Dt-22/07/2020 3rd Sem CS&E

Th1 (Computer System Architecture)

Unit1

1.1) Basic structure of Computer Hardware

Computer architecture in generally Covers three aspects of Computer design i.e. Computer Hardware, Instruction set and Computer Organizatn.

→ Computer hardware are the physical part of Computer such as CPU, Monitor, keyboard and mouse etc.

→ "A digital Computer takes input from keyboard / mouse and process it according to a list of internally stored instructions and procedures that resulting output information."

Computer Hardware

→ It Consists of electronic circuits, displays, magnetic & optical storage media and Communication BUS.

Instruction Set

→ It provided by the user to perform task. It performed by using register, memory system etc.

→ Instruction sets are two types CISC (Complex Instruction set Computer) and RISC (Reduced Instruction set Computer)

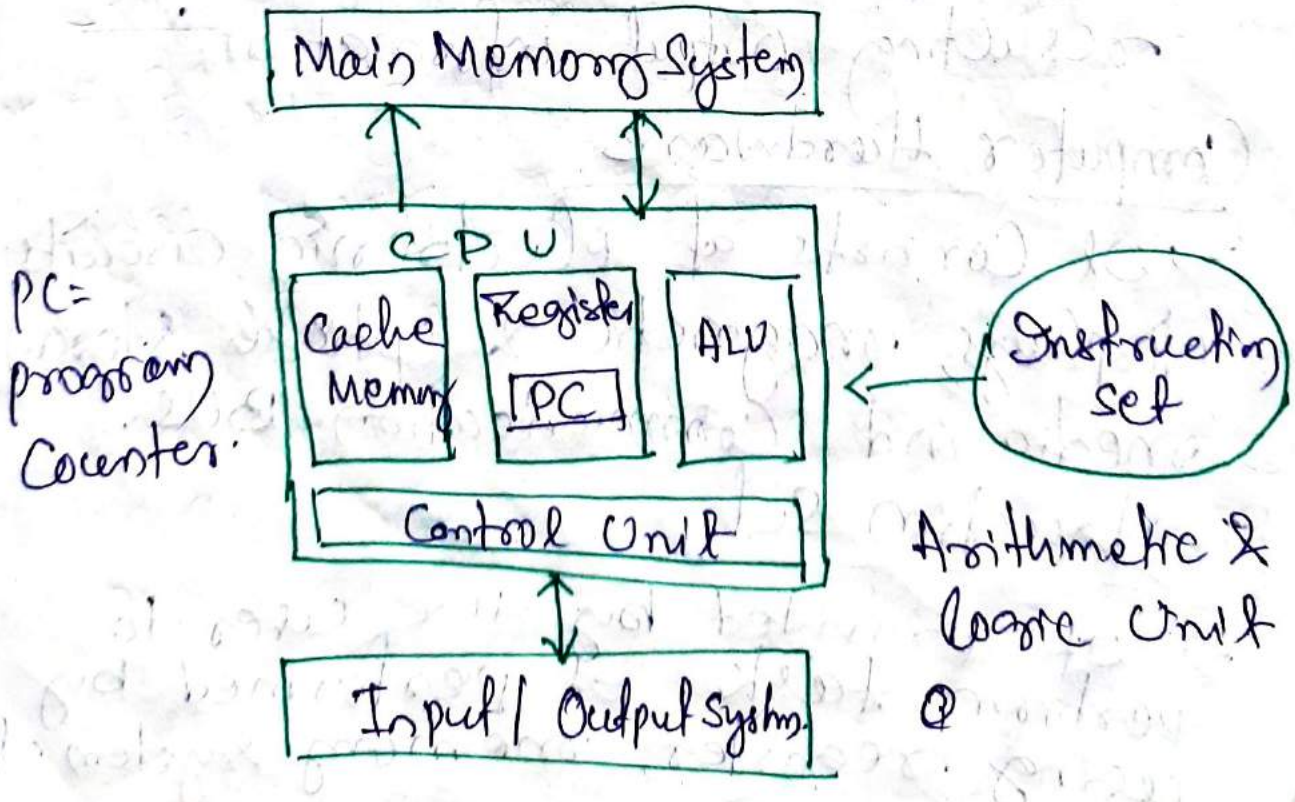
Computer Organization

→ It includes high level of design such as memory system bus structure and the design of the CPU.

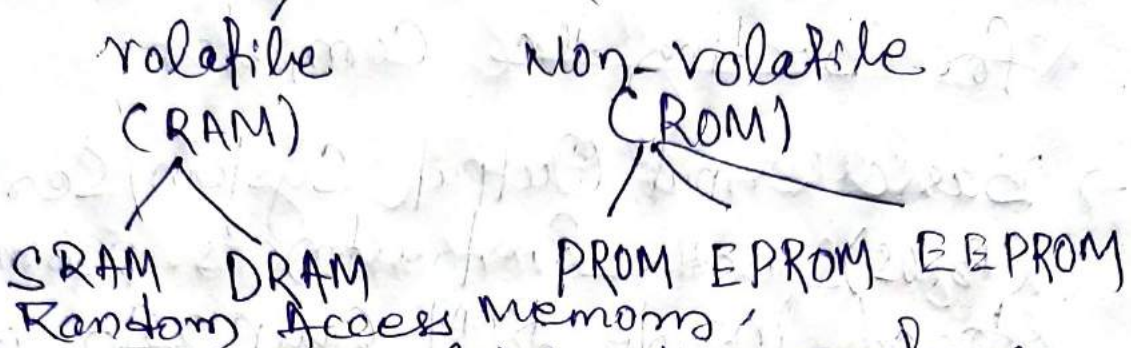
1.2) Functional Units

There are three major parts of function units of a Computer System

1) Main Memory System
2) CPU (Central processing Unit)
3) Input / Output System



Main Memory System



Random Access Memory

RAM :- It is volatile in nature;

→ It is the internal memory of CPU for storing data, program & result of program.

→ It is volatile because data stored in memory is losted when Computer turned off / power failure.

→ It is two types
- SRAM (static RAM)
- DRAM (Dynamic Random Access Memory)

SRAM :- This memory maintain its data as long as power remains applied, so it is not refreshed as DRAM.

DRAM :- This memory continously refreshed in order to maintain the data. These cell are composed of one capacitor and one transistor.

ROM :- It stands for Read Only Memory.

→ It stored BIOS instructions for booting the Computer;

→ Basic Input Output System/Services (BIOS) is the instruction is permanently stored inside ROM chip so it is non-volatile in nature.

→ Data are remain constant, when the computer is turned off.

→ ROM is three types :-

⇒ PROM (Programmable Read Only Memory)

→ This ROM is read-only memory that can be modified only once by a user.

* EPROM (Erasable & Programmable ROM)

→ It can be erased by Ultra-Violet (UV Ray) and again data stored into it and so on.

* EEPROM (Electrically Erasable & programmable ROM)

→ It can be programmed and erased electrically.

→ It can be reprogrammed about ten thousand times.

CPU

* Cache Memory

→ Cache memory is a very high speed Semiconductor memory which can speed up CPU.

→ Memory of cache size is very low but processing is very fast. So it is called as fastest memory in the CPU.

→ It used to hold those parts of data & program which are most frequently used by CPU.

→ The parts of data and program are transferred from disk to cache memory by operating system. then CPU can access them.

Advantages

→ It is faster than main memory.
→ It consumes less access time as compared to main memory.

Disadvantages

→ It has limited capacity.
→ It is very expensive
→ It is two types L1 cache & L2 cache.

3rd Sem 24/07/2020 Sub- TI:- CSA

Registers

→ Registers are a type of Computer memory used to quickly accept, Store and transfer data and instruction that are used by CPU.

→ A register may hold an instruction, storage address or any data.

→ There are various registers such as:-
* Data Register (DR) - 16 bit
→ It holds memory operand
Ex:- +, -, *, / etc

* Address Register (AR) - 12 bit
→ It holds address for the memory.

* Accumulator (AC) - 16 bit
→ It is processor register also called CPU register.

* Instruction Register (IR) - 16 bit
→ It holds instruction code

* Program Counter (PC) - 12 bit
→ It holds the address of the next instruction

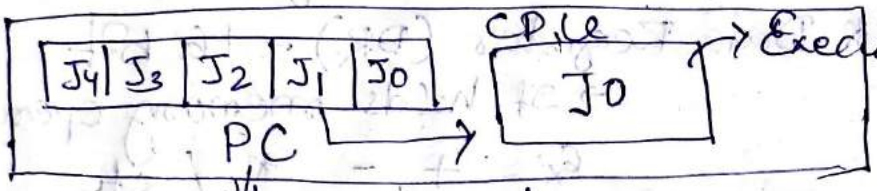
* Temporary Register (TR) - 16 bit
→ It holds temporary data

* Input Register (INPR) - 8 bit
→ It carries input character by user.

* Output Register (OUTR) - 8 bit
↳ It carries output character after processing the input data.

Program Counter :-

→ It holds the address of the next instruction to be read from memory after the current instruction is executed.



ALU (Arithmetic and Logic Unit)

→ ALU performs mathematical Calculations and takes logical operations.

→ Arithmetic Calculations include addition, subtraction, multiplication and divisions.

→ Logical Operations used for Comparison of two items to see which is larger, smaller or equals

CU (Control Unit)

→ It Controls all operations from input instruction to output result.

→ It decodes the fetched instruction and sends control signal for I/O operation.

Input / Output System

Input System / Unit

→ It Consists of input devices (peripheral) that are attached to the Computer.

→ This devices takes data as input and Convert it into binary code which can be understand by CPU for execution.

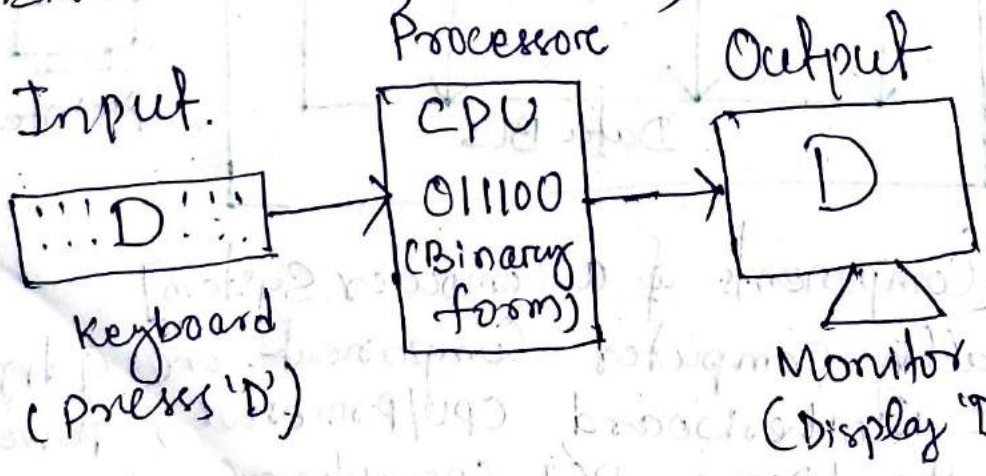
→ Ex:- keyboard, mouse, NIC card. etc. PCI Express card,

Output System / Unit

→ It Consist of output devices that are attached with the Computer

→ It Converts the binary data sends from CPU to human understandable language.

Ex:- Monitor, Printer, plotter etc.



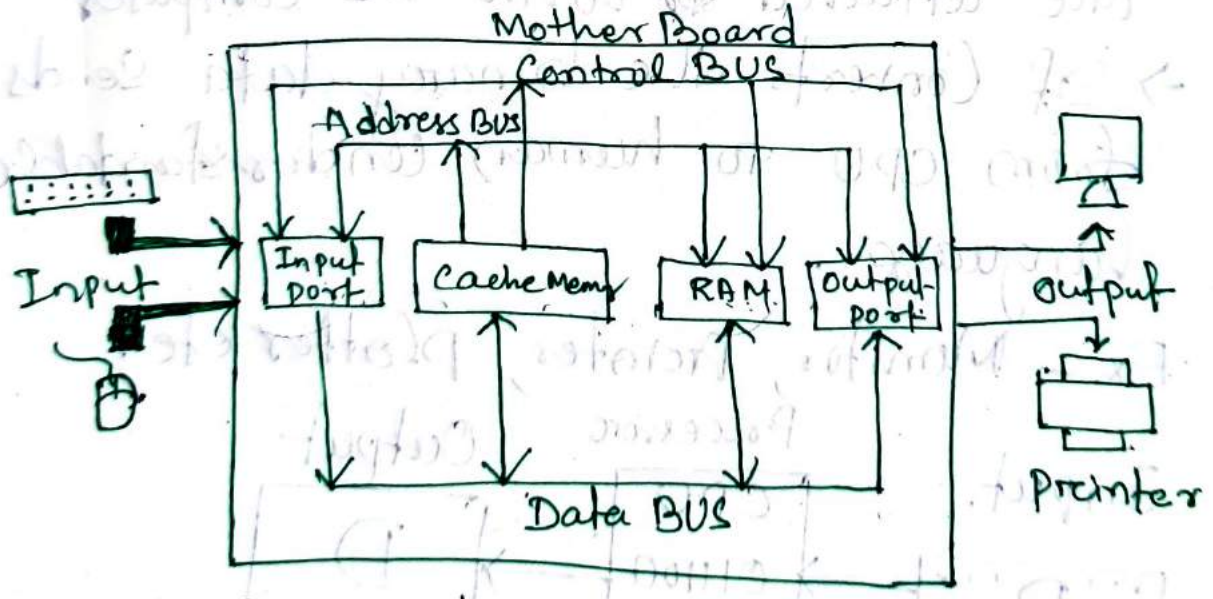
1.3) Computer Components

→ The major components of general-purpose computer system are Input Unit, Main/Internal Memory or Secondary storage, Output Unit, CPU.

→ The CPU is included ALU, CU and Registers.

→ So all units also refered to function units

→ Input/Output devices are not integral part of CPU so that it is called peripherals.



[Components of a Computer System]

→ Basically Computer Component are 9 types such as:- Motherboard, CPU/Processor, power supply, Hard Drive, PCI (peripheral Component Interconnect) card, Graphics Card, RAM, I/O Devices.

3rd Sem 25/07/2020 Sub:- TH1 (CSA)

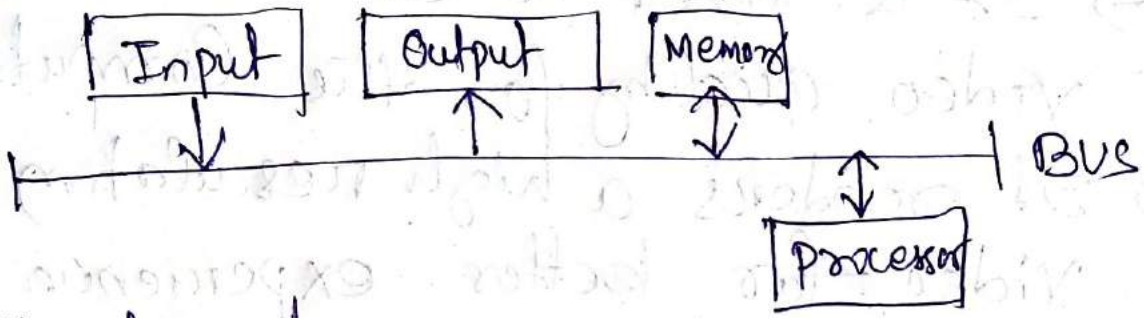
Continue from Components of Computer ....

BUS

→ A group of lines that served as connecting path for several devices called a BUS.

→ A Bus is used to transfer data, address and control. (one bit per line)

Ex) - A processor to printer shared a common bus for communicating



Motherboard

→ It is the main circuit board which is connected to all other components for run a computer.

Processor (CPU)

→ It is also called microprocessor.

→ It is placed on the motherboard on CPU Soket area.

→ It is measured by megahertz and gigahertz.

Hard Disk Drive / Storage Drive

→ It stores data for feature use.
→ It is called as permanent storage medium.
→ The size of hard disk is measured by gigabyte and terabyte.
→ HDD (SSD)

Graphics Card (Video Card)

→ VGA Card
→ It is used to enhanced the video quality for the Computer.
→ It renders a high resolution video for better experience.
→ PCI slot (peripheral Communicaly Interconnected)

PCI Expansion Slots

→ Additional audio card, video card, N-Computing, additional USB Connection, NIC card etc.

USB :- Universal Serial BUS
NIC :- Network Interface Card (Internet Connectivity is provided)
(RJ45)

1.4) Performance Measure.

→ The total time required to execute a program is called performance measure.

→ The important measures of a computer is how quickly it can execute a program.

→ There are three major components that affect for performance.

1) Hardware Design (Architecture)
2) Instruction Set
3) Compiler.

→ Processor time to execute a program depends upon the hardware involved the execution of individual machine instruction.

→ A processor is very small part and cache memory is attached on a single integrated chip is called CPU.

→ All performance is measured by CPU Clock, CPU Cycle, Clock Cycle and Clock rate.

→ Basic performance measured equations as follows.

$$T = \frac{(N * S)}{R}$$

T = Execution Time
N = Number of Instruction
S = Average Cycle per Second.
R = Clock rate per Second.

→ The Execution of each instruction is divided into several parts / steps each of which Complets in <u>one Clock Cycle</u>.

Hertz :- One Cycle per Second.

1 Megahertz (MHz) = One million Cycle per Second (1000) Cycle

1 Gigahertz (GHz) = One Billion Cycle per Second.

10,000 lakhs
One million thousand Cycle per Seem.
1000000

3rd Sem 27/07/2020 Sub:- TH1 (CS&A)

1.5) Memory Addressing and Operations

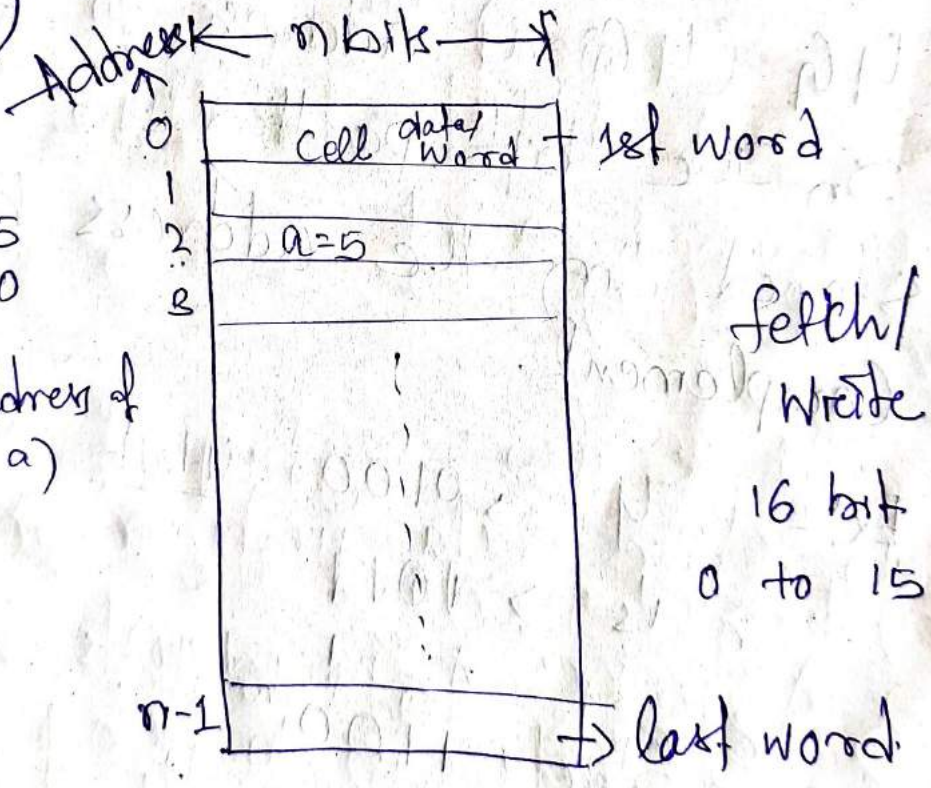
what is a memory?

