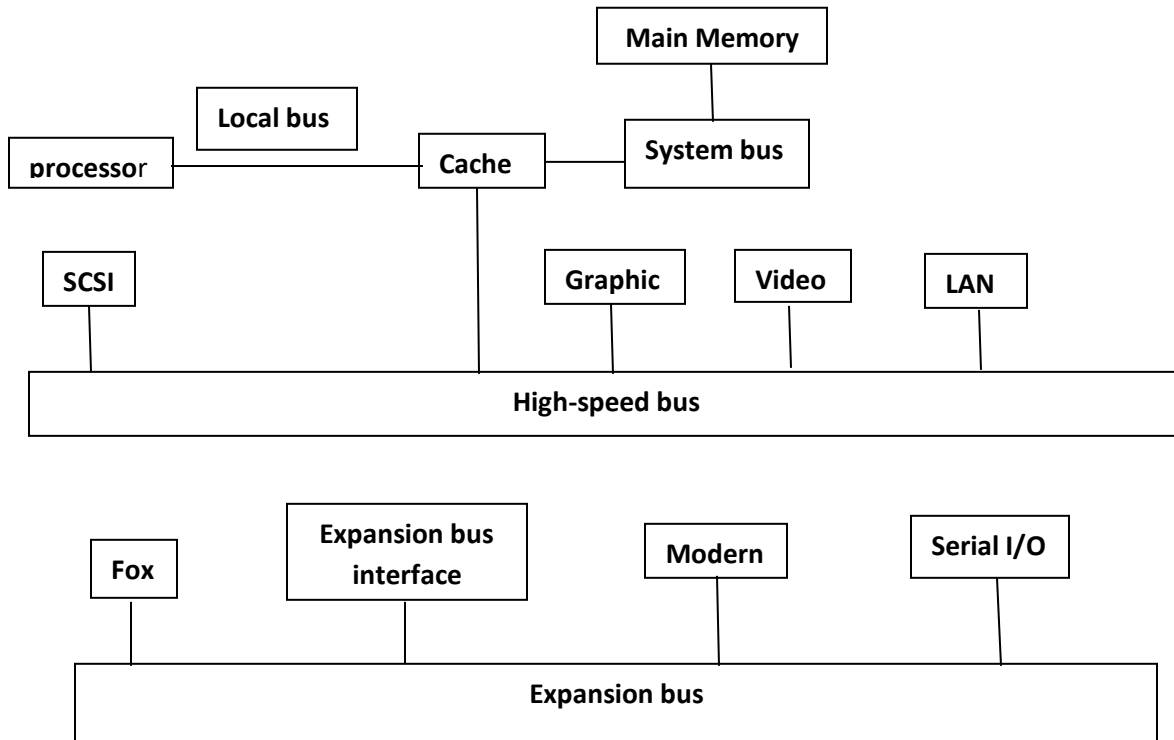
High performance bus architecture;

This architecture is consist by of

1. System bus

2. High speed bus
3. Expansion bus

High performance integrated bus architecture is similar to the traditional architecture .



(High performance integrated bus Architecture)

This way then high speed devices are more closely integrated with the processor through the high speed and of the same line leaving the processor independent.

Basic parameter of bus design :

Basic parameter of bus design is

1. Type of bus

The bus is two types (i) dedicated and (ii) Multiple

Dedicated bus : A bus is called a dedicated bus when a computer component permanently assigned to a function .

Multiplexed bus : Multiplexed connect each module by a interconnect all I/o module .

2. Width of bus : In three type address data and control bus .

The width of data bus has an impact on the performance of the system .wider data bus, the greater no of bits transferred at a time .

3. Method of Arbitration : in two type

- (i) Centralized (ii) distributed

Centralized: in a centralized scheme , a single hardware device known as “ bus controller or orbiter. In responsible for allocating time on the bus.

Distributed:- In this system each module control access control logic and the modules act together to share the bus.