→ ~~A~~ A group of adjacent cell is called memory.

→ A cell is a smallest area of memory where we stored data, word in bits are bytes.

→ Data is stored in to memory in bit pattern that 0 & 1 (binary bits)

→



Ex:- a=5
b=10

&a (address of a)

→ the memory organisation is stored the data or word, in the length of bits or byte pattern is by ASCII characters.

American Code for Information Interchange

Ascii of A = 65, a = 97 & so on
0 = 48 etc

→ All memory is store in the sequence of cell which is represented in bit form.

→ A memory address is 16-bit so that 2^{16} (167,77,336) length of bit stored.

→ A 32 bit addressing that provides 4G (4 Giga) address to store data.

→ In the bit pattern the data are stored in the address by 2's Complement.

0
1
10
11
100

Ex:- 4 → 0100 4-bit

1's → 1011

+ 1

2's 1100

→ 4 bits binary pattern 2's 1100 address for 4-bit pattern address

Bit addressing (32-bit)

(a) All addressing in memory are 2's complent of signed integer value.

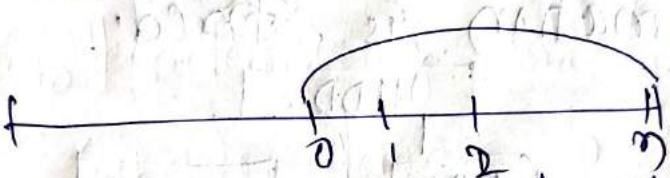
b_{31}			...	b_1	b_0

So

0 = for positive value address

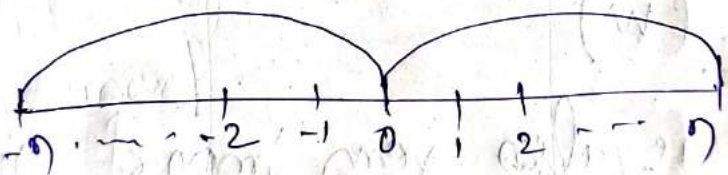
1 = for negative value address

Signed Integer (0 to +, ... n)



Signed integer.

Unsigned (-n, ... -2, -1, 0, 1, 2 ... n)



5) Byte of Word (32-bit)

→ 8 bits = 1 byte.

→ So the word length of addressing is divided by 8 bits.

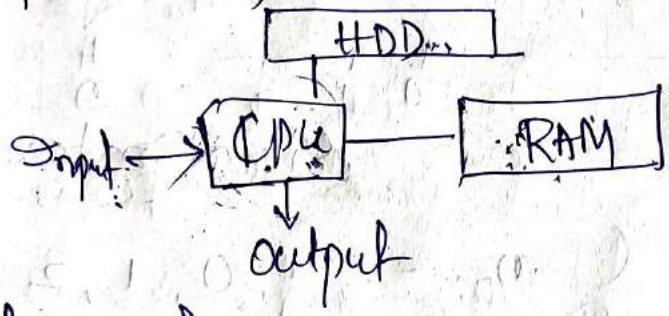
8bits	8bit	8bit	8bit
ASCII Character	ASCII Character	ASCII Character	ASCII Character.

Memory Operations

→ Basically we can perform two major operations i.e Load or read ; Write or operation.

→ **Read (R)**

Ex:- A processor is wants to read a location then processor sends address to memory where the data/ information is stored.



→ **Write (W)**

→ It is also very import operation to store data into memory.

Ex→ A processor is wants. Write some data in to the HDD then the main memory sends the address of current holds data to the HDD for save the the data/ information.

3rd Sem 29/07/2020 Sub:- TH1 (CS9)

memory addressing and operations.

→ Read | Write.

1.3) Components of computer.

Memory :- Secondary, Main Memory, Cache, Register etc

Secondary Storage

→ It is stored data ~~permanently~~ permanently (in cell).

→ Address of memory location.

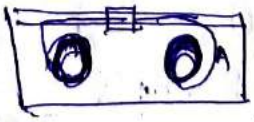
→ By access of secondary storage it is two types:-

① Sequential Access

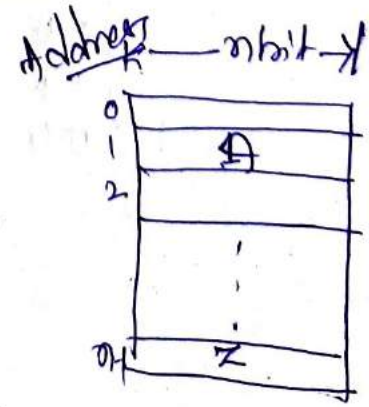
② Direct Access

Sequential Access

→ The data may be read or write in to the memory is sequential that means data should be store one after another.

Ex:- Magnetic Tape 

Direct Access



→ In this medium the data should read or wire in the direct memory address.

Ex:- Hard Disk Drive.

Pendrive (flash Drive)

Digital Data

→ It stores data in the medium in binary bits i.e 0 and 1

Computer Datatypes (Number System)

① Decimal :- (Base 10)

0, 1, 2, 3, — — —, 9

② Octal (Base 8) :-

0, 1, 2, 3, 4, 5, 6, 7

③ Hexadecimal (Base 16)

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

④ Binary (0 and 1)

! AND ⊕
OR
NOT
NOR

Binary

0 — 0
1 — 1
10 — 2
11 — 3
100 — 4
101 — 5
110 — 6
111 — 7
1000 — 8

Equivalency Table

Decimal	Binary	Octal	Hexadecimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	61	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	f

ASCII :- ~~Extended~~ American standard Code for Information Interchange.

A = 65 ; a = 97 , 0 - 48

Signed / unsigned bit

3rd Sem 31/07/2020 Sub:- TH1(C&A)

1.5) Memory address

Memeroy Addressing Scheme (word) (byte)

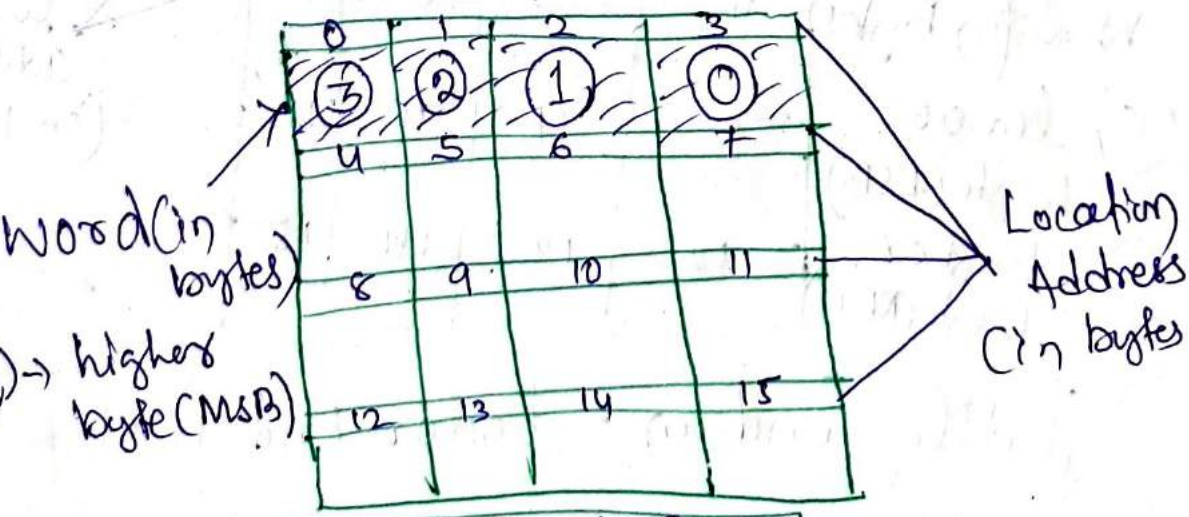
① Big-Endian Scheme
② Little-Endian Scheme.

Big-Endian Scheme

→ The term of "endian" or endness or endianness" is a type of byte ordering used to represent words generally byte ordering and its used to represent integer as a sequence of bytes (word).

→ On the big-endian scheme the higer bytes of a word are stored on lower location address.

Ex:- 0 1 2 3 (word)



Big-Endian: Big End first

(MSB) | (LSB)

More Significant Bits/ Byte,

Ex:- 1 3 (4) (MSB)
1st 2nd 3rd

Less Significant Bits/ Byte

Ex:- (1) 3 4
1st 2nd 3rd.

1 nibble = 4 bits
1 Byte = 8 bits.

→ Here the highest byte (MSB) denoted as ③ (byte-4) of the word is stored in the lower byte location i.e (0).

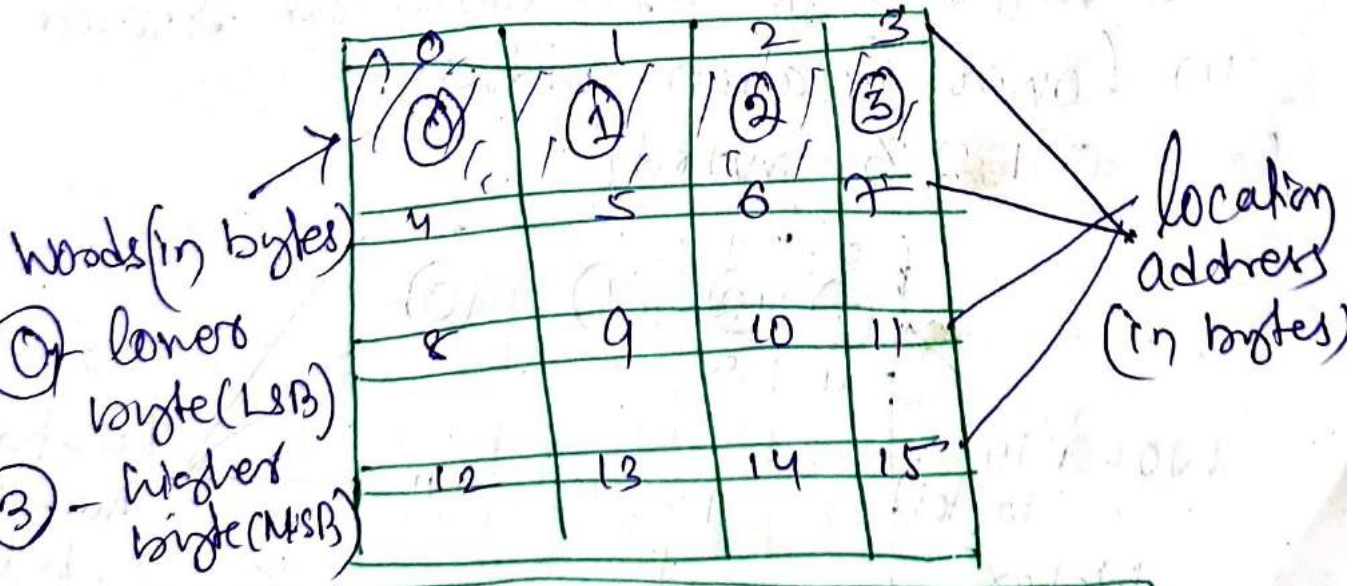
→ The 3rd byte (denoted as ②) is stored the (2) location i.e 1 and so on.

→ Some well known processor architecture using big-endian scheme include Motorola, Spark, PowerPC, System 370 et

2) Little-Endian Scheme

→ In the little-endian scheme the lower byte (less significant bit/bytes) of a word occupies the lower byte address of the memory location.

Ex. - 0123 (word)



Little-endian = Lower End first

→ Here the lower byte (less Significant byte) 1st byte is denoted as (0) of the word is occupies in the lower byte location address (1st denoted as (0)).

→ The 2nd byte denoted as (1) of the word is occupies the location address is (1) and so on.

Ex. - The processor architecture cered little-endian scheme ~~source~~ Z80, intel x86 (32 bit) etc.

a) What are the major difference between big-endian scheme and little-endian scheme.

3rd Sem 04/08/2020 Sub:- TH1(CSA)

Unit-2

Instructions and Instruction Sequence.

2.1) Fundamentals to Instructions

What is an instruction?

→ The Combination of Operand and Operator is known as an instruction.

Ex:- a + b → operator; a, b → operand; { Expression / Instruction

→ An instruction consists of two parts

① Opcode ② operand

| Opcode | Operand |

Ex: ADD LOCA, R0
(operator) (source) (destn)

a=6, b=5
C=11

C = a + b Expression

C, a, b = operand

+ = operator.

→ The instruction is addition operation and the following steps are executed.

1) Fetch the instruction from "Main Memory" to the processor (CPU).

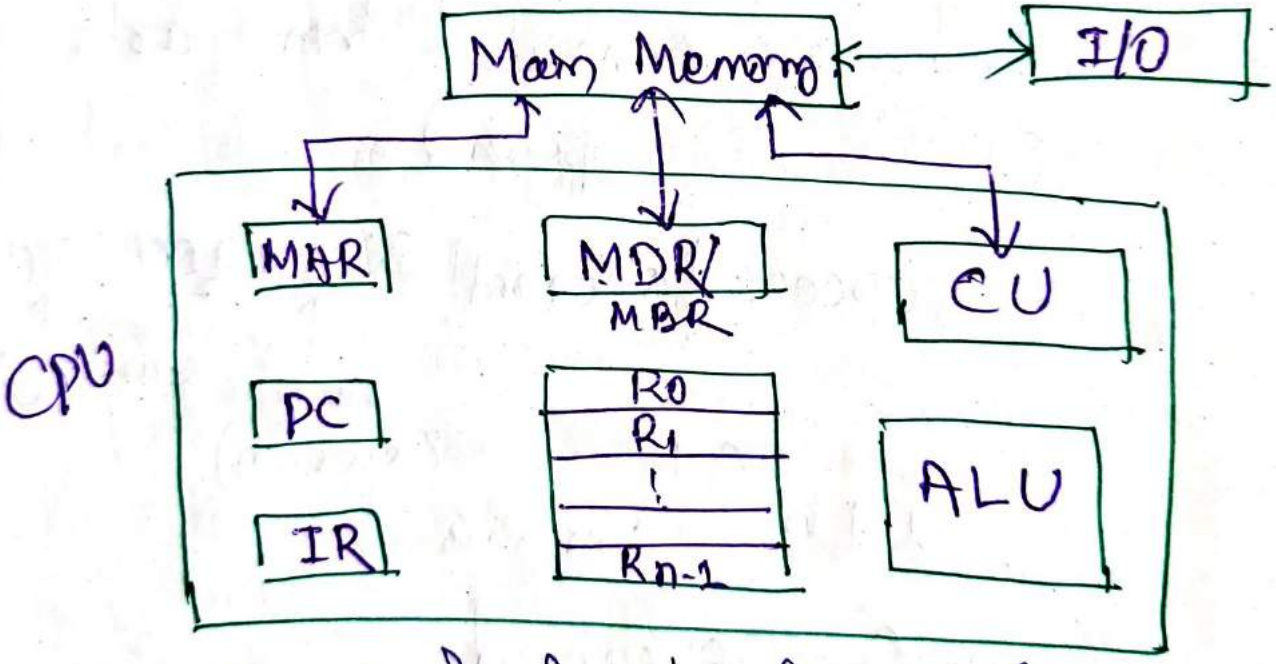
2) Fetch the operand (Add) then the Content of Register (R0) is performed. (Execution)

3) Store the result in R0 register for output.

Ex A = 6
R0 = 2 8
R0 = 8

Ex!- Add LOC A, R0
Add 6, 2
Print R0

Connection between processor and memory



-; so the instruction is fetched from main memory by CPU.

→ After fetching the instruction is decoding on machine language (0 & 1) opcode.

→ Then it start executing for provided result.

IR! - Instruction Register
PC: Program Counter.
→ It keeps tracks of the the instruction which wants for execution of the program and it Contains memory address of the next instruction.

MDR/MBR:- Memory Data Register / Memory Buffer Register.

a=6 (value) &a = 101 → address of a

→ R_0 to R_{n-1} (General purpose register) which is used to write the instructions of a little bit Code.

Instruction format

→ The instruction formats vary depending on the requirments like length of instruction, number of fields & number of bits in each field differs in various systems but all fields called opcode.

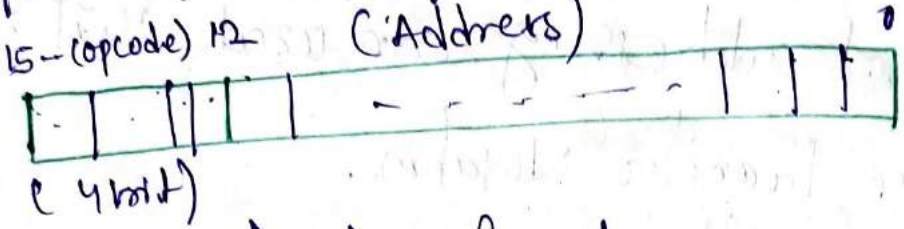
→ Structure and format of other fields depending on the type of Computer architecture.

→ these fields determine the flow of data between registers, memory and input/output peripherals.

3rd Sem 05/08/2020 Sub:- TH1(CSA)

Continue from instruction format .....

→ The Computer System presented 3 instruction formats the instruction register is of 16 bits, the opcode has 4 bit and the remaining bits depend on the type of instruction.



Types of Instruction format

1) Memory reference instruction:- It uses 12 bit @ address to specify the address of data. Bit 0 for direct address and Bit 1 for indirect address.

2) Register reference instruction:- It having opcode has value 111 and 0 in the leftmost bit (4-bit). Range start from 0111 first 4-bit of instruction in hexadecimal value as 7 (numerical form). The Other bit position in the instruction set that position here to be taken.

3) Input/Output:- the value as 111 in Opcode and 1 in the leftmost bit of IR, the remains bit of IR used to specify the Address of I/O instructions.

Register Transfer Notation (RTN) Vs Assembly Language Notation (ALN)

→ **Notation** :- It is the way to represent of different entities involved in an instruction. So the entire registers, memory location, I/O port addressing are used, that is called Register Transfer Notation.

→ The other opcode are used in assembly language notation.

Ex:- $R_1 \leftarrow R_2 + R_3$ (Add contains of R_3 and R_2 then assign(store) the result in R_1)
(RTN)
$R_1 \leftarrow R_2 + R_3$

Ex2 (ALN) $R_1 \leftarrow [LOC]$ (transfer the content of the location 'LOC' to R_1 register)

Opcode

Ex:- MOVE A, #10 (Copy of 10 in to accumulator 'A')
ADD B (add the content of B register to accumulator.
STA LOC (Store the result to LOC accumulator)

N:B:- MOVE, ADD, STA are the assembly lang. notation which are called "mnemonics" which tells the type of operations and A, B, are the general purpose register. and LOC is the location in the memory.

Basic Instruction types :-

→ A standard look at an instruction, defined in operation in a program.

Operation Source, Destination

Addition! -

ADD A, B
↑ ↓ ↳ Destination register.
Operation Source register

Subtraction SUB A, B

→ The instruction perform an ADD operation on the content A and B and the SUM value is stored in register 'B'. |B ← |A| + |B|

Ex!- LOAD A → A is used to as a source
ADD B → B add to the content A and Sum is stored in A
STORE C → The result in A will be transfred into C as a destination.

→ Instruction using general purpose register
let R_i → GPR so LOAD A, R_i
STORE R_i, A

→ The content at the accumulator is loaded into R_i and in the 2nd instruction, the content of R_i is stored into A.

Ex:- $C = A + B$ can be represented as

```
MOVE A, Ri
MOVE B, Rj

ADD Ri, Rj

MOVE Rj, C
```

A, B, C, R_i, R_j are the general purpose register.

$A \rightarrow R_i$

$B \rightarrow R_j$

$R_j \leftarrow |R_i| + |R_j|$

$C \leftarrow R_j$

3rd Sem 07/08/2020 Subj:- TH1 (CHA)

2.2) Operand

→ An operand is the part of a Computer instruction which specifies what data is to be manipulated or operated on, at the same time it representing the data itself.

Ex:- a (+) b ADD a=6 b=-5

operand

container

→ A Computer instruction describes an operations specifies as Addition, Subtraction etc while operand specifies the value to be operated of a variable.