Method of Timing:- Asynchronous and synchronous

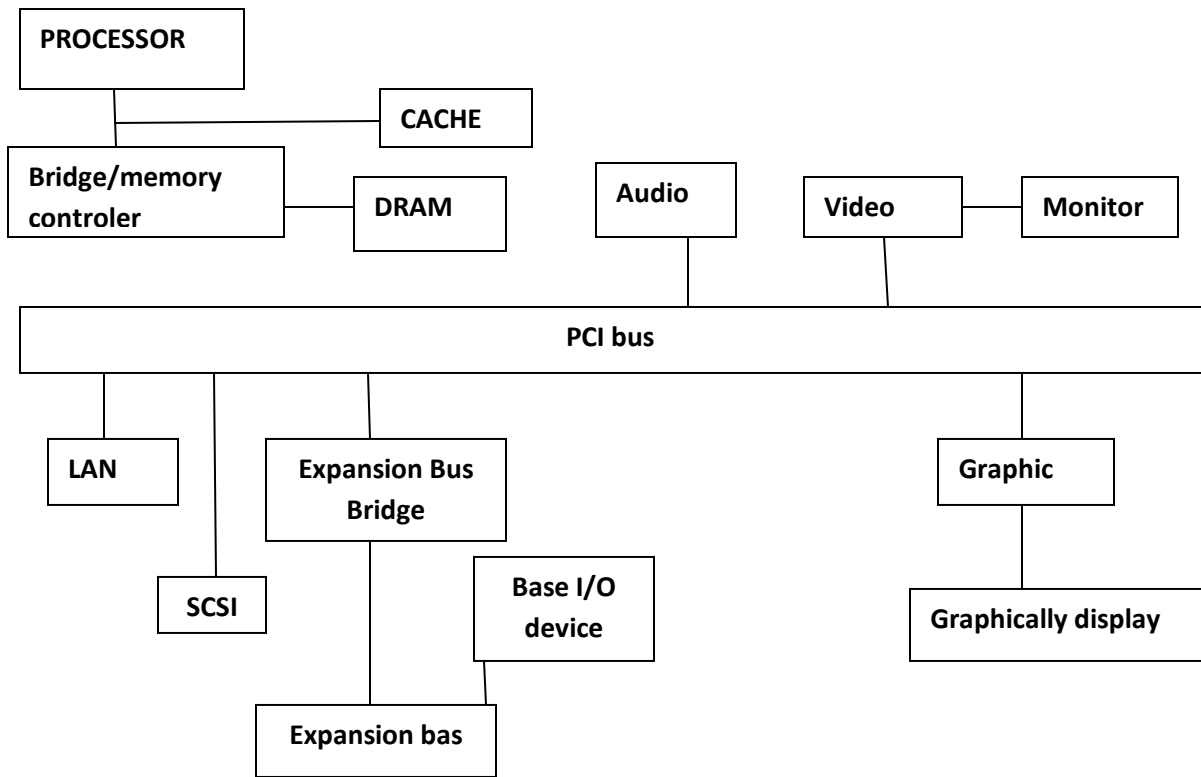
- I. **Synchronous timing**:- In this method of timing , a clock controls the occurrence of events on the bus.
- II. **Asynchronous timing**:- In this method of timing, the occurrences of the previous event are sequentially follows that event.

Types of data transfer:-

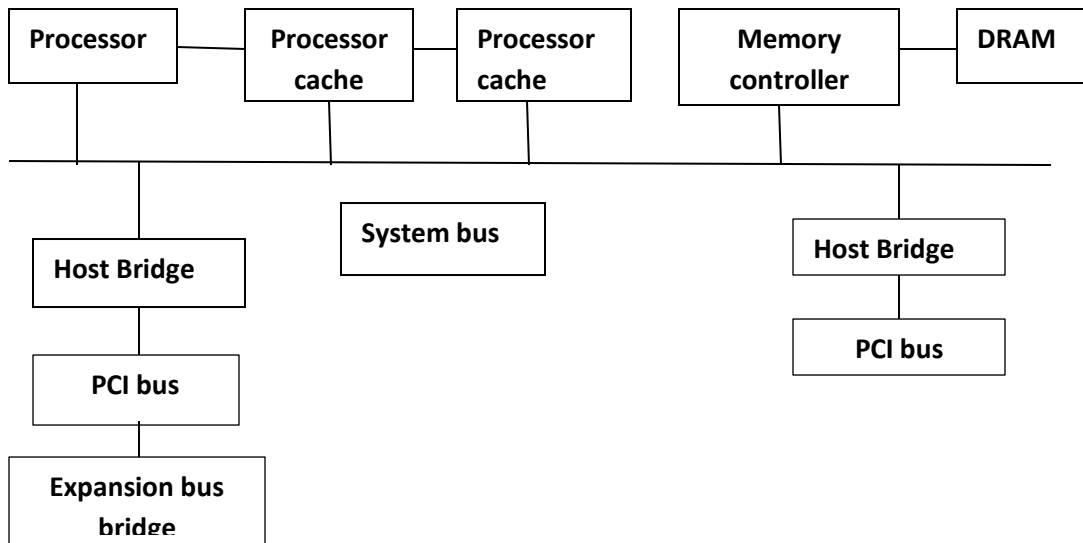
- i. **Read/write**:- All buses supports both read and write operation.
- ii. **Read-modify-write**:- It is simple a read followed by an immediate write to the same address.
- iii. **Read-after-write**:- this operation consisting of a write followed by an immediate read operation from the same address.
- iv. **Block transfer**:- Same bus system supports block data transfer operation.

Peripheral Component Interconnection (PCI) Bus.

It is a high b and processor independent peripheral bus. It is using 64 data lines at 66MHZ. Data transfer rate is 528 mbps to 4224 gbps. It is a high speed and income operation for most I/O system requirement. It makes use of synchronous time and a centralized obliteration scheme.



(PCI bus in a single- processor system)



SCSI

From Wikipedia, the free encyclopaedia. A SCSI connector (pronounced “scuzzy”) is used to connect computer parts that use a system called SCSI to communicate with each other.

A SCSI connector (pronounce “scuzzy”) is used to connect computer parts that use what is called SCSI to communicate with each other. Generally, two connectors, designated male and female, plug together to form a connection which allows two components, such as a computer and a disk drive, to communicate with each other. SCSI connectors can be electrical connectors or optical connectors.

A stack of external SCSI devices displaying various SCSI connectors. There have been a large variety of SCSI connectors in use at one time or another in the computer industry. Probably no computer interconnect (with the possible exception of RS-232 serial) has caused as much confusion. Twenty-five years of evolution and three major revisions of the standards resulted in requirements for Parallel SCSI connectors that could handle an 8, 16 or 32 bit wide bus running at 5, 10 or 20 Mbit/s, with conventional or differential signalling. Serial SCSI added another three transport types, each with one or more connector types. Manufacturers have frequently chosen connectors based on factors of size, cost, or convenience at the expense of compatibility.

SCSI often makes use of cables to connect devices together, in a typical example, a socket on a computer motherboard would have one end of a cable plugged into it, while the other end of the cable plugged into a disk drive or other device. This would mean that four connectors were involved in wiring the disk drive and computer together: the connector on the motherboard, the connectors at each end of the cable, and the connector on the disk drive. It is sometimes possible to have cables which have different types of connectors on them, and some cables can have as many as 16 connectors (allowing 16 devices to be wired together). Some types of connectors are typically used inside a computer or disk drive case, while others are used to connect a computer to a separate device such as a scanner or external disk drive SCSI and devices

Although not all devices support all levels of SCSI, SCSI standards are generally backward compatible. That is, if an older peripheral device is attached to a newer computer with support for a later standard, the older device will work at the older and slower data rate. In personal computing SCSI interfaces have been replaced, for the most part, by Universal Serial Bus (USB). In the enterprises, SCSI is still used in server farms for hard drive controllers.

Common SCSI components

There are several components used in SCSI/ storage systems: Initiator. An initiator issues requests for service by the SCSI device and receives responses. Initiators come in a variety of forms and may be integrated into a server's system board or exist within a host bus adapter. iSCSI connectivity typically uses a software-based initiator. Target. A SCSI target is typically a physical storage device (although software-based SCSI targets also exist). The target can be a hard disk or an entire storage array. It is also possible for non-storage hardware to function as a SCSI target. Although rare today, it was once common for optical scanners to be attached to computers through the SCSI bus and to act as SCSI targets. Service delivery subsystem. The mechanism that allows communication to occur between the initiator and the target; it usually takes the form of cabling.

Expander. Only used with serial-attached SCSI (SAS); allows multiple SAS devices to share a single initiator port SCSI standards Current SCSI technologies can transfer up to 640 megabytes per second (Mbps).