→ In assembly language; an operand is a value on which the instruction is operated, the operand may be processor register, memory address of general purpose register.

Ex:- 32 bit / x86 architecture.

MOVE D, A value of 'A' operand (accumulator) is stored/moved to the register D (General purpose register).

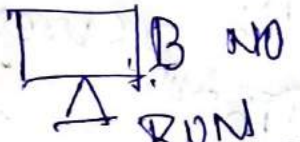
2.3) Opcode (Operational Code)

opcode | operand ML:- 0/1

→ An opcode is the portion of a machine language (instruction) that specifies the operation to be performed.

→ Most instructions also specifies the data they will process in the form of operands.
→ the opcode ceres the instruction set archi. that architecture varies from machine to machine.

Ex!- 

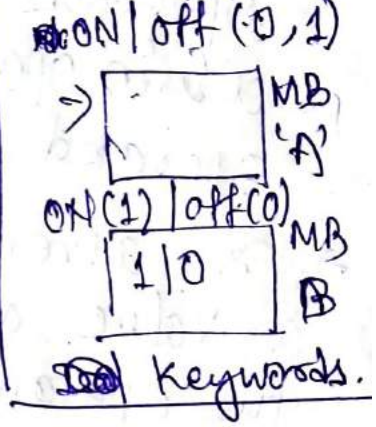


NO OS
Win7(SW)

→ the opcode can also be cered in abstracts hardware computing machine as a part of bytecode specification.

→ The opcode is differs to every Computer lang (C, C++, C#, jav),

→ The every Comp. lang. the CPU process that adopt code by using opcode table.



Opcode for assembly language

→ It is a middle level as well as low level lang. which operated by some specific assembly code.
→ It has a set of code (instruction set of CPU).
Ex!- Intel 8086 processor instruction set.
Arithmetic Code!- ADD, SUB, MUL, DIV
Syntax:- operation source, destination
Data Transfer!- MOV, POP, IN, OUT
Logic Code!- AND, OR XOR, NOT
Jcomp code:- JMP, JZ etc.

Execution of programs (Instruction Set)

→ The assembler scans the whole source code and keep tracks of all name and the numerical values that related to the symbol table so when a name appears second time (destination) it is replaced with its value from the table.

→ The loader is a program which loads opcode for operation inside the memory for executing next instruction.

MOV A, #10

ADD A, B

|B| = |A| + |B|

→ the assembler stores the object program on magnetic disk. the object program must be loaded into memory of computer before it is executed.

3rd Sem 08/08/2020 Sub:- TH1(CSA)

2.4) Instruction formats

-> Computer perform task on the basis of instruction.

-> the field in computer is consists of 0 and 1, so each field has diffent significant. on the behalf of CPU is decided what to be performed.

The most common fields

* Operation:- It specifies the operation like addition etc
* Address:- It contains the location of operand i:e register or memory location
* Mode:- It specifies how operand is to founded.

On the basis of address, instruction are classified by ~~5 types~~ 4 types:-

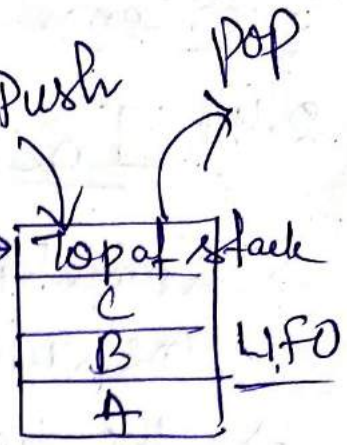
1) Zero Address Instruction
2) One Address Instruction
3) Two Address Instruction
4) Three Address Instruction

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

1) Zero Address Instructions

→ A stack based Computer do not use address field..

→ To evalute an expression first it is it converted to reverse polish notation. i.e post fix notation.

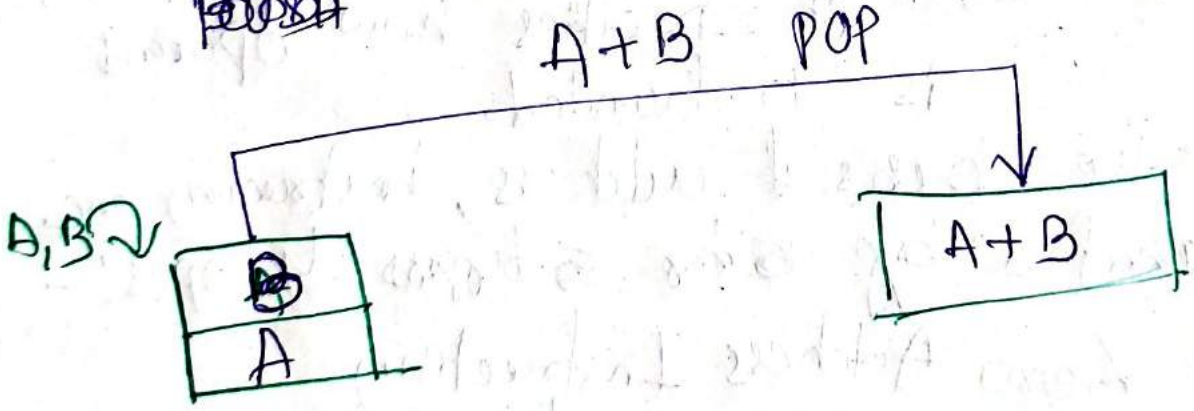


Expression :- $X = (A+B) * (C+D)$

Postfix :- $X = AB+CD+*$

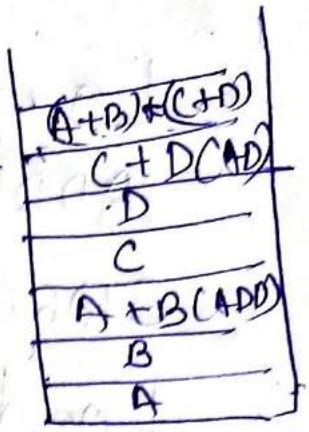
TOP :- top of stack

M[X] :- It is any memory location



Push A
Push B

Push	A	TOP = A
Push	B	TOP = B
ADD		TOP = A+B
Push	C	TOP = C
push	D	TOP = D
ADD		TOP = C+D
MUL		TOP = (C+D) * (A+B)
POP	X	M[X] = TOP



2) One Address Instruction

→ This address used of accumulator register for data manipulation.

→ One operand is in accumulator and other in any other register or memory location.

→ that means the CPU is already known one operand which is in accumulator so there is no need to specification.

opcode	operand/addr of operand	Mode.

Exp:- X = (A+B) * (C+D)

AC = Accumulator

M[ ] = is any memory location

M[T] = is temporary location.

Load	A	AC = M[A]
ADD	B	AC = AC + M[B]
STORE	T	M[T] = AC
Load	C	AC = M[C]
ADD	D	AC = AC + M[D]
MUL	T	AC = AC * M[T]
STORE	X	M[X] = AC

3) Two Address Instructions

→ In this instruction there are two address can be specified in the instruction

→ But here the result store at a different location but it require more number of bits to represent the address.

opcode	Source address	Destination address	Mode

Exp:- X = (A+B) * (C+D)

R_1, R_2 are register.

M[] :- any memory location.

MOV	A, R_1	R_1 = M[A]
ADD	B, R_1	R_1 = R_1 + M[B]
MOV	C, R_2	R_2 = C
ADD	D, R_2	R_2 = R_2 + D
MUL	R_2, R_1	R_1 = R_1 * R_2
MOV	R_1, X	M[X] = R_1

4) Three Address Instruction

→ In this instruction address the address field to specify a register or a memory location.

→ Program created are much short in size but number of bits per instruction increase.

→ These instruction makes creation of program much easier but doesn't mean the program. Contain more information will be performed in one cycle only.

Opcode	Source	Source	Destination	Mode

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

R_1, R_2 are register

M[] :- is any memory location.

ADD A, B, R_1 $R_1 = M[A] + M[B]$

ADD C, D, R_2 $R_2 = M[C] + M[D]$

MUL R_1, R_2 X $M[X] = R_1 * R_2$

Q) What is an instruction format? Define one address instruction with example.

3rd Sem 10/08/2020 Sub:- TH1(CSA)

2.5) ADDERESSING MODE

-> Addressing mode refers to the mechanism that employed for specifing operands.

Ex:- ADD A, B | let A=10, B=20

↓ operand

|B| = |A| + |B|

B = 30

&A = 102

&B = 103

Index	value
100	3
101	6
→ 102	10
103	20

-> operand specifications can be done by three concepts / classification.

* Part of the instruction A → B → C (location)
* Refrese to the memory location
* Address of CPU Pregister.

Various types of addressing modes

1) Immediate Addressing
2) Direct Addressing
3) Register Addressing
4) Indirect Addressing
5) Indexed Addressing
6) Auto Increament addressing
7) Displacement Addressing.

Why so much address is needed?

→ The basis of an addressing mode we have chosen! -

* Efficiency → Time Complexity. (best / worst)
* Economy.
* programming flexibility.

Ex:- A register addressing is efficient in terms of both time and space.

Ex MOV R_1, 200 | value of 200 is moved to the register R_1

1) Immediate Addressing

* Initializing variables.
* No additional memory is required.
* Operand is mentioned explicitly.

Ex:- MOV R_0, 300

CPU Register. ↳ Initialised value

The value of 300 is moved / stored to the register R_0 directly.

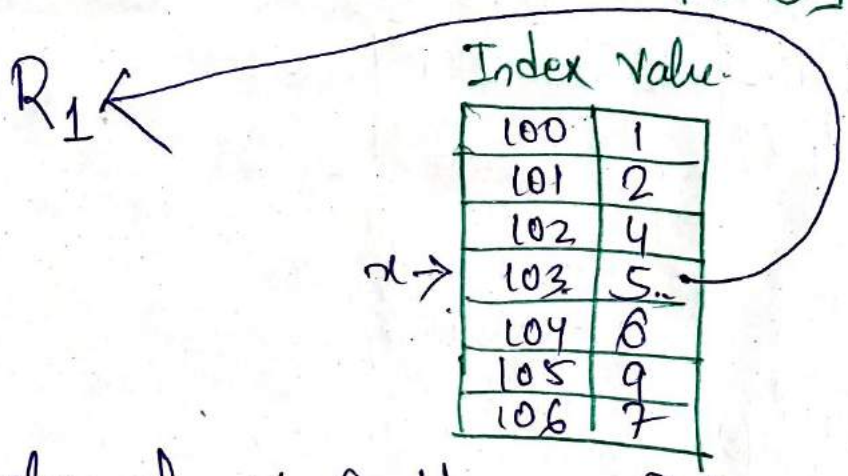
2) Direct Addressing Mode

→ The address of memory location is given explicitly.

→ The contents of the memory location is provided.

Ex:- MOV R_1, x | x = 103

↙ CPU Register. ↳ Memory location [content = 5]



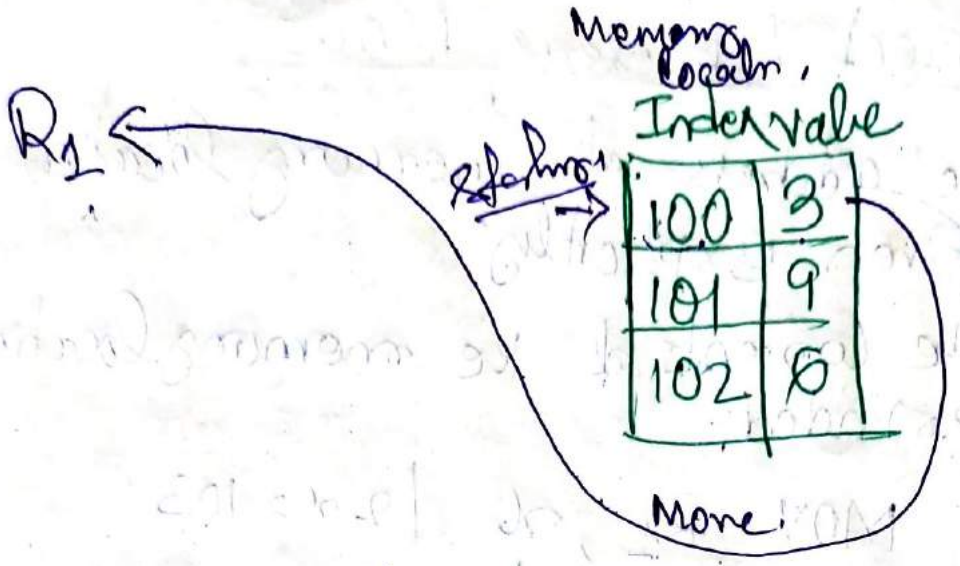
value of x (address 103) is moved to the CPU register R_1

3) Register Addressing

→ Instruction specifies the address of the register.

Ex:- MOV R_1, 100 → Address of the register

↙ CPU Register. ↳ Memory location (content)



(Register Addressing)

3rd Sem 12/08/2020 Sub:- Th1(CSA)

Continue from "Addressing Mode"....

4). Indirect Addressing

(variable) a = 10
(variable, value/data, Content)

Pointer (addr of another variable)
*P=&a;

-) In this type of addressing mode the instruction gives the address where the effective address (EA) is stored in memory.

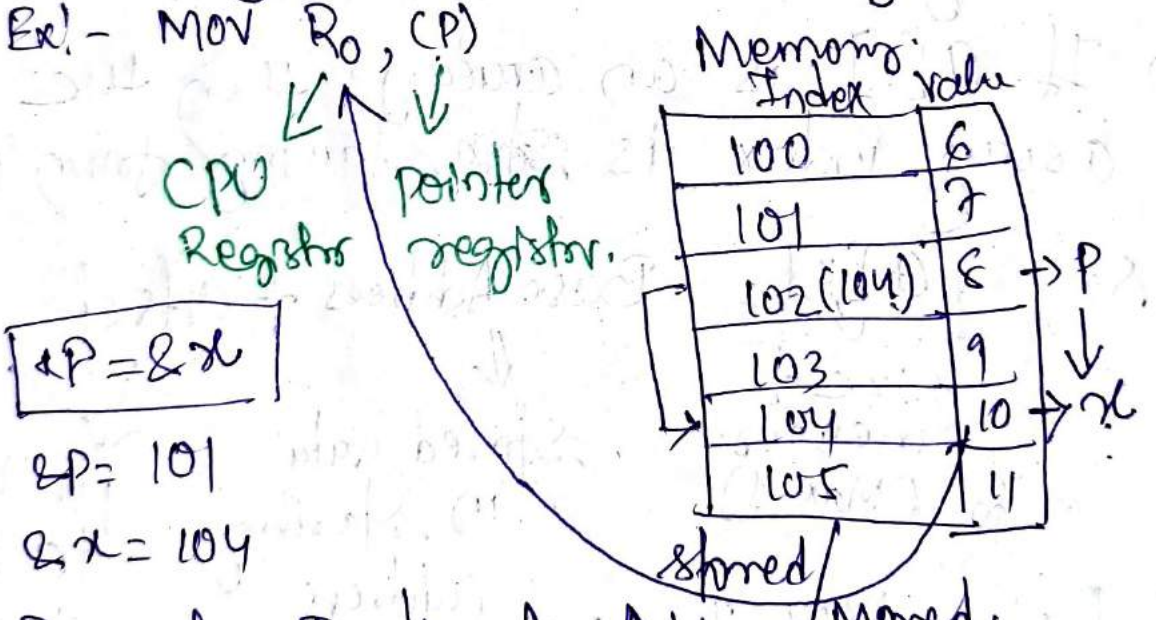
-> Address of variable is stored.

-> The variable contains the Content(data).

-> It is two types! -

a) Memory Indirect Addressing

Ex!- MOV R0, (P)



*P = &x

&P = 101

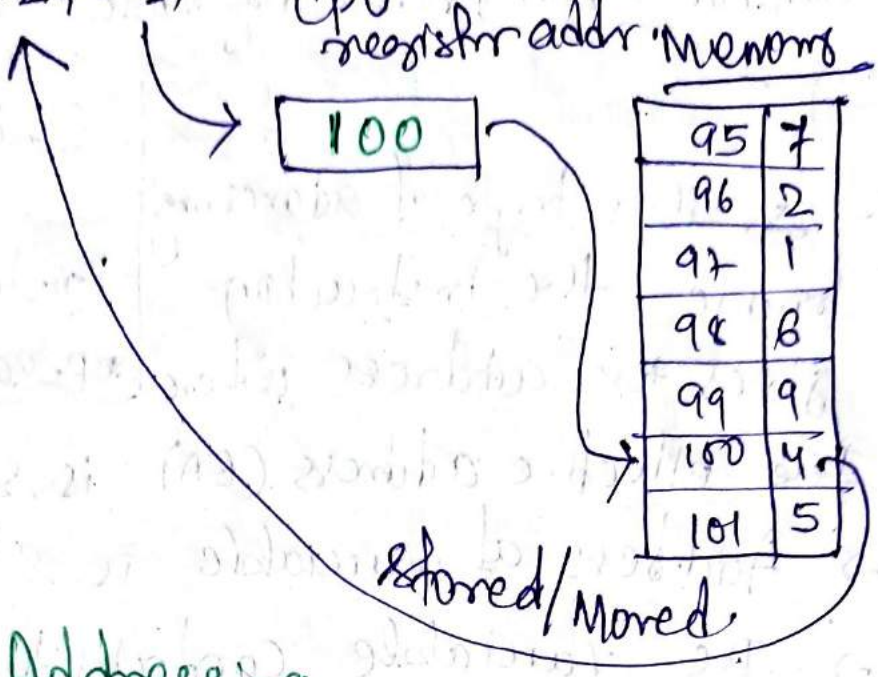
&x = 104

b) Register Indirect Address

-> In this addressing, the use of CPU register so no need any memory.

-> Address stored in CPU it access the address to get the Content(data).

Ex MOV $R_2, (R_1)$



5) Indexed Addressing

→ This addressing mode is used an array element are accessed.

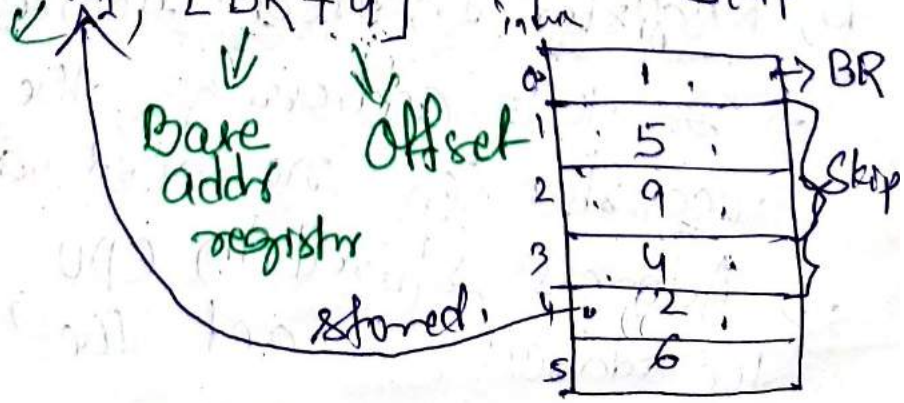
Array index - 0.

→ If A[.] is an array then the array index is starting from '0'.

So $A[i] = \text{Base Address} + \text{offset}$

- $A[i]$: size of memory. 0 to (Max-1)
- Base Address: Stored data in starting address
- offset: position to be skip

Ex:- MOV R_1, [BR+4]



6) Auto Increment Mode

→ In this addressing mode the content (data) are incremented by register increment.