[SCSI standards chart]

Conventional PCI, often shortened to PC, is a local computer bus for attaching hardware devices in a computer PCI is the initialism for Peripheral Component interconnect(2) and is part of the PCI Local Bus standard. The PCI bus supports the functions found on a processor bus but in a standardized format that is independent of any particular processor's native bus. Devices connected to the PCI

bus appear to a bus master to be connected directly to its own bus and are assigned addresses in the processor's address space.[3][page needed] it is a parallel bus, synchronous to a single bus clock

Attached devices can take either the form of an integrated circuit fitted onto the motherboard itself (called a planar device in the PCI specification) or an expansion card that fits into a slot. The PCI Local Bus was first implemented in IBM PC compatibles, where it displaced the combination of several slow ISA slots and one fast VESA Local Bus slot as the bus configuration. It has subsequently been adopted for other computer types. 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 PCI video cards replaced ISA and VESA cards until growing bandwidth requirements outgrew the capabilities of PCI The preferred interface for video cards then became AGP, itself a superset of conventional PCI, before giving way to PCI Express.[4]

What makes the PCI bus one of the fastest I/O bus used today?

Three features make this possible:

Burst Mode: allows multiple sets of data to be sent (Kozier, 2001a) Full Bus Mastering: the ability of devices on the PCI bus to perform transfers directly (Kozier, 2001c)

High Bandwidth Options: allows for increased speed of the PCI (Kozier, 2001a) How PCI Works: Installing A New Device How PCI Works: Installing A New Device:

Once a new device has been inserted into a PCI slot on the motherboard

1. Operating System Basic Input/Output System (8105) initiates Plug and Play (PnP) 8105
2. PnP BIOS scans the PCI bus for any new hardware connected to the bus. If new hardware is found, it will ask for identification. The device will respond with its identification and send its device ID to the BIOS through the bus

PnP checks the Extended System Configuration Data (ESCD) to make sure the configuration data already exists for the card. (If the card is new, then there will be no data for it.)

Example: PCI-based sound card

The sound card will convert the analog signal to a digital signal.

The digital audio data carried across the PCI bus to the bus controller, which determines which device on the PCI device has the priority to send data to the central processing unit (CPU) and whether the data will go directly to the CPU or to the system memory.

USB Device Basics

- A USB device can never start sending data without first being asked by the host controller
- Single-master implementation
- Host polls various devices
- A device can request a fixed bandwidth (for audio and video I/O)
- Universal Serial Bus is a misnomer...
- Actually a tree built out of point-to-point links
- Links are four-wire cables (ground, power, and two signal wires)
- Fast
- Bi-directional
- Isochronous
- Low cost
- dynamically attachable serial interface
- consistent with the requirements of the PC platform of today and tomorrow

Parallel processing: - The purpose of parallel processing is increasing the computational speed of CPU. Instead of processing each instruction sequentially a parallel processor performs concurrent data processing tasks to achieve faster execution time. Suppose when an instruction is being executed in the ALU, the next instruction can read from memory. If two or more ALUs are included within the processor unit, two or more instructions execute at the same time. The purpose of parallel processing is to speed up the processing capabilities. The amount of hardware increases the parallel processing with it, the cost of the system increases.

To reduce the cost there are some parallel processing techniques

1. Large computation can be divided into many points that can be performed in parallel (PTO)
2. Pipeline processing

Parallel structure: -

SISD: - A single processor computer system is called a single instruction single data stream. The instruction reading from memory forms.

Instruction stream: - Various operations performed on the data in the processor form continuity a data stream. A program executed by the processor continues a single instruction stream, and the sequence of data items that it operates on continues two single data streams.

SIMD: - Single instruction stream and multiple data stream. Single stream of instruction is broadcast to each processor operates on its own data. All processors execute the same program but operate on different data is called SIMP.

MIMD: - Multiple instruction stream multiple data stream it involves a number of independent processors, each executing a different program and accessing its own sequence of data items. Such mechanism is called MIMD.

MISD: - Multiple instruction single data stream. In such a system a common data structure is manipulated by separate processors. Each executing a different program parallel processing is established by distributing the data among the function units. All function units under the supervision of a control unit

Adder subtractors

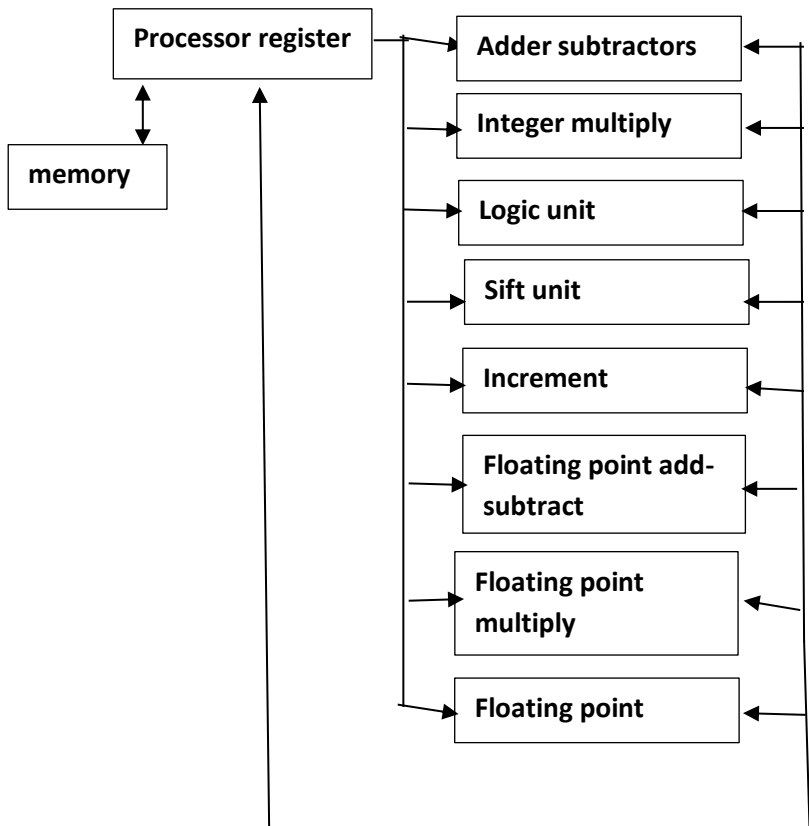


Figure should the possible way of separating the execution unite into eight fusional units. Operating in parallel.

Deferent: - A parallel processing in a technology in which a multiprocessor system in used to salve complex problem faster by breaking the problem to be processed simultaneously by different processors of the multiprocessor system.

Different system bus: - A bus that connect major components in a multiprocessor system, such as CPU, IOPs and memory in called system bus. A typical system bus consists of approximately 100 signal lines. Wise line are address. Data control and power distribution line that supply power to components.

Pipelining

Pipelining is a technique of decomposing a sequential processing into sub operation, with each subprocess being executed in a special dedicated segment that operates concurrently with all other segments. Pipelining provides a way to start a new task before an old one has been completed Each segment performs partial processing. The result obtained from the computation in each segment is transferred to the next segment in pipeline. The final result is obtained after the data have passed through all segment.

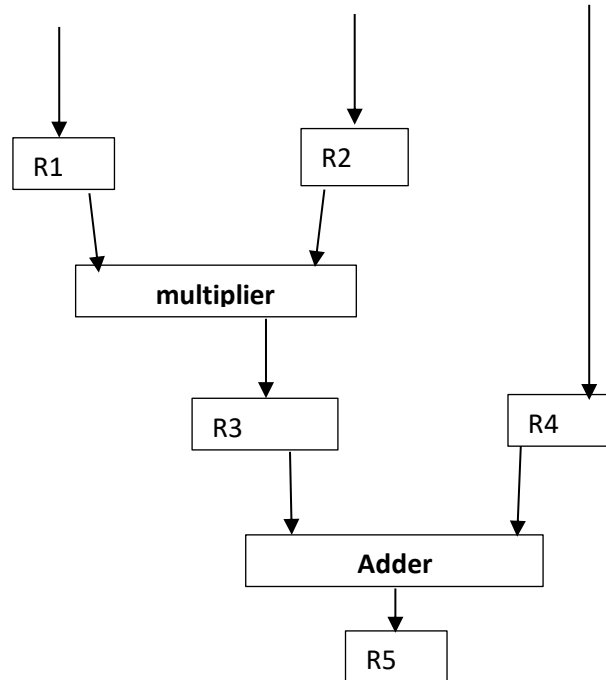
[The characteristic of pipelines that several computations can be in progress in distance segments at the same time.]