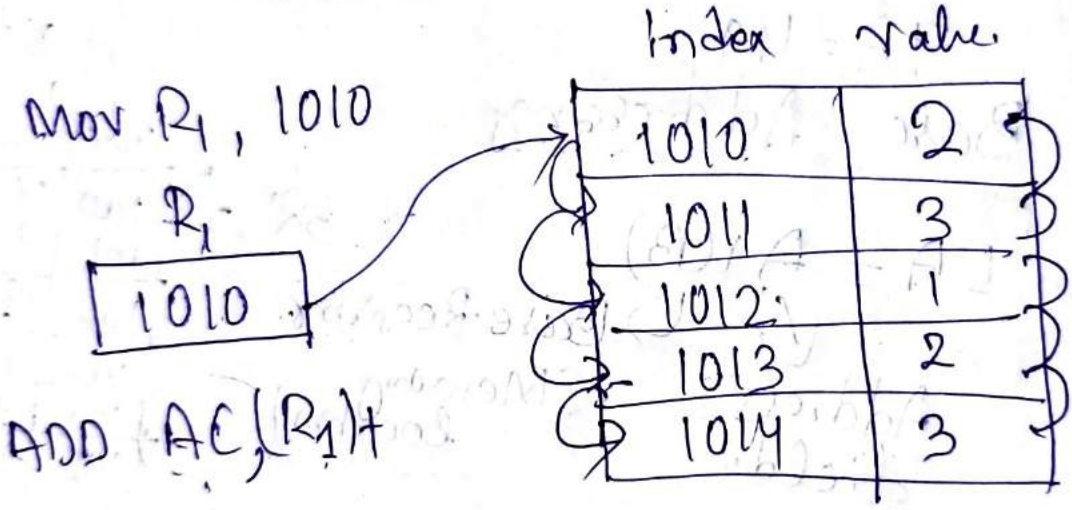
→ which is denoted by (R)+

Ex)- MOV R_1, 1010

ADD AC, (R_1)+ = { ADD AC, (R_1) ; $R_1 \leftarrow R_1 + 1$ }

a = b+c ; c = c+1

AC → Accumulator Register



$R_1 = 2 + 3 + 1 + 2 + 3 = 11$

7) Displacement Addressing Mode

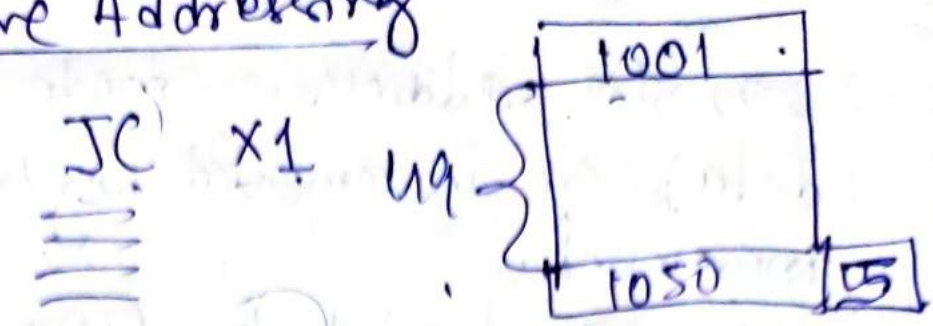
→ Direct addressing and register indirect addressing.

→ It is a very powerful addressing mode.

→ It has three types! -

a) Relative Addressing

Ex) - 1001 JC X1



1050 X1: ADD R_1, 5

Target Address = Ref. Addres - EA

TA = 1050 − 1001 = 49

TA = 49

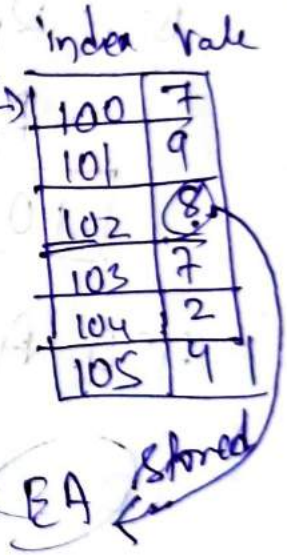
b) Base Addressing

EA = A + (B)

A → Address field.

B → Base Register (Memory location)

Data = EA



c) Indexing

EA = A + (R)

A → Address field. (Base Address)

R → Register (offset)

→ It has stored/access data in an array which is in sequential.

3rd Sem 14/08/2020 Sub!- TH1 (CSA)

Unit-3 Processor System

3.1) Register files

→ The register files is nothing but the registers within CPU is called register files.

→ Register files is an array of processer register in CPU.

→ The register file is the highest level of memory hierarchy.

→ It is implemented over SRAM (Static RAM) with multi-ports. (Write/Read).

→ RAM having read and write ports, where Ordinary multiports SRAM will usually read and write with the same port.

There are two type of registers

1) User-Visible register
   - i) General purpose register.
   - ii) Data Register
   - iii) Address Register.
   - iv) Conditional Code Register.
2) Control and Status Register.

1) General purpose Register:-
-> It can contain the operand for any opcode also it is used for specific operations like floating point, and stack operation.

2) Data Register:- It is used to hold data and cannot be used the calculation of an operand address. (Accumulator)

3) Address Register:-
-> It holds the address of variable.
-> It is divided by 3 types such as

i) Segment Register:- It holds the base address of a segment.

ii) Index Register:- It is used for indexed register addressing or may be auto-indexed addressing.

iii) Stack pointer:- It used to point the top of the stack.

4) Condition Code Register:-
-> It is used to the set of bits codes by the CPU hardware for the result of an operation.

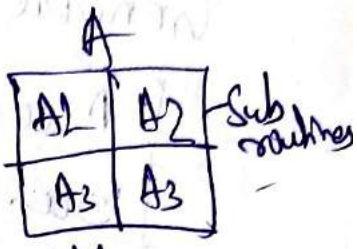
Ex:- a>b a=6, b=12
b is greater.

2) Control and Status register (flag).

→ It is not visible to the user.

→ It may be visible but in a control at Operating System.

→ Machine instruction allows these bits (flag) to be read by implicit reference with the programmer can't use them



→ In some machines, Sub-routines Calls will get result automatically

→ So it store values and returns inform.

→ Set of individual bits (flag) - result for the last operand

→ The value of flag i.e zero(0).

→ Register Essential to Instruction Execut

→ PC (program Counter)

→ IR (Instruction Register)

→ MAR (Memory Address Register) → Connects the BUS.

→ MBR (Memory Buffer Register) →

→ It Connects to data bus, feeds it the other register.

Example of Register Organisation:-

Ex:- Intel 8086

General Purpose Reg.

Pointer and Index

Segment

3rd Sem 17/08/2020 Sub:- TH1(CSA)

3.2) Complete Instruction Execution

Bare fundamental Concepts! -

→ The primary function of a processor System (CPU) is to execute the sequence of instruction stored in a memory, which is external to the processor System (I/O devices).

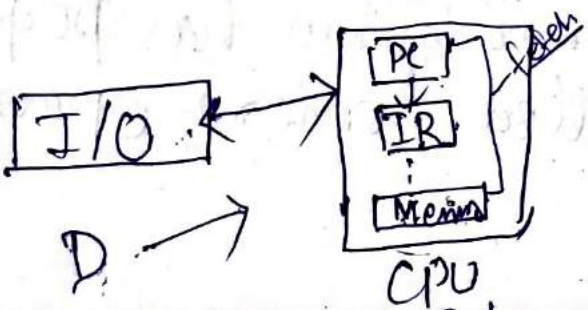
→ The Sequence of operations involved an instruction Cycle, which can subdivided into three major phase such as! -

1) Fetch Cycle 2) Decode Cycle
3) Execution Cycle.

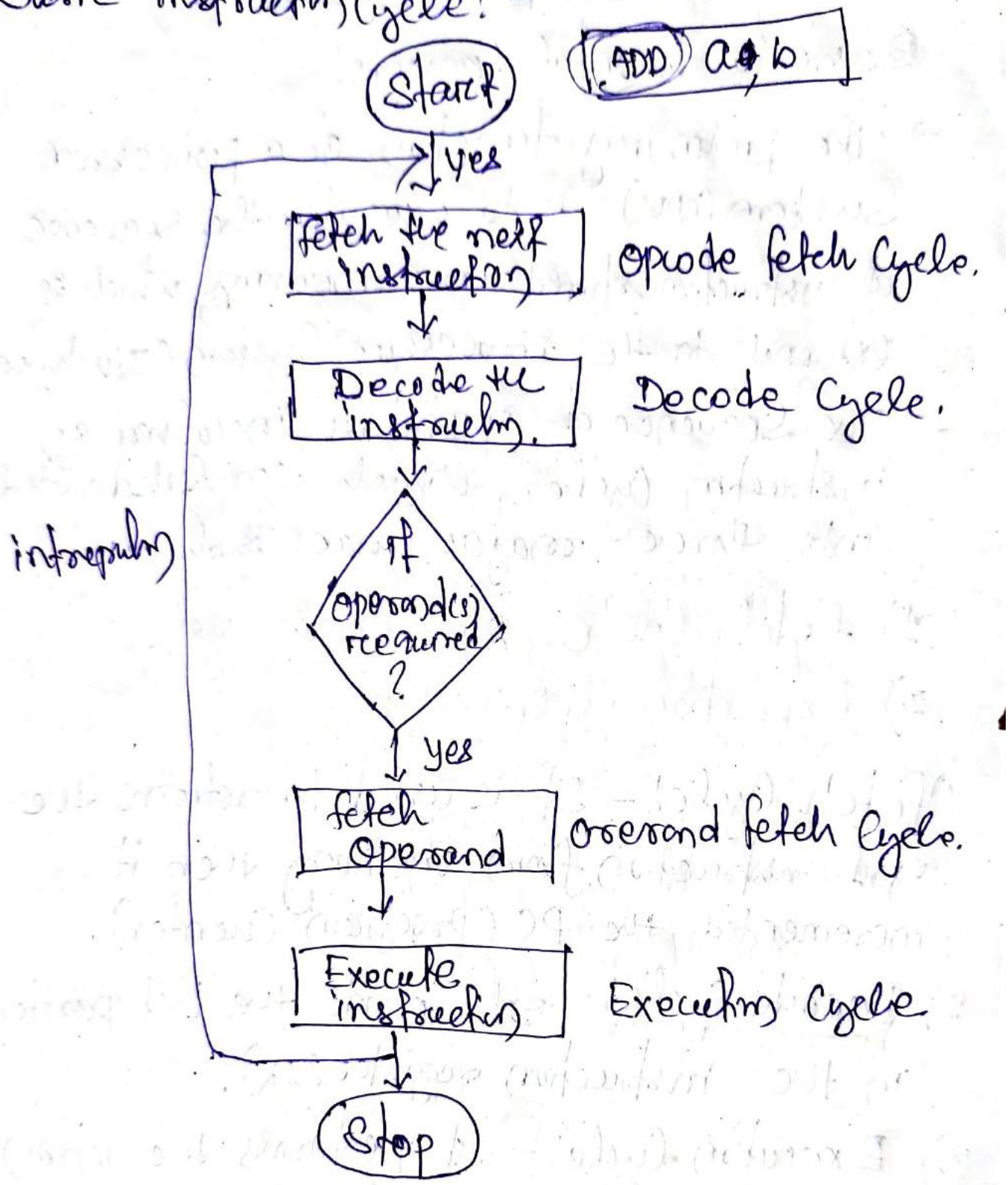
1) Fetch Cycle! - It is used to retrive the next instruction from memory then it incremented the PC (Program Counter).

2) Decode Cycle! - It decode the bit pattern in the instruction register (IR).

3) Execution Cycle! - It performs the action requested by the instruction in the IR.



Basic instruction cycle! -



-) To perform fetch, decode and execute cycle the processor system has perform set of operations called micro-operations.

Data (Source/Bshop-3 (Destination) → Data BUS
→ Address BUS
→ System BUS

→ Single BUS organisation of processor System shows the building block of processor System are how organised and how they interconnected.

→ They can be organised in various ways in which ALU and all processor register / CPU register are connected through a Single Common BUS.

→ It also shows the external memory bus are connected to the Memory Address Register (MAR) and Memory Data Register (MDR).

3rd Sem 19/08/2020 Th-1-(C&A)

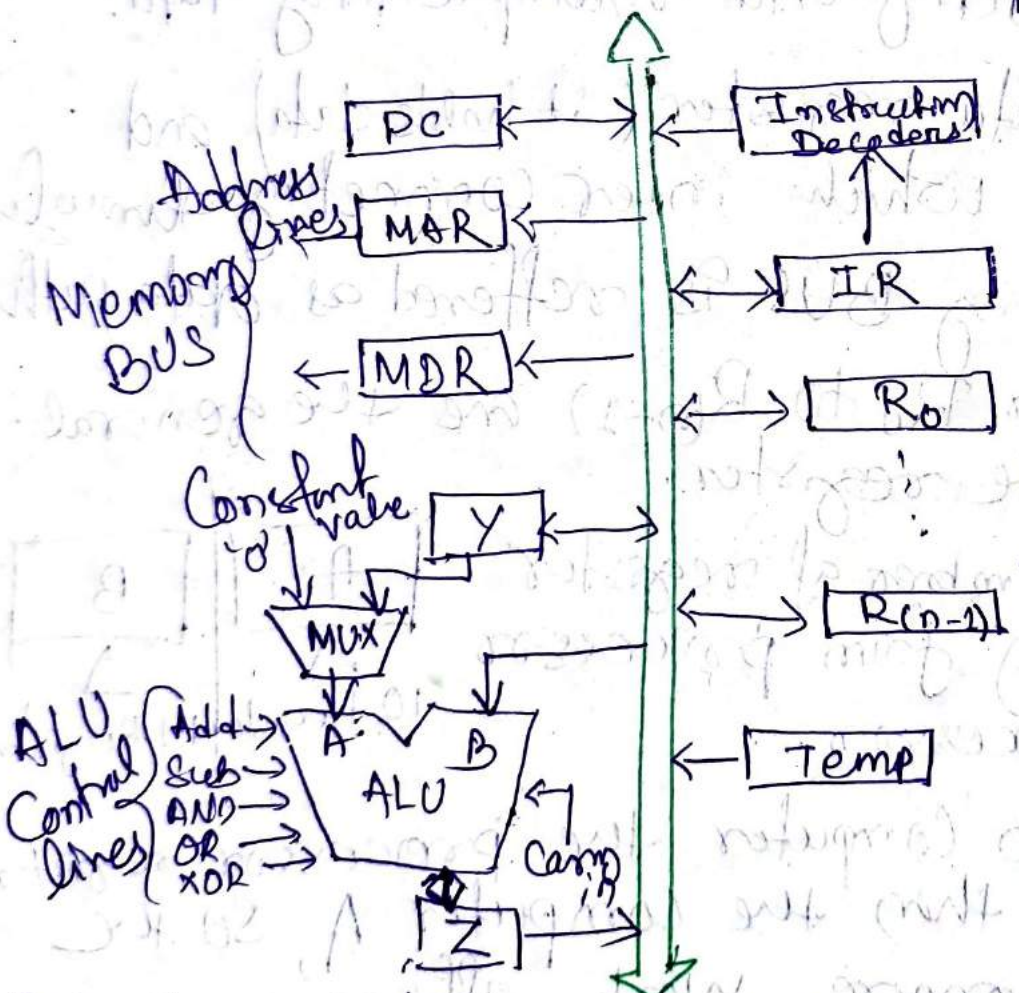
Continue from basic fundamental concepts! — fetch Decode Execute

Single Bus Organisation of Processor

→ The communicating from input to output in the CPU, use the system BUS, where the data should travel from input device, then goes to the output devices.

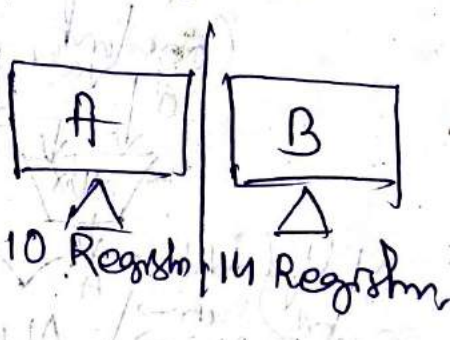
→ The BUS is the medium or path where data are travelled.

CPU Bus



→ The register Y, Z and Temp are used by the processor so it is used as a temporary storage during the execution of instruction.

→ It is not used to store data and only one instruction store for execution.
→ The programmers cannot access these registers.
→ The IR and Instruction decoder are the integral parts of Control Unit of Processor System. (CPU).
→ All register and ALU are used for storing and manipulating data.
→ The data register (it holds data) and ALU which inter connecting through Connecting BUS is reffered as data path.
→ Register R_0 to $R_{(n-1)}$ are the general purpose register.
→ The number of register is vary from processor to processor.



Ex)- In B Computer the processor register is 14 than the computer A, so the B is more Work efficiency than A.
→ The registers such as general purpose, Special purpose, Stack pointer, index register and pointers.

↳ There are 2 options for A input of ALU i.e from MUX and ALU control lines

→ The MUX (Multiplexer) used to select on of two input i.e Y or constant (1 or 0).

The execution of instruction is following basic operations

a) Transfer of word of data from one processor register to another to the ALU.

b) Then perform the arithmetic and logic operation has data and process register

c) the fetch a word of data from specified memory location and load them in to a processor @ register.

d) Store a word of data from a processor register into a specified memory location.

Ex:-
```
MOV R1 20
MOV R2 50
ADD R1, R2
```

3rd Sem 21/08/2020 Sub:- TH1(CSA)

Execution of Complet Instruction:-

Let us find the Complet Control sequence for execution of an instruction i.e ADD $R_1, (R_2)$ for the Single bus processor.

→ This instruction ADD operation, the contents of memory register R_2 and the contents of register R_2, then the result is stored on R_2.

→ To execute bus instruction it is necessary to perform the following actions:-

1) Fetch the instruction.
2) Fetch the operand from memory location which is pointing R_2 and decode by machine code.
3) Perform the addition operation
4) Store the result in R_2

Ex!- A program for $C \leftarrow [A] + [B]$

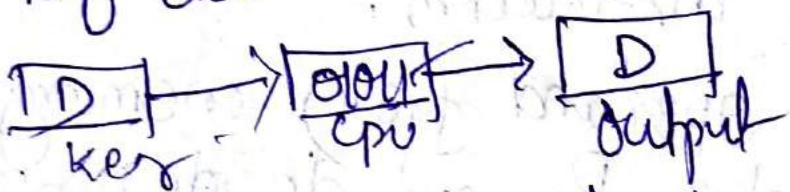
MOVE A, R_0	MOVE 10, R_0
ADD B, R_0	ADD 15, R_0
MOVE R_0, C	MOVE R_0, C

R_0: 10 + 15 = 25

C: 25

control signal to access the byte (1st byte) which is stored in a location that is transfred to IR (instruction register).

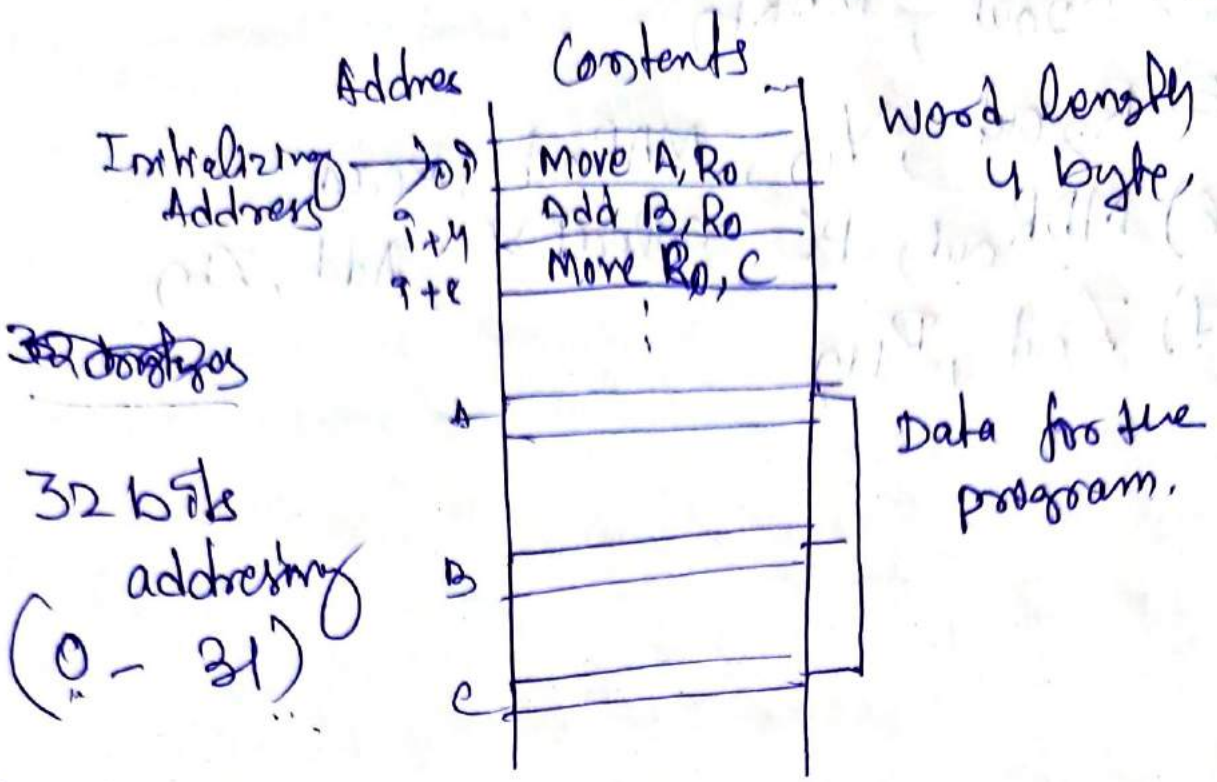
Step3:- The fetched byte then is decoded to calculate the number of operand (data) by associated with an instruction.