Let’s consider an example to perform the combined multiply and add operation with a stream of numbers. For I = 1 to 7 do

$$A [i] * B[i] + c[i];$$

Each segment consists of an input register followed by a combinational circuit. The register holds the data and the combination circuits performs the sub operation in the particular segment.

[In pipelining several computations can be in progress in process in distinct segments at the same time. The overlapping of computation is made possible by associating a register with each segment in the pipeline. The registers provided isolation between each segment so that each can operate on distinct data simultaneously.]



R1, -----R5 are registers that revise new data with every clock plus. The multiplier and adder and combinational circuits. The sub operation for the above processing can be written as

$R1 \leftarrow A[i]; R2 \leftarrow B[i];$ Inputs A[i] and B[i]

$R3 \leftarrow R1 \cdot R2; R4 \leftarrow C[i]$ Multiply and input C[i]

$R5 \leftarrow R3 + R4$ Add C[i] to product

The five registers are loaded with new data every clock pulse.

The first clock pulse transfers the values A [1] and B [1] into registers R1 and R2 respectively.

The second clock pulse transfers the product of r1 and R2 registers into register R3 and input data C [1] into register Ra. The same clock pulse transfers A [2] and B [2] into R1 and R2. The third Clock pulse places A [3] and B [3] R1 and R2, transfers the product of R1 and R2 into R#< transfers C [2] into R4 and places the some of R3 and R4 into R5.

Clock pulse numbers	Segment 1		Segment 2		Segment 3
	R1	R2	R3R4		R5
1	A1	B1			
2	A1	B2	A1*B1	C1	
3	A3	B3	A2*B2	C2	A1*B1+C1
4	A4	B4	A3*B3	C3	A2*B2+C2
5	A5	B5	A4*B4	C4	A3*B3+C3
6	A6	B6	A5*B5	C5	A4*B4+C4
7	A7	B7	A6*B6	C6	A5*B5+C5
8			A7*B7	C7	A6*B6+C6
9					A7*B7+C7

The process continues in the same manner for the first seven clock pules. In the eight clock pulse, no data to input the segment 1 Of the pipeline consisting of registers R1 and R2 lie idle, While the product of previous R1 and R2 computed and stored in the register R3 and new value of C input In the register R4. In the 9th clock cycle both segment 1 and 2 lie idle, while segment 3 continues with its computation. When no more input data are available, the clock continue until the pipeline produces the last output.

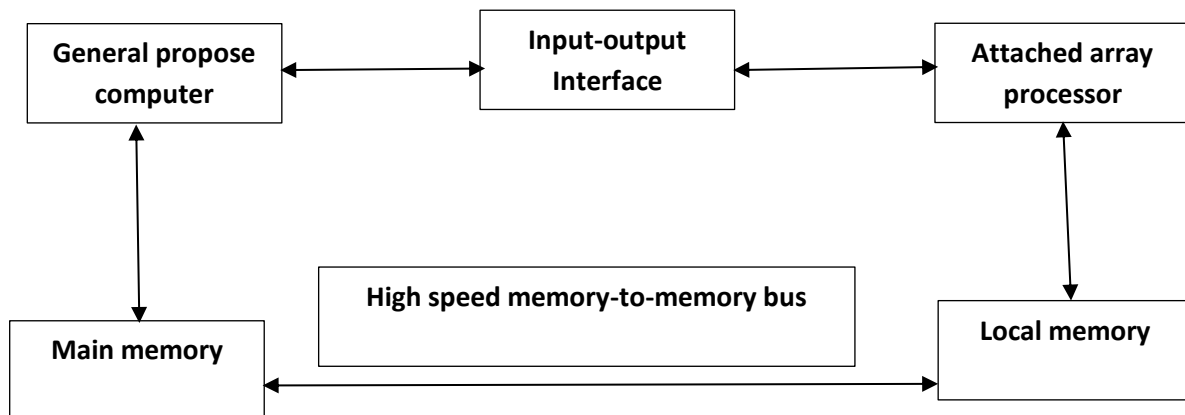
Array Processors

Array processors are highly specialized machines. These machines perform computation on large arrays of data. Two different types of processors are;

- Attached array processor
- SIMD array processor

Attached Array Processor

An attached array processor is an auxiliary processor attached to a general-purpose computer. Attached array processor is designed as a peripheral to the host computer and its purpose is to improve the performance of the computer by providing vector processing for complex scientific applications.



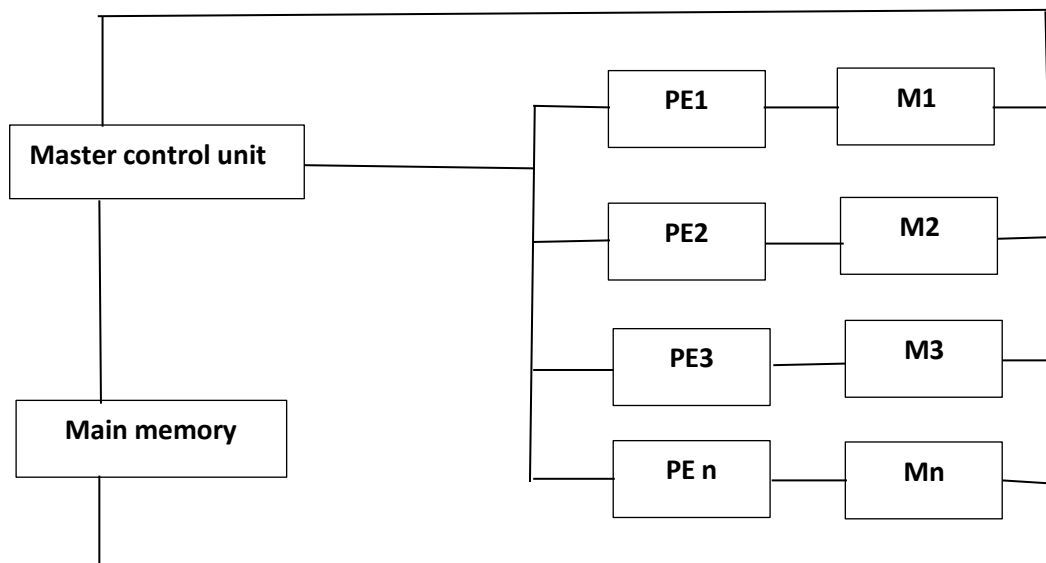
[Attached Array processor with host computer]

It has multiple functional units, which work in parallel over different type of data. It includes an arithmetic unit consisting of one or more floating multiple and adder. It has multiple functional units, which work in parallel over different stream of data.

The host computer is any general-purpose computer and the array processor is a back-end machine driven by host computer. The array processor is connected by an input-output controller to the host computer treats it like an external interface. The data for the attached processor are transferred from main memory to a local memory through a high-speed bus. The general-purpose computer performs all arithmetic and logical operations.

SIMD Array Processors

A SIMD array processor is processor that has a single-instruction multiple-data organization. An array processor consists of multiple processing elements (PEs) each having local memory in under the supervision of one control unit (U). An array processor can handle data streams.



[SIMD array processor organization]

The main memory is used for storage of the program. The function of the master control unit is to decode the instructions and determine how the instructions is to be executed. Scalar and program control instructions are directly executed within the master control unit. SIMD processors are highly specialized computers. They are suited for numerical problems that can be expressed in vector or matrix form.

Flynn's Classification

Flynn's classification is based on the multiplicity of instruction stream and data streams in a computer system. The sequence of instruction mead forms the memory constitute the instruction stream, and the data they approve on in the processor constitute the data stream.

Flynn's classification divided the computer into 4 categories

1. Single instruction Single data stream (SISD)
2. Single instruction Multiple data stream (SIMD)
3. Multiple instruction single data stream (MISD)
4. Multiple instruction Multiple data stream (MIMD)

SISD: - It represents the organization of a single computer. containing a control unit, a processor unit and a memory unit.