Step4:- Now the Content of Program Counter (PC) is incremented by 'i' (index).

4 byte address (incremented by PC)

PC ← Initial address (i.e) 0
PC ← PC+4 (i.e i+4) 4
PC ← PC+4 (i.e i+8) 8
PC ← PC+4 (i.e i+12) 12



Step-5:-

Now the bytes are fetched and the final executing of the instructions is done. Then the result is stored at the desired location is (C).

→ So this phase is called instruction executing phase/cycle.

→ The Sequence of control steps required to perform these operations for single bus architecture are as follows:-

1) PC_{out}, MAR_{in}, Y_{in}, Select C, Add Z_{in}

2) Z_{out}, PC_{in}, MAR_{in}, MAR, M, Read.

3) MDR_{out}, PC, MAR_{in}

4) R_{2out} , MAR_{in}

5) R_{2out} , Y_{in} , MAR_{out} , Read

6) MDR_{out}, PC , Select Y , Add , Z_{in}

7) Z_{out} , R_{1in}

Sub:- TH (CA) 24/08/2020 @ 3rd Sem

Branch Instruction

Comp A > B

```
Move A, R0
Move B, R0
Add  R0, R1
Move R1, C
```

(PC)

→ The branch instruction loads the branch target address in Program Counter (PC).

→ So that the PC will fetch the next instruction from the branch target address.

→ The branch target address is usually obtained by adding the offset (T/F) is the content of PC.

→ So the offset is specified within the instruction.

→ The control sequence of conditional branch instruction is as follows:-

1) PCout, MARin, Read, Yin, Select C, Add Zin
2) Zout, PCin, WMFC
3) MDRout, IRin
4) offset_field-IRout, Select Y, Add Zin
5) Zout, PCin, End

Step 1 to Step 3 :- It fetch of the opcode for operation

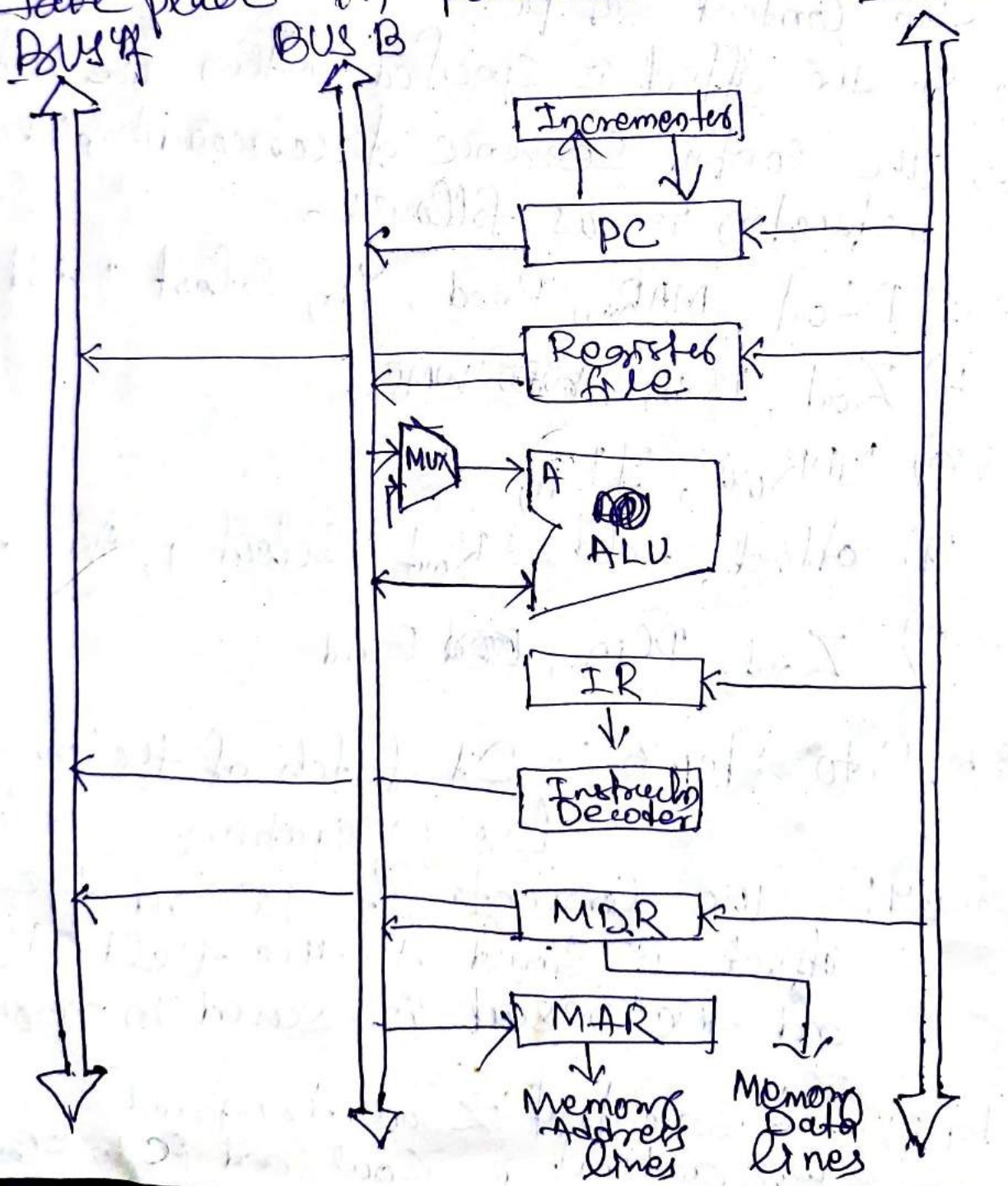
Step 4 :- The contents of PC and the offset is saved to the field IR and the result is saved in register Z.

Step 5 :- The contents of Z are transferred to PC for actualing Zout and PCin Signal.

Multiple Bus Organisation

→ The multiple bus organisation increases the steps required to complete the execution of the instruction.

→ To reduce the number of steps needed to execute instruction, most commercial process provide multiple internal path which increases the several transfers to take place in parallel.



→ The three buses used to connect registers and the ALU of the processor.

→ All general purpose registers is used to process by a single block called register file.

→ There are 3 ports, one input port, and two ports are used for output port.

→ So it is possible to access data of 3-registers in one clock-cycle, the value can be loaded in one register from BUS "C" and data transfer to the BUS "B" and BUS "A"

→ The Buses A and B are used to transfer the source operand to the A and B input of ALU.

→ After performing ALU operation, the result is transferred to the destination operand over the BUS "C".

→ The "incrementer" is used to increment the contents of PC to fetch the next instruction. a separate unit is provided that is called "incrementer".

→ let us consider the execution of instruction.

Add R_1, R_2, R_3

→ There instruction adds the contents of R_2 and R_3 and store into R_1.

→ So the three bus organization control steps for Add R_1, R_2, R_3 as follows

1) PC_{out}, MAR_{in}
2) MAR_{out}, MDR_{in}, M, Read
3) MDR_{out}, IR_{in}
4) R_2out, R_3out, Add, R_1in

Step1:- The Contents of PC are transferred to MAR through Bus B

Step2:- The instruction code from the address memory location is read into MDR.

Step3:- The code is transfred from MDR to IR made available at A and B inputs.

Step4:- Two operands from R_2 and R_3 are made available at A and B inputs of ALU through Bus A and B.

These two inputs are added Add signal and result is stored in R_1 through bus C.

Instruction Cycle State Diagram

- At the beginning of each instruction cycle was fetched from memory by the processor.
- Then the PC holds the address of an instruction to be fetched next. The PC increments value for next after next instruction still last execution.
- The fetched instruction is loaded into IR.
- Then the IR decode the instruction, the action is takes place.
- The processor interprets the instruction and perform the required action.

Execute Cycle:- In general the actions required in to 4 catagory

1) Processor Memory:- Data may be transfred from processor to memory or vice versa.
2) Processor I/O:- Data may be transferred to or from a peripheral devices by transferring betn the processor and I/O module.
3) Data Processing:- The processor perform ALU operation on data.
4) Control:- An instruction may specify that the seqence of action to be altered.

→ For any given instruction cycle, some states may be null (zero) and other may be visited more than once, so the state diagram as follows:-

1) Instruction address Calculation (Iac):-
→ It determine the address of the next instruction to be executed

2) Instruction fetch! (If)!-
→ ~~Decoded code~~
→ This state reads the instruction from its memory location and send them into the processor.

3) Instruction Operation decoding (IOd):-
→ It analyze the instruction to determine type of operation to be performed and Operand(s) to be used.

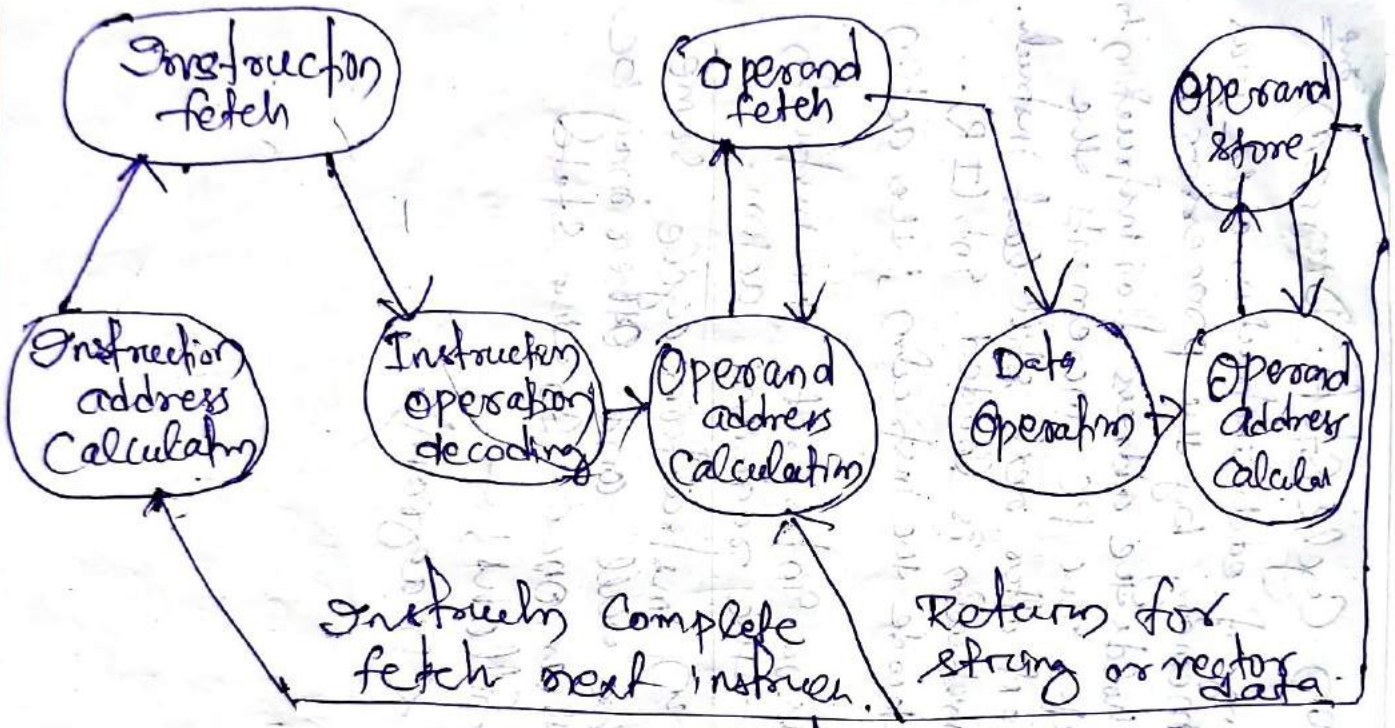
4) Operand address Calculation (Oac):-
→ In this state, if the operation reference to an Operand in memory or available by I/O module then it determine the address of the operand.

ADD R_1, R_2
10 20

5) Operand fetch (Of):-
→ It fetch the operand from memory or read it from I/O module.

6) Data operation (Do):-
→ It perform the operation that is indicated in the instruction.

7) Operand store (OS) :-
→ It is used to provide the result into the memory or out to the I/O module.



(Instruction cycle state diagram)

3rd Sem 28/08/2020 Sub:- Th1(CSA)

Control Unit

→ To execute an instruction, the control unit of CPU must generate the required control signal in proper sequence (I/O).

→ The control unit generates timing and control signal for the operation of the computer.

→ The CU communicates with ALU and main-memory, also between processor, memory and various peripheral devices.

→ A control unit can be implemented by two techniques such as:-
* Hardwired Control * Microprogrammed Control.

3.4) Hardwired Control

→ This control uses digital logic components such as NAND gate, flipflops, decoders and counters etc.

→ The input to control unit are the instruction register, flags, timings, signals etc.

→ The hardwired control can be viewed as a state machine changing from one state to another in every clock cycle, depending on the components of IR, Codition Codes and external inputs.

→ The sequence of the operation carried out by this machine is determined by the wiring of logic elements (gates etc) and hence named as "hardwired" control.

→ Its output of state machine is the control signals that runs various parts of the computer.

* fixed logic circuit that correspond directly to the Boolean expressions are used to generate the control signal.
* Hardwired control is faster than the micro-programmed control.
* A controller that uses this approach can operate at high speed.
* RISC (Reduced Instruction set Computer) architecture is based on hardwire control unit.

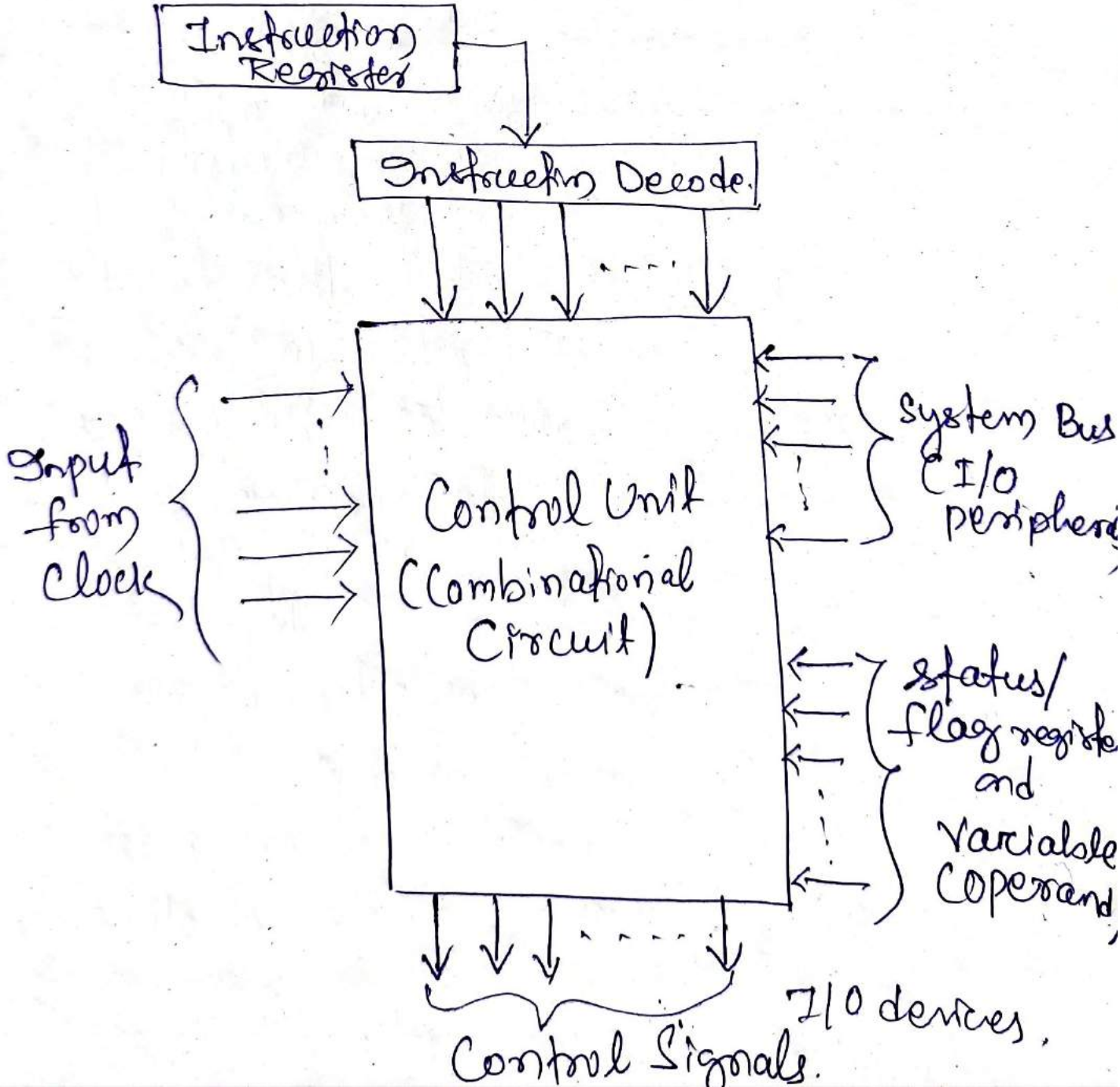
Advantages:-

1) It is faster than the microprogrammed control unit.
2) It can be optimized to procedure the fast mode of operation.

Disadvantages:-

1) The instruction set, the control logic is directly implemented.
2) It Complex decoders and the sequence logic also difficult.
3) The detection of error is more.
4) It is costlier Control Unit

Diagram of Hardwired Control:-



3.4) Micro Program Control

→ It is implemented by using programming approach.

→ A software approach is called micro-programming.

→ In this instance, a programmer can create a special program called the micro-program to store machine instructions.

→ A sequence of micro operations (230) are carried out by executing a program consisting of micro instructions. (firmware)

Ex:- BIOS program.

→ The program then translates those instruction into 1 and 0 for processing.

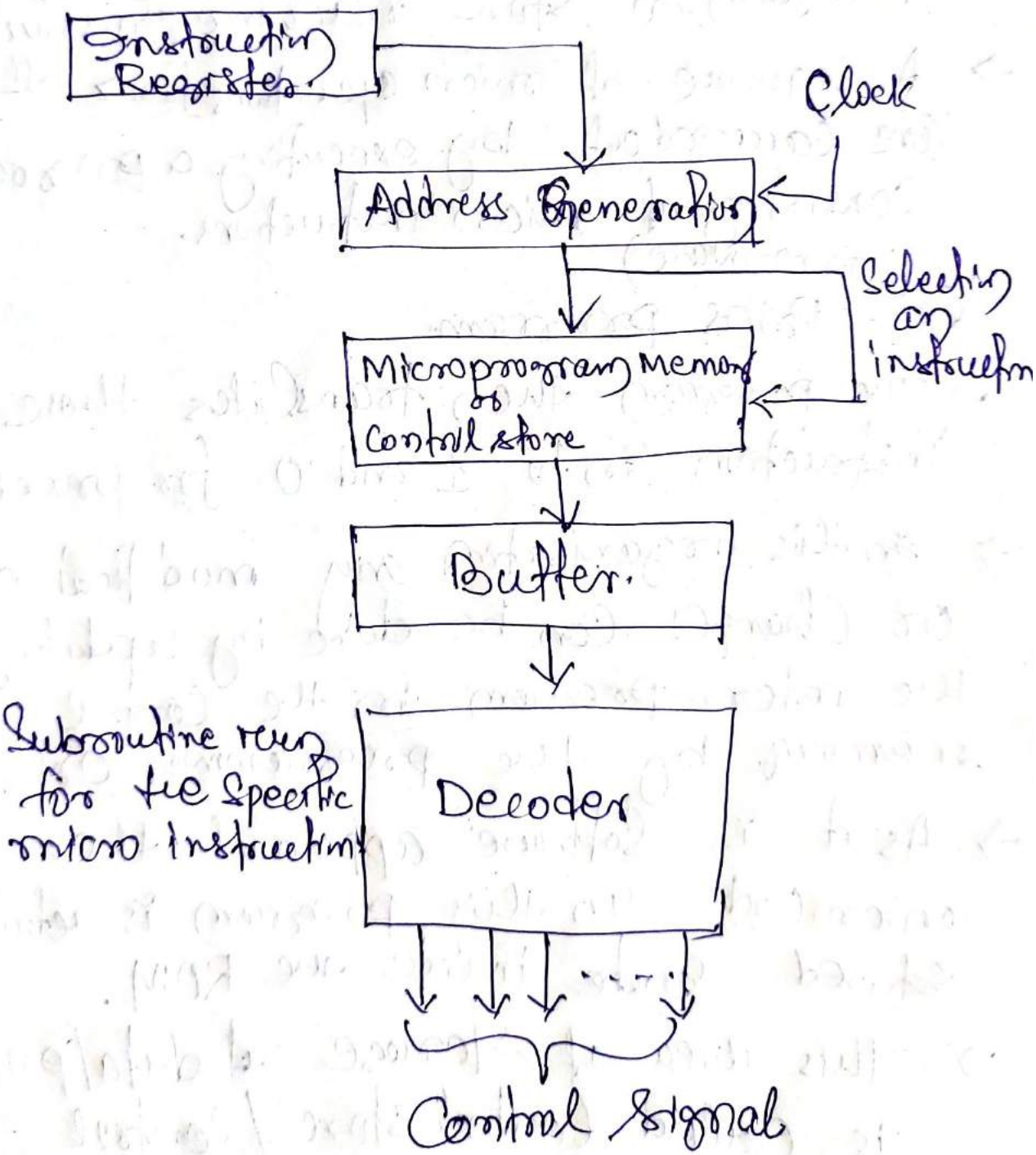
→ In this organisation any modification or changes can be done by updating the micro-program in the control memory by the programmer only.

→ As it is software approach the microcode in this program is actully stored inside the ROM.

→ This area of storage of data/program is called Control store / Control memory.

→ The Control Signal are generated by the program are Similar to ML (Machine language)

→ It is quit slower in speed due to the time takes to fetch the micro instructions from the control memory.



Advantages

1) It is very flexible to accomodate with new instructions.
2) It is both Cheaper and the detection of error is very less
3) Easier to decoding and sequencing can be done,
4) Easier to handle the complex instructions sets.
5) It required a less Chip area.

Disadvantages

→ It is very slow as compare to the hardwired control due to micro-instructions are fetched from control memory so that is time consuming.

Important Questions

Long Questions
5 Marks / 10 Marks

Q1) What is register? Explain all types of register with example.

Q2) Difference processor register and CPU register.

Q3) What is a Complet Instruction Cycle? Explain about fetch, decode, and execution Cycle with proper diagram.

Q4) Differentiate between Single bus and multibus organization.

Q5) Explain the instruction Cycle state diagram.

Q6) Differentiate between hardwired Control and micro program Control.

—— x ——

3rd Sem 31/08/2020 Sub:- TH1(CSA)

Revision Class

Unit-3

Processor System

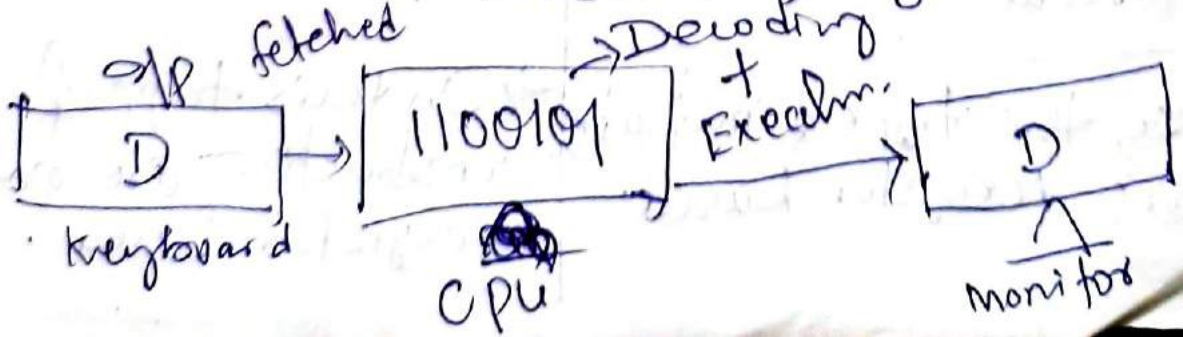
Register Files (CPU Register)

1) User Visible Register.
   a) General purpose register.
   b) Data Register
   c) Address Regiser.
   d) Conditional code register.
2) Control and Status Register
   → flag (1/f) (0/1), (Y/N)

```
n=1;
while(n<=10)
{
  if(n%2==0)
  printf("%d is an even no", n);
  n=n+1;
}
```

A program to find out all even no bet 1 to 10
n=1 to 10
even { n%2==0)

3.2) Complet Instruction execution :-
Basic fundamental Concept
↳ fetch cycle
↳ Decode cycle.
↳ Executing Cycle.



→ Single bus organisation of processor.

→ Execution of Complet instruction, Branch instruction with example.

→ Multiple Bus organization.
Bus A, B & C.

→ Instruction cycle state diagram.
7 state.

→ Control Unit
3.3) Hardwired Control
3.4) Multiprogram Control.

0) Hardwired Control	Micro program Control
→ In this control unit the technology is Circuit based.	→ In this type of control unit the technology is Software approach.
→ It is implemented through flip-flop, logic gates, NAND, decoders, Channels etc	→ Micro instructions generates control Signal to control the execution of instruction.
→ In this control unit has fixed instruction set format.	→ It is variable instruction format.
→ In this type of instruction are register based.	→ In this type of instruction are not register based.

→ It is in RISC architecture.	→ It is used in CISC architecture.
→ It is faster for decoding.	→ It is slower decoding
→ It is very difficult to modify.	→ It is easily modified.
→ The Chip area is large.	→ The Chip area is less.
→ It is very costly	→ It is Cheaper.

3rd Sem 02/09/2020 Sub:- TH1(CSA)

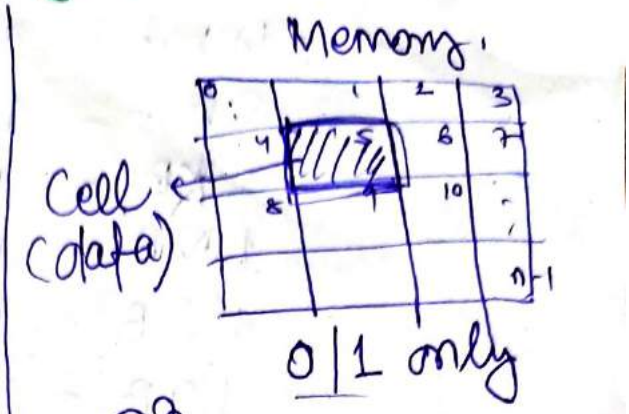
Unit-4 Memory System

Introduction

Data:-

→ It is a collection of raw fact.

→ Text, audio, video, animation etc



→ the memory devices plays an important role in digital Computer System.

→ All program, data, address of variable are stored in memory device.

→ Data are stored in cell area of memory.

→ A cell is the lowest unit of memory area.

→ Memory provides various performance of Computer from booting of process to storing of data for future used.

→ Data are stored in the form of binary i.e 0 and 1 and this is called digital data.

Memory Units

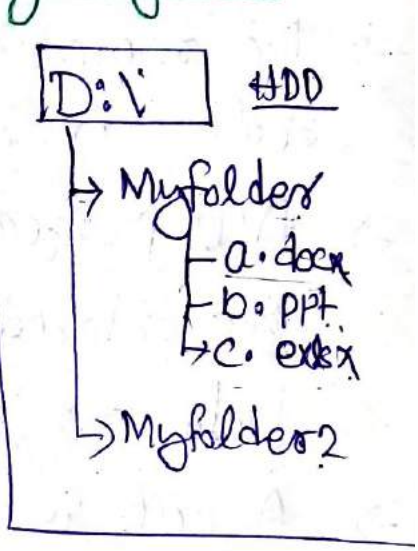
1 bit = 0 | 1
4 bits = 1 Nibble.
8 bits = 1 Byte.
1 KB = 1024 Bytes
1 MB = 1024 KB
1 GB = 1024 MB
1 TB = 1024 GB
1 PB = 1024 TB

GB = GigaByte
TB = Tera Byte.
PB = Petta Byte

4.1) Characteristics of memory System.

1) Location:-

→ The memory like register which is included within the processor and it is termed as processor memory.



→ The location of main memory (RAM / ROM) is resides within the CPU that is called internal memory location.

→ The location of peripheral storage devices such as hard disk, magnetic tape, pendrive, SSD (Solid State Disk) etc. which is accessable to processor by I/O controller is called external memory location.

2) Capacity

1) Number of words:- Common word length i.e 8, 16, 32 bits or bytes.

2) Word size:- Capacity is expressed in the term of words or bytes.

Ex:- the natural unit of organisation is your name, dob, address etc.

String:- It is collection of more than one character is called String.

Ex:- int n=10; Number

↳

Char name[30]= "Balasore";

↙

String/word.

3rd Sem 04/09/2020 Sub:- Th1 (CSA)

Continue from Characteristics of memory sys.

3) Unit of Transfer

a) Internal:-

→ for internal memory, the unit of transfer is equals to the no of data line. in to the memory or out to the memory.

b) External:-

→ for external memory, they are transfer reed on block wise (more than a word)

→ The file size should be in mb, gb etc.

c) Addressable Unit:-

→ It is the small location can have individual address of any variable or register etc.

Ex:- a Sector of a harddisk.

4) Access Method

→ By access of data (read/write) is divided by four types:-

a) Sequential Access
b) Direct Access
c) Random Access
d) Associative Access.

a) Sequential Access :-

→ In this access method, it must start from beginning and reads data one after another till end of file in a sequence basis (linear).

Ex:- Magnetic Tape

b) Direct Access :-

→ In this method there have no specific start location, a user can access files and jumped to another file i.e files (direct access).

→ It is not in a sequence.

Ex:- Magnetic Disk (Hard Disk)

c) Random Access :-

→ In this location can be select out randomly and directly stored or accessed.

Ex:- RAM

d) Associative Access :-

→ This access method has used bit location of memory access for logical operation, matching conditions etc.

5) Performance :-

a) Access Time :- The time was taken to perform read / write operation i.e time taken to address of a memory location and perform read / write from that location.

b) Transfer Rate :-

→ This is the rate at which data can be transferred in and out of a memory.

Ex :- 4mbps - (1s - 4mb data)

Physical Types of Memory

* Semiconductor :- RAM / ROM
* Magnetic :- Hard Disk | Magnetic Tape.
* optical :- CD | DVD
* flash :- Pendrive | Memory Chip etc.

4.2) Memory Hierarchy

→ Memory hierarchy consists of total memory system of any computer.

→ The memory components ranges from higher capacity slow auxiliary memory to a relatively fast main memory like cache memory that can be accessible to high-speed processing logic.

→ The memory hierarchy is depend upon 3 key parameters:-

1) Access Time
2) Storage Capacity
3) Cost

1) Access Time

→ Lower memory like CPU register are CPU local memory which are accessed in few nanoseconds.

→ Cache memory is very-faster accessing speed.

→ Main memory access time is little bit slower than CPU register and Cache memory.

→ Access time of a hard disk is 10 msec. and the access time of optical disk magnetic tape measured in seconds only for data fetched or stored in to them.

2) Storage Capacity

→ The storage capacity increases as we go down to the hierarchy.

→ The lowest storage is cache memory, storage capacity is:- 32, 64, 128 bytes.

→ Next CPU register, storage capacity is- 128 Kb, 256 Kb, 512 Kb etc.

→ Next main memory (RAM and ROM), storage capacity of ROM is:- 4 to 8 mb
storage capacity of RAM is:- 2, 4, 8, 16 etc

→ Next hard disk / magnetic disk and optical disk.
storage capacity of HDD:- 500 GB, 1 TB etc.
storage capacity of CD = 700 mb
storage capacity of DVD is: 4.7 GB

3) Cost

→ Cost is directly proportional with access speed, and vice versa.

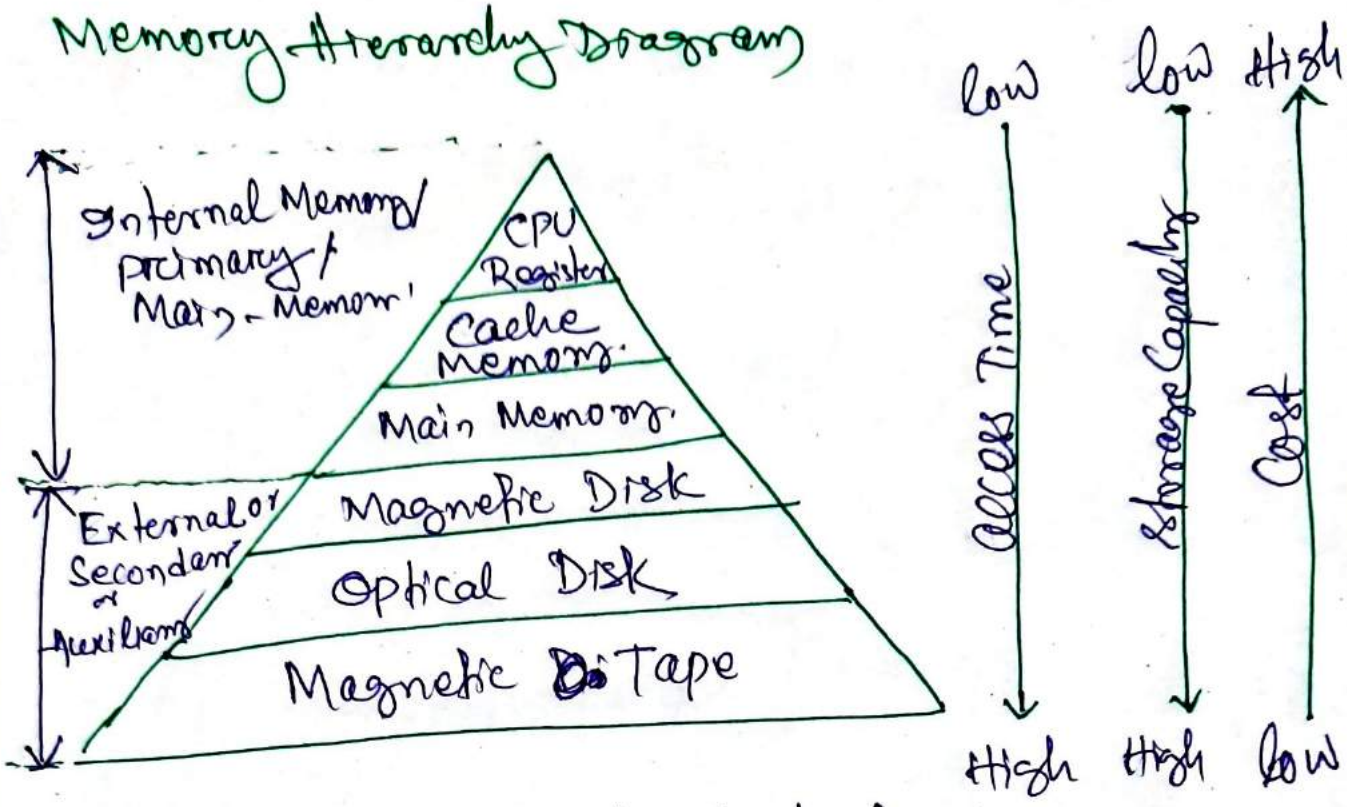
Access speed ∝ Cost

→ lower access speed with higher cost.

→ higher access speed with lower cost.

→ Memory is primarily of two types:
* Internal Memory & External Memory.

Memory Hierarchy Diagram



(Memory Hierarchy System)

Cache Memory

→ It is a very High Speed Semiconductor memory which can Speed up CPU.
→ It is placed in between the CPU and main memory.
→ It is used to holds parts of data or program which is executed by CPU.
→ The parts of data and program are transferred from disk to Cache by operating System than the CPU can access them.
→ It acts as a buffer for temporary use.
→ It is two types L1 Cache & L2 Cache.

Advantages:-

→ It is faster than main-memory.
→ It consumes less access time than main memory.
→ It stored data for temporary use.

Disadvantages:-

→ It has limited Capacity
→ It is very expensive

3rd Sem 07/09/2020 Sub- TH1 (CSA)

Main Memory

→ It is two types such as RAM and ROM.

RAM (Random Access Memory)

→ It is a type of internal memory of CPU for storing data, program and its result.
→ It is a semiconductor type of memory.
→ As it is random access, so access time is independent of the address to the word in each storage location inside the memory.
→ To read or write operation in to RAM is extremely fast.
→ It is quit expensive.
→ It is the category of volatile memory, it means data stored in it is losted when we switch off the Computer or power failure.
→ To maintain power failure, backup System used as UPS (uninterruptible power System) in Computer.
→ Capacity of RAM is 2, 4, 16, 32, 64 GB.

Types of RAM

→ It is two types SRAM and DRAM.

SRAM (Static RAM)

→ This RAM maintains its Content as long as power remains applied.
→ Data is lost when the power gets down due

to volatile in nature.
→ SRAM chips used a matrix of 6-transistors and no capacitors.
⇒ Transistors do not required power to prevent leakage.
→ So SRAM need not have to be refreshed on a regular basis.
→ It is very expensive then DRAM.
→ SRAM used as cache memory to be very fast processing.

DRAM (Dynamic RAM)
→ DRAM needs continuously refreshed to maintain the data on it.
→ The word dynamic means changes, so in DRAM a refresh circuit used to rewrites the data several hundred times per second.
→ It is mostly used RAM due to less cost as compare to SRAM.
→ DRAM are made up of memory cells, these cells are composed of one capacitor and one transistor.

ROM (Read Only Memory)
→ The memory from which we can read only but not write on it.
→ It is a type of non-volatile.

→ The information stored permanently during manufacturing.
→ In ROM Chip BIOS program is stored for startup of a computer when first electricity turned on.

Types of ROM

1) Flash ROM

→ It is also called firmware.
→ This ROM chip used in electronic items like washing machine, refrigerator, AC, microwave oven etc..

2) PROM (Programmable ROM)

→ It can be modified only one by a user.
→ The user buys a blank PROM and enters the content using PROM programmer.

3) EPROM (Erasable and Programmable ROM)

→ It can be erased by ultra-violet light up to 40 minutes.
→ After that again reprogram it by enter the content.

4) EEPROM (Electrically Erasable and Programmable ROM)

→ This type of ROM is programmed and erased by electrically.
→ It can be erased and reprogrammed 10000 times. and time takes 4 to 10 ms.
→ It can be erased one byte at a time. but reprogrammed is slow.

3rd Sem 09/09/2020 Sub:- TH1 (CSA)

Secondary Memory / Auxiliary Memory.

→ It is also called external memory.
→ As per the access mechanism it is two types

① Sequential Access
Ex:- Magnetic Tape

② Direct Access.
Ex:- HDD, Optical Disk Pendrive.

1) Sequential Access

→ In this memory devices data read or write in a sequence.
→ Data can be acess one after another
→ A best example is magnetic tape.

→ Secondary memory refers to storage devices, such as hard disk drive & solid state drive.
→ It also may refer to removable storage media such as USB flash drive, CD's and DVD's.
→ The Secondary memory is not access directly by the CPU. It is accessed from RAM. and then send to processor.
→ RAM plays an important intermediate role, since it provides much faster data access speeds than Secondary memory.
→ By loading software programs and files into primary memory, Computer can process data much more quickly.

→ As Secondary memory is much slower than primary memory so it offers a greater storage capacity.

Ex:- A computer has 500GB HDD with 2GB RAM can process it.

2) Direct Access:-

→ In this type of memory, the data access any where in the address of memory location.

→ the data read/write in to direct storage there is no order to access.

Magnetic Disk or Hard Disk Drive

→ It is made up of a series of Circular disk called platters arranged one over another atleast 1/2 inch gap around a Spindle.

→ Disk are made up of non-magnetic material like aluminium alloy and coated with 10-20 nm of magnetic oxide.

→ It rotate speed is varied from 4200 to 15000 rpm.

→ Two arm used for read or write data on it.

→ A typical modern HDD capacity in Gigabytes or Terabytes.

Optical Disk

→ It is two types such as CD and DVD

CD (Compact Disk)

→ CD are circular disk that use optical ray (usally laser) to read or write data on it.
→ Data are stored in the form of digital.
→ The size of CD is 700 MB
→ It is portable
→ It has three types such as:-
  a) CD-ROM (CD-Read Only Memory)
    → It is recorded by manufacturer, the user can read only.
  b) CD-R (CD-Recordable)
    → In this CD, data can be write once and cannot be erased or modified.
  c) CD-RW (CD-Rewritable)
    → Data can be write and delete on it.

DVD (Digital versatile Disk)

→ It is same as CD but storage capacity is more.
→ The storage capacity of DVD is 4.7GB
→ It has three types:-
  a) DVD-ROM (b) DVD-R (c) DVD-RW

Pendrive or flash Drive

→ It is also called data traveller it means a user can move from one computer to another by hold it.
→ Size of pendrive is 4, 8, 16, 32, 64 etc.

3rd Sem 11/09/2020 Sub:- TH1(CSA)

4.3) RAM and ROM Organization

RAM Organization

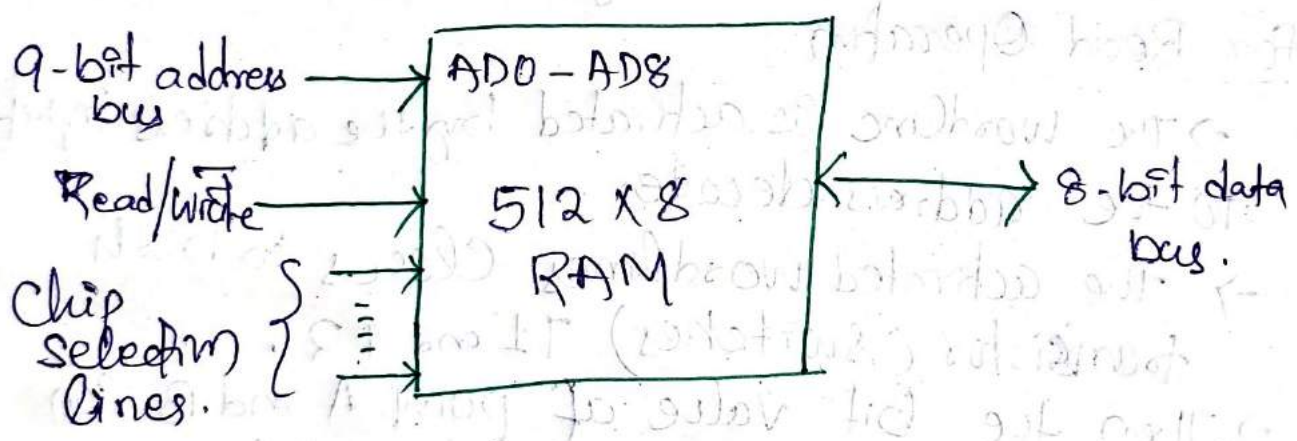
→ RAM is a part of Computer's main memory which is directly accessible by CPU.

→ It is used to Read and Write data into it which is accessed by CPU randomly.

→ Integrated Semiconductor chip (RAM) are available in two form:-

1) SRAM (Static RAM) 2) DRAM (Dynamic RAM)

Block diagram of RAM Chip such as:-



SRAM

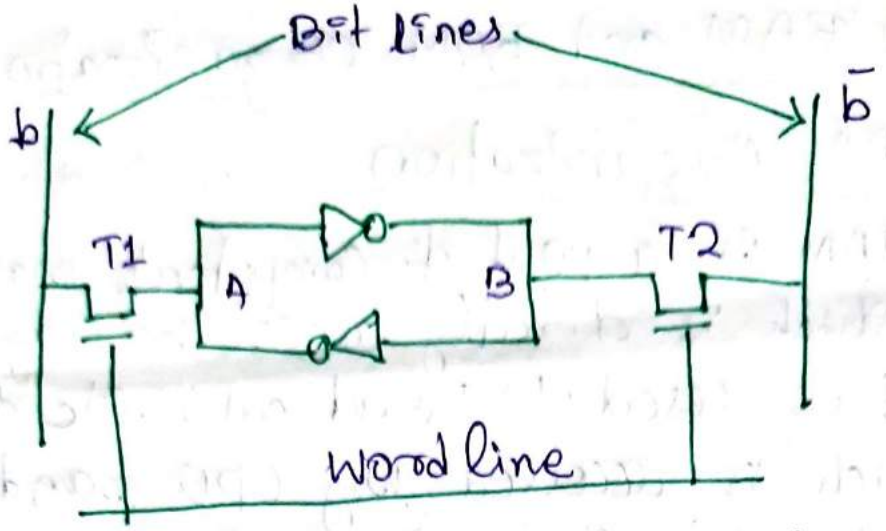
→ The SRAM memory consists of circuits capable to maintaining the stored information as long as power is applied.

→ SRAM is used to build Cache memory.

→ This memory is called volatile

→ The cell diagram of SRAM as follows:-

Cell diagram of SRAM



→ In the above figure., two transistor T1 and T2 are connected with two bit lines.

→ These transistors is to act as switches for open or closed control of the wordline, which is controlled by address decoder.

For Read Operation

→ the wordline is activated by the address input to the address decoder.

→ the activated wordlines closes to both transistor (switches) T1 and T2.

→ Then the bit value at point A and B can transmits to their respective bit lines.

→ At the end of bit lines it sends output to the processor.

For Write operation

→ the address provided to the decoder activates the wordline to close with T1 and T2 switches.

→ Then the bit value be written in to the cell provide the write circuit and the signal in bit lines are then stored in the cell.

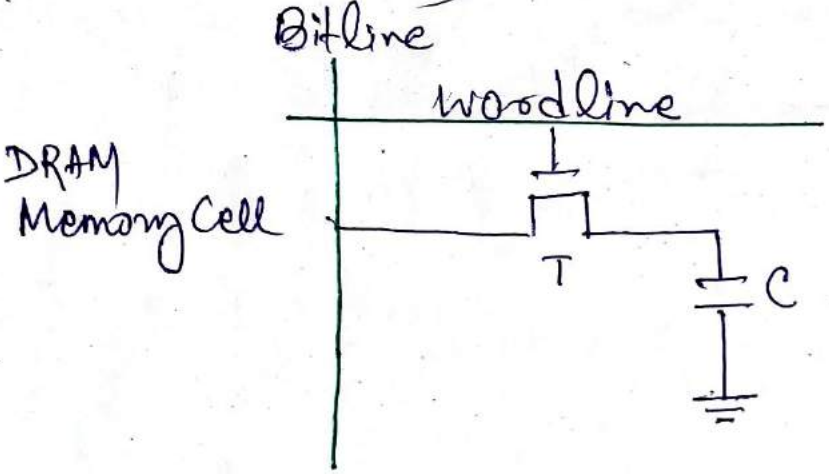
DRAM

→ It stores binary information in the form of electric charge that applied to circuits.

→ The stored information on the capacitors tend to lose over a period of time and to maintain the information the capacitor must be ~~period~~ regularly refreshed.

→ DRAM uses one capacitor and one transistor

→ The cell diagram of DRAM as follows.



→ In the above figure 'C' is the capacitor and T is the transistor.

→ ~~Information~~ Information is stored in a DRAM cell in a form of charge on a capacitor. and this charge needs randomly recharged.

→ For storing information in this cell, the transistor T is turned on and an appropriate voltage is applied to the bit line. So a known amount of charge to be stored in the capacitor.

→ After that the transistor ~~turned~~ turned off, then capacitor starts discharge.

3rd Sem 12/09/2020 Sub:- TH1 (CSA)

ROM Organization

→ As the name read only memory (ROM) is a memory unit that performs only read operation not write operation.

→ It stores binary information permanently during the hardware production and it cannot be altered or changed.

→ ROM is the part of main memory but it is non-volatile category.

→ The information is embedded in the ROM in the form of bits by a process is called programming the ROM.

→ Booting instruction is programmed on ROM chip.

→ Programming is used to refer the hardware procedure which is properly configuration of the computer:

→ If not configured properly it's show an error message as blue screen.

→ This is possible by use of PLD (Programmable Logic Device)

Programmable Logic Device

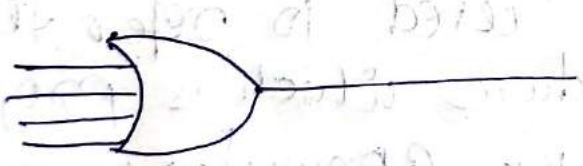
→ A programmable logic Device is an IC (integrated Circuit) with internal logic gates, which connected through electronic path that behaves similar to fuses.

→ In original state the fuse are intact otherwise not

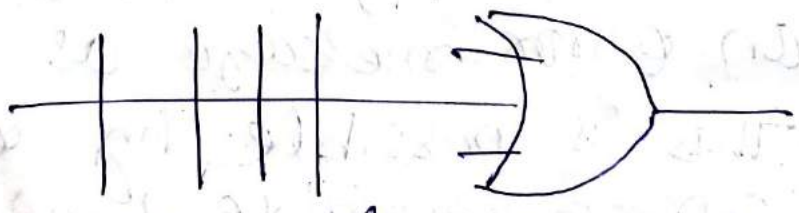
→ ROM Consists of nothing but basic logic gates arranged in such a way that they store the specific bits.

→ A PLD can have hundreds to millions of gates interconnected through hundreds to thousands of internal paths.

→ The internal logic diagram as follows.



(Conventional Symbol)

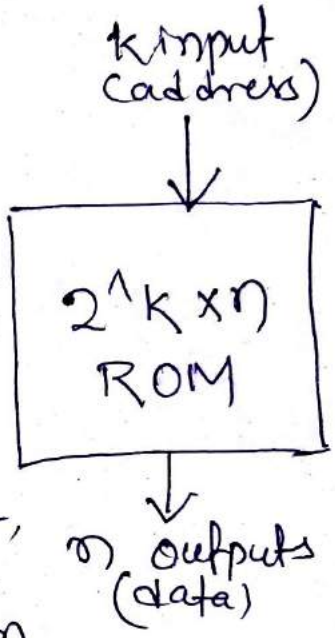


(Array logic Symbol)

→ The first image shows the Conventional Symbol, it represent as inputing logic gates

→ The Second Symbol shows the Special ways of showing inputs to a logic gate called as Array logic Symbol. where vertical lines represents the input to the logic gate.

Block Diagram of ROM Chip



Block Structure

→ It Consists of k-input lines and n-output lines.

→ k-input used take input address to access the Content in ROM.

→ k-input lines can be either 0 or 1, so there are 2k total address. which Contain n bit information as output to ROM.

→ So a ROM Specified as 2k x n ROM.

Internal Structure

→ It Consists of two basic Components - Decoders or OR gates.

→ A decoder is a logic Circuit which decode from any encoded form.

→ All the OR Gate present in the ROM will have output of decoder as there input

4.4) Interleaved Memory

→ It is a technique for compensating the relatively slow speed of DRAM (Dynamic RAM).

→ In this technique, the memory is divided into memory banks (modules) which can be accessed individually without any dependent on the other memory.

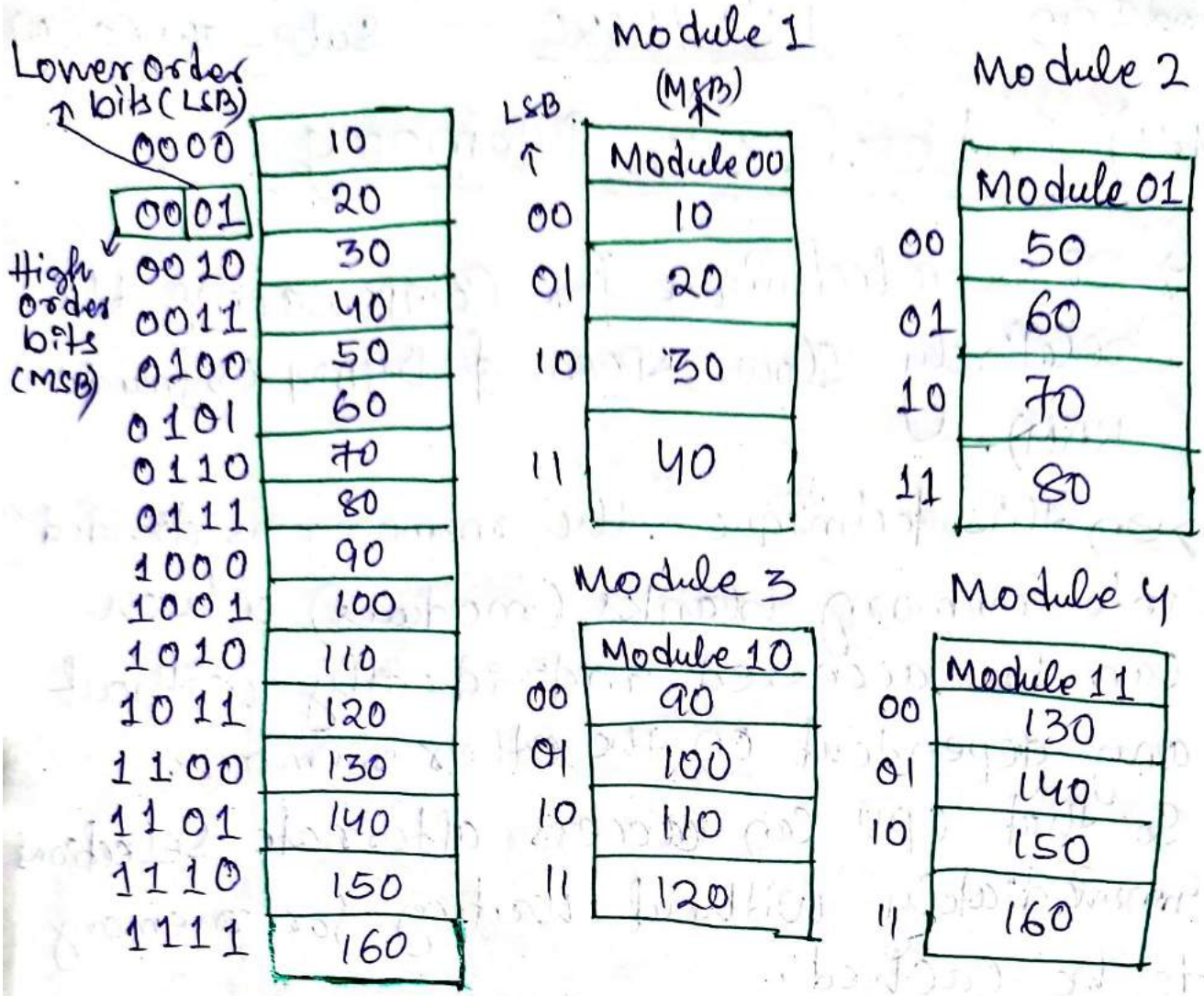
→ So that CPU can access alternate sections immediately without waiting for memory to be cached.

→ Memory interleaving is a technique for increasing memory speed, so the system have more efficient, fast and reliable.

→ It is a technique which divides memory into a number of modules such that successive word in the address space are placed in the different module.

→ To improve the access time of the main memory interleaving is used.

Ex:- Consecutive word in Consecutive module.



→ 16 Data's are to be transferred to four module. where Module 00 is Module 1, Module 01 is Module 2, Module 10 is Module 3 and Moule 11 is Module 4. also data in Consecutive as 10, 20, 30 ... 160.

→ The data are consecutively in module till its maximum capacity.

→ MSB (Most Significant Bit) provide the address of module and LSB (Least Significant bit) provides the address of the data module.

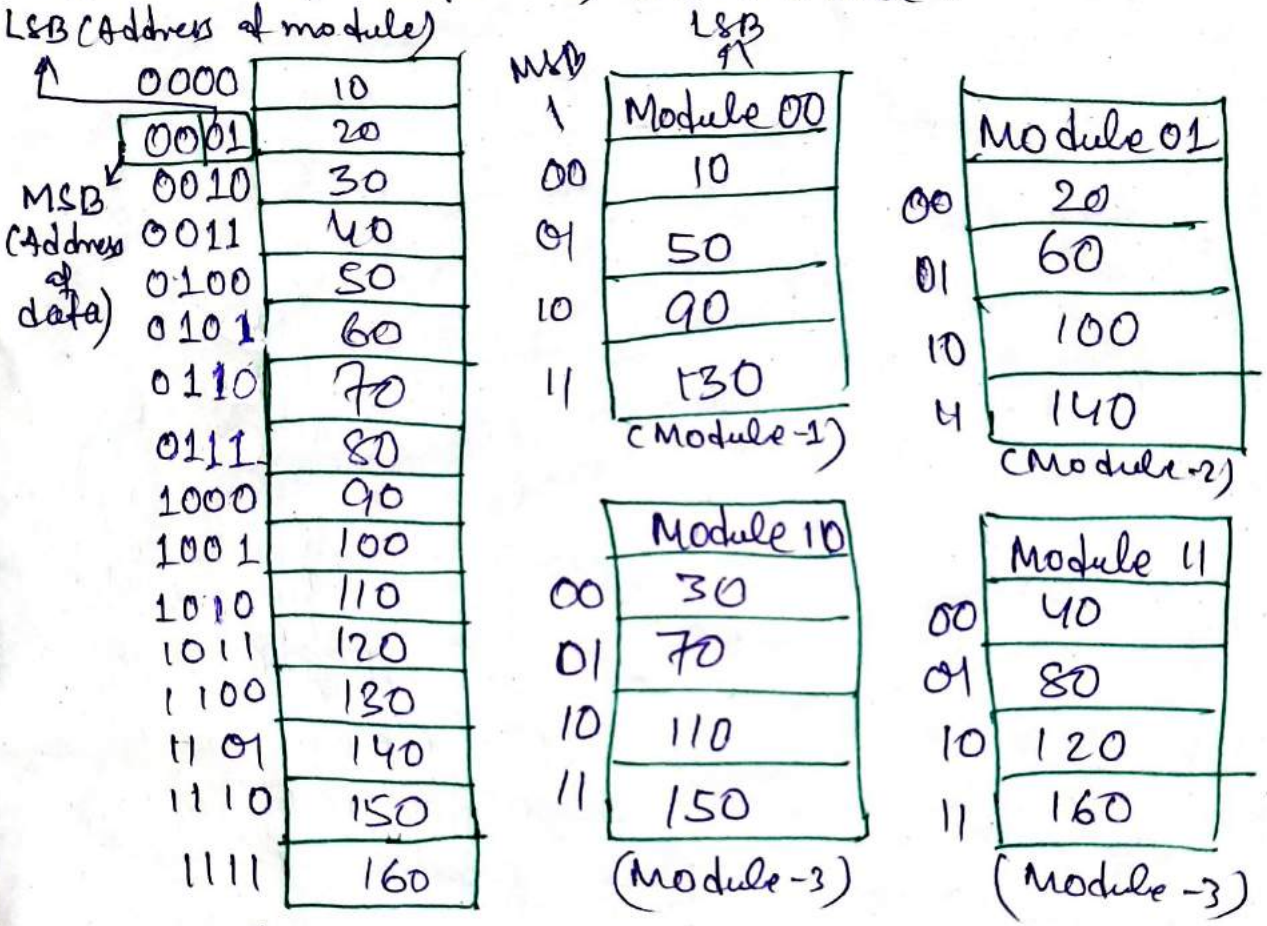
→ So to get data 90, then the processor provided 1000. on this 10 (MSB) indicates data in module 10 (module 3) and 00 (LSB) is the address of 90 in module 10 (module 3)

So,

Module 1 Contains Data!- 10, 20, 30, 40
Module 2 Contains Data!- 50, 60, 70, 80
Module 3 Contains Data!- 90, 100, 110, 120
Module 4 Contains Data!- 130, 140, 150, 160

Ex-2 Consecutive word in different module.

→ In this technique the LSB provides the address of module and MSB provides the address of data in the module.



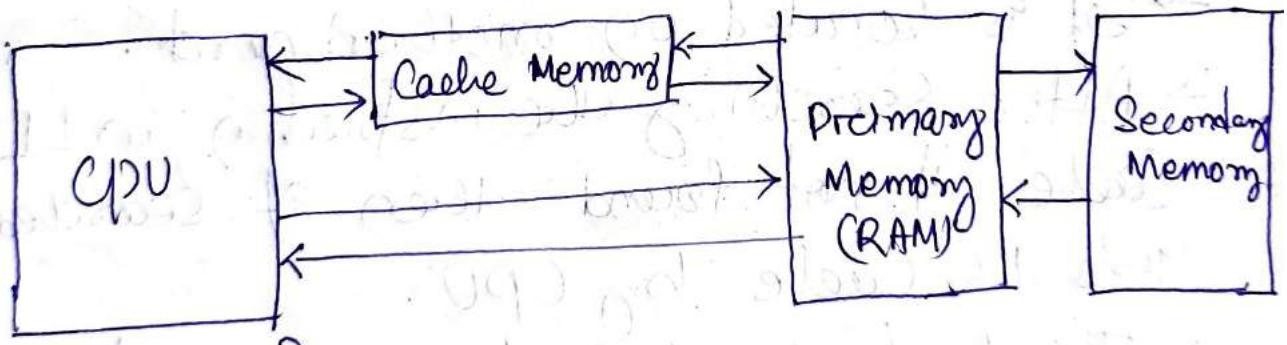
→ Now 16's Data be tranfered to four module. so Consecutive words (data) are added in different module.

Ex:- to get 90 (Data) 1000 provided by processor. In this 00 (LSB) is the address of module (module 00) indicates module-1 and 10 (MSB) is the address of data in module-1. so that

Module-1 Contains data!- 10, 50, 90, 130
Module-2 Contains data!- 20, 60, 100, 140
Module-3 Contains data!- 30, 70, 110, 150
Module-4 Contains data!- 40, 80, 120, 160

4.5) Cache Memory

→ Cache memory is a special memory used for very fast processing of program.
→ It is a very high-speed executing memory
→ It is used to speed up and synchronizing with high-speed CPU.
→ It is costlier than main memory and Secondary
→ Cache memory is act as a buffer between RAM and CPU.
→ Cache means to store information from user and send to CPU for execution.
→ The capacity of Cache is small so it is very faster in execution.



Importance of cache memory

→ The cache memory lies in the path between CPU and RAM, so it required less access time than RAM and it is faster than RAM.
→ A cache memory access time of 100 ns, while RAM may an access time 700ns.

Types of Cache Memory

→ It is divided into three levels such as:-

1) Level 1 (L1) Cache or Primary Cache

→ L1 is the primary type of cache memory.
→ It is embedded in the processor (CPU)
→ The size of L1 cache is between 2KB to 64KB., it depend on computer processor.
Ex:- accumulator, address register, program counter etc

2) Level 2 (L2) Cache or Secondary Cache

→ L2 is secondary type of cache memory.
→ the size of L2 cache is between 256 KB to 512KB.
→ It is located on motherboard.
→ After searching the instruction in L1 cache, if not found then it searched onto L2 cache by CPU.
→ The high speed system bus is interconnecting the cache(L2) to the CPU.

3) Level 3 (L3) Cache or Main Memory

→ L3 cache is larger in size but slower in speed than L1 and L2, its size between 1MB to 8MB.
→ Multicore processor, each processor have separate L1 and L2, but all core share a common L3 cache.

Cache performance

→ When the processor needs to read or write location in main-memory, it first checks for entry in the cache.

* If the processor finds that the memory location is in the cache, a cache hit has occured and data is read from cache.
* If the processor does not find the memory location in the cache, a cache miss has occured.

→ for a cache miss, the cache allocates a new entry from main-memory.

→ The performance of cache memory is called hit ratio.

Hit ratio = hit / (hit + miss)

⇒ Total hits / total access

Continue from Cache memory ...

Cache Mapping

→ The transmission of data from main-memory (RAM) to cache memory is called mapping, the process is called Cache mapping

→ There are three types of mapping such as:-

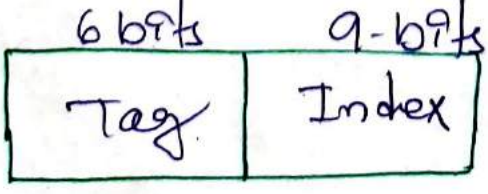
1) Direct Mapping
2) Associate Mapping
3) Set Associative Mapping

1) Direct Mapping

→ The CPU address of 15 bits is divided into two blocks or fields. i:e Tag and Index.

→ In this the 9 least significant bits (LSB) constitute the index field and remaining 6 bits constitute the tag field.

→ In direct mapping technique the number of bits in index field is equal to the number of address bits is required to access Cache memory.



(Direct Mapping)

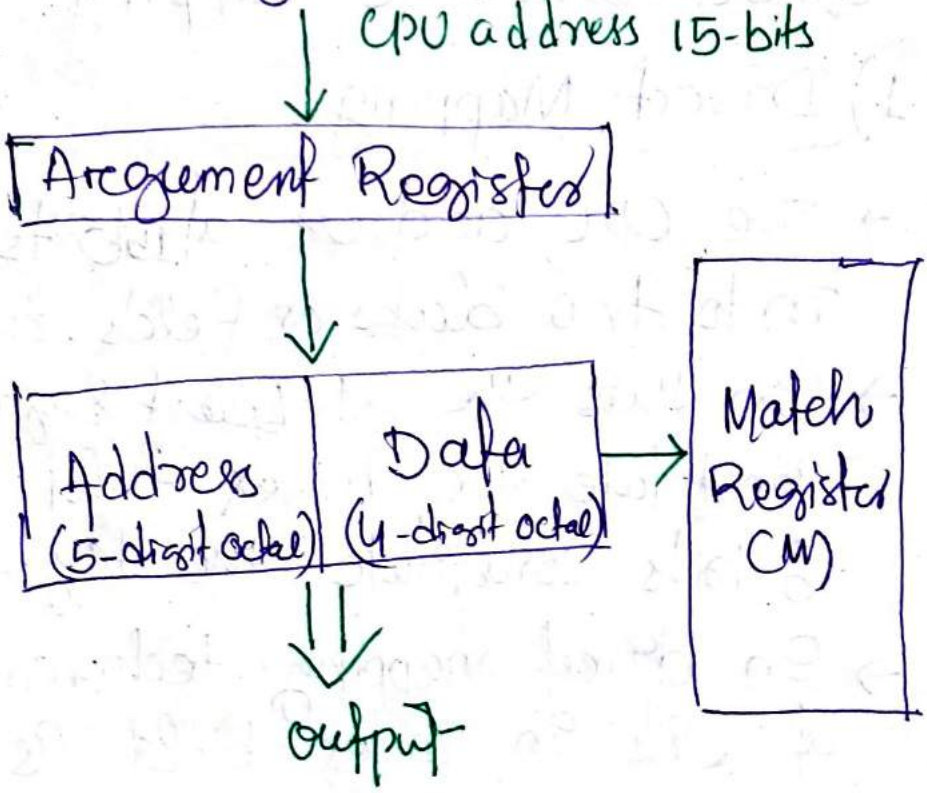
2) Associative Mapping

→ In associative memory mapping stores both address and data.

→ The address value of 15 bits is holds 5 digit octal numbers and data field of 12 bits word (data) holds 4 digit octal number.

→ A CPU address of 15-bits is placed in argument register.

→ The associative memory is searched for matching register address then access cache memory.



(Associative mapping diagram)

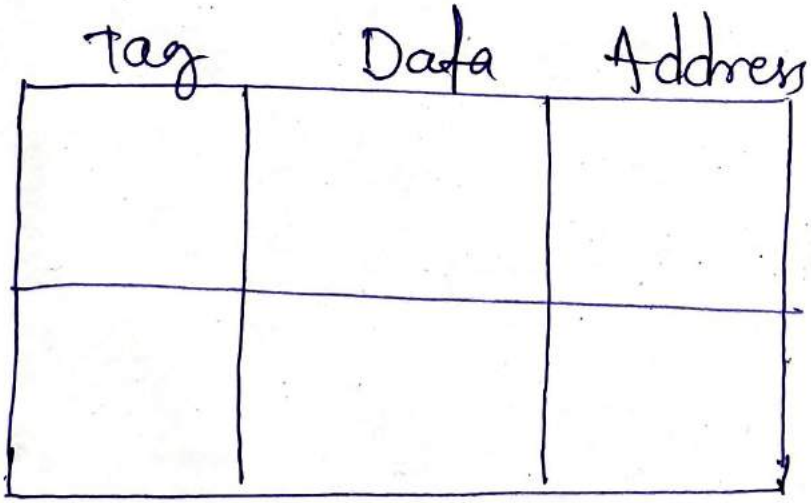
3) Set Associative Mapping

→ The disadvantages of direct mapping is that two words with same index address can not reside in cache memory at the same time.

→ To overcome this problem so set associative mapping technique is used.

→ In this technique we can store two or more words of memory under the same index address

→ Each data word is stored together with its tag and it forms a set of data and address.



3rd Sem 19/09/2020 Sub:- TH1 (CSA)

4.6) Virtual Memory

→ Virtual memory is a valuable concept in computer architecture that allows users to run large programs on computer even if small amount of RAM.

→ A computer fulfill the demands of multi-programming approach with in fixed amount of physical memory.

→ A Computer can address more memory than the amount of physical memory is installed on the system. This extra memory is actually called "Virtual memory".

→ Virtual Memory is a section of a hard disk is used to compensate the shortage of physical memory (RAM) by transferring pages of data from RAM to HDD.

→ Virtual memory process is done temporarily and is designed to work as combination of RAM and space on hard disk.

→ This means when RAM runs low memory Virtual memory can move data from it to a space called a page. This process allow for RAM to be free till completing the task.

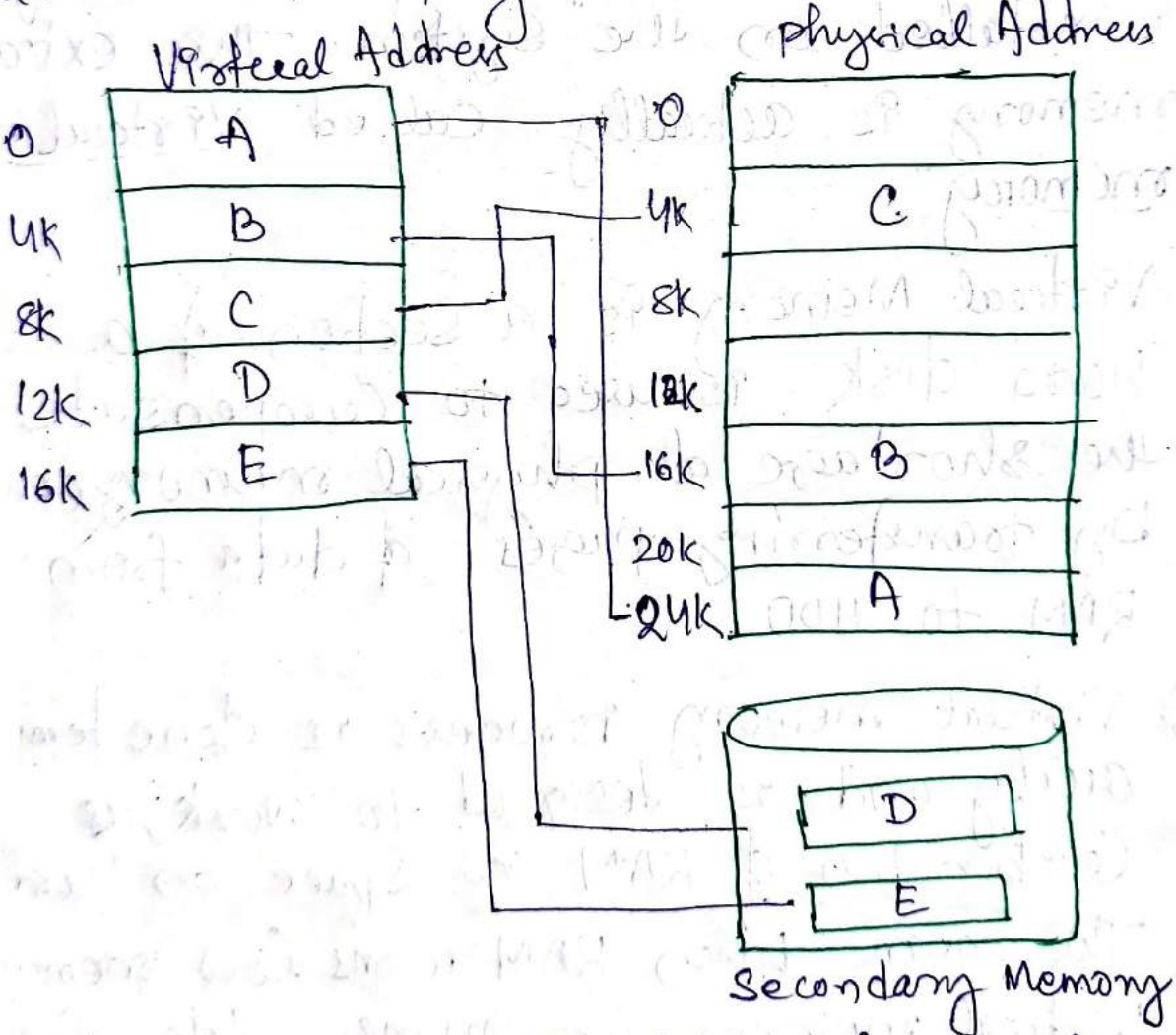
→ Virtual memory has two purposes such as.

a) first, it allows to extended the use of physical memory by using HDD.

b) Second, it allows to memory protection because each virtual address is translated to physical address.

→ Modern microprocessors (CPU) intended for general purpose use, so a main-memory unit (MMU) is built into the hardware.

→ The MMU is used to translate virtual address into physical address.



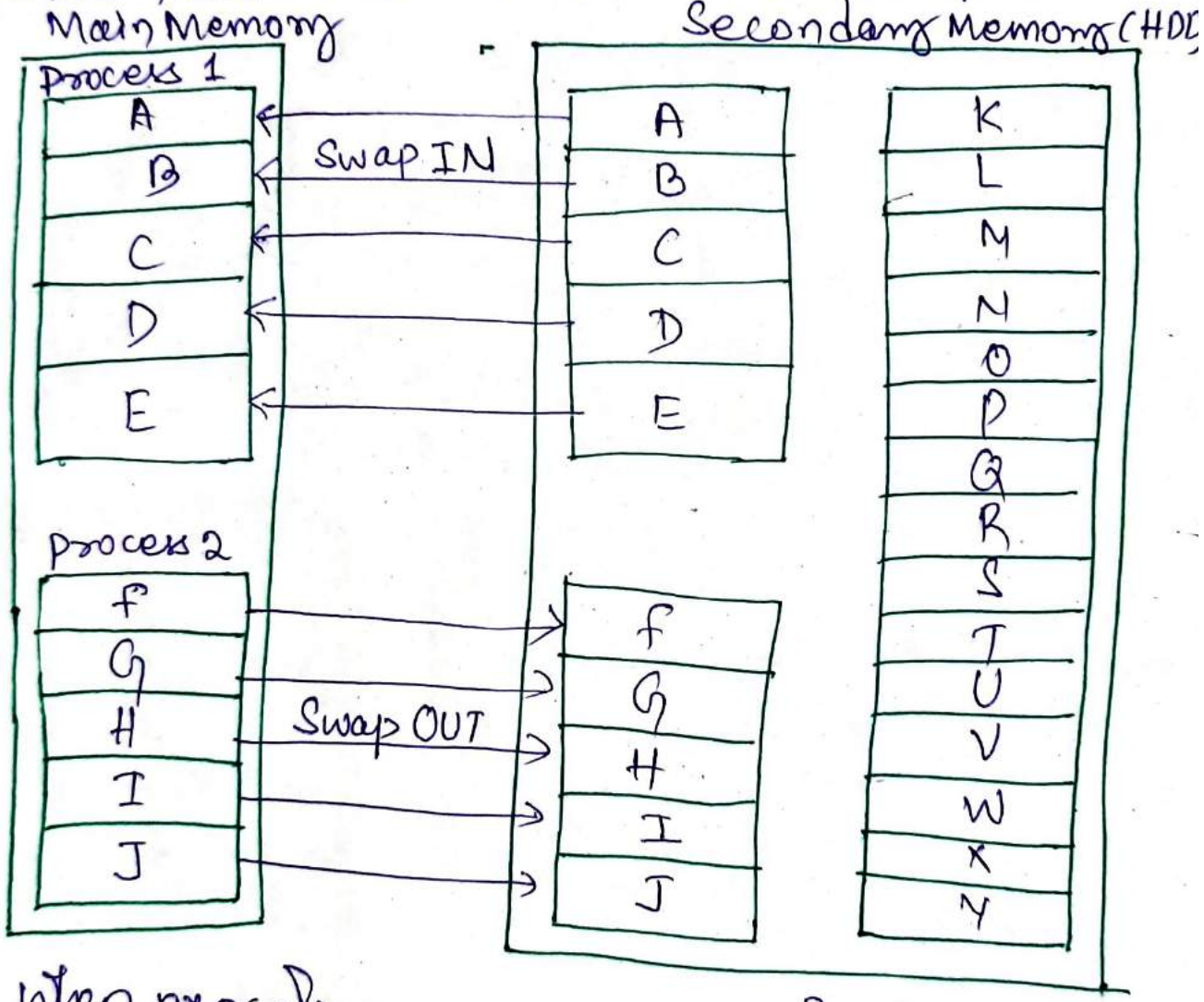
→ Virtual memory is commonly implemented by demand paging.

→ Demand Segmentain can also be used.

Demand paging

→ A demand paging is similar to a paging system with swapping technology where the process reside on hard disk and the pages are loaded only on demand, not in advance.

Paging :- Virtual memory breaks program into fixed-size block is called pages.



→ When executing a program if a page is just swap OUT from RAM than the CPU treat as invalid page reference is called page fault.

→ Than Operating System to demand the page back into RAM.

3rd Sem 21/09/2020 Sub:- TH1 (CSA)

Unit-5 # Input - Output System

What is Input / Output System?

→ The input/output (I/O) system of a