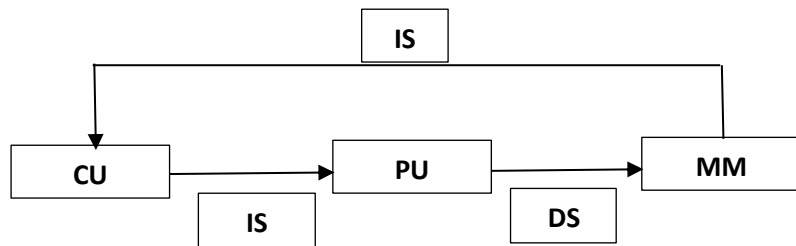
CU- Control unit

PU- Processor unit

IS- Instruction stream

DS- Data stream

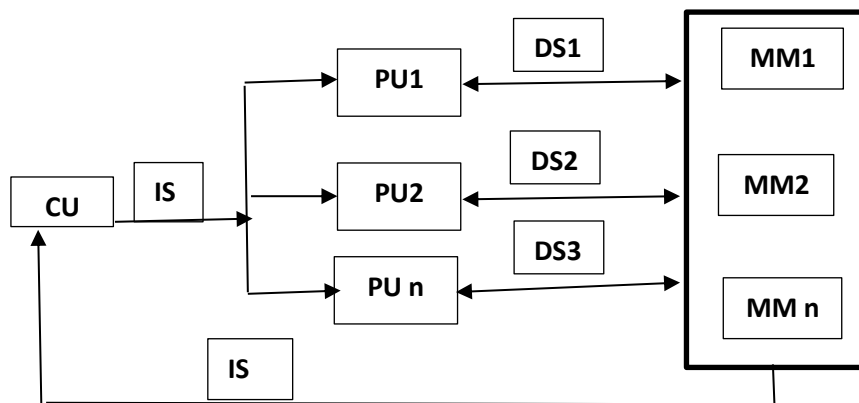
MM- Memory module



Most computers are built in SISD principle. Here instructions are executed sequentially and the system may or may not have internal parallel processing capabilities. A SISD system may have more than one functional unit but all their functional units are under the supervision of one control unit. In order to increase the processing speed, most SISD Uni processor systems are pipelined.

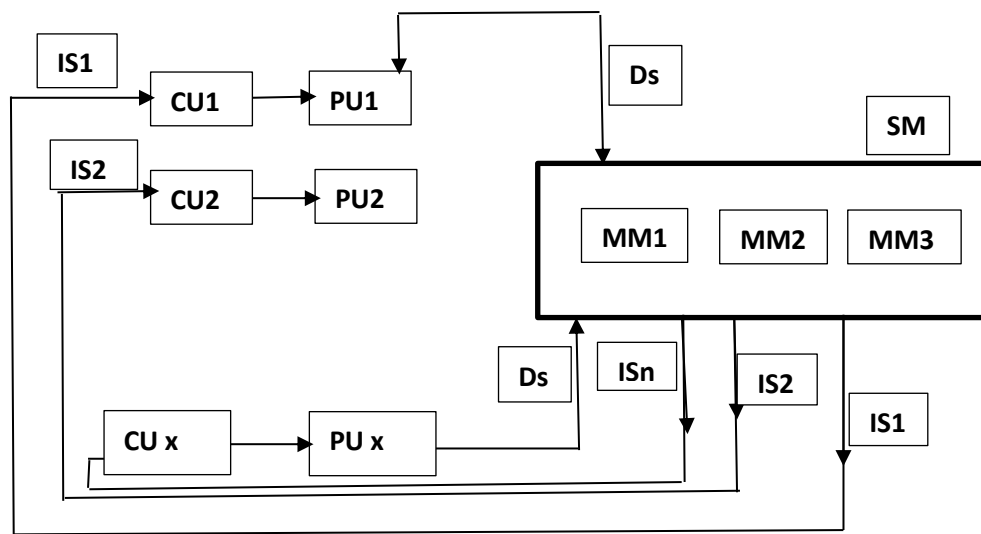
SIMD: It consists of multiple processing elements supervised by the same control unit.

All processing units receive the same instruction which is broadcast by the control unit but work on different data streams. The shared memory unit most contain multiple modules, so that it can communicate with all the processors simultaneously.



[SM- shared memory]

MISD: MISD organization consists of n processor units, each working on a different set of instruction but working on the same set of data. The output of one processor becomes the I/P to the other unit. This configuration is yet to be realized practically. Therefore it is given less importance.



MIMD : It implies interaction between 'n' processors because all memory stream and derived from the same data stream shared by all processors.

If the interaction between the processor is high, it is called a tightly coupled system. If the interaction between the processor is low, it is called a loosely coupled system. [Parallel computers appear as either SIMB or MIMD configuration]

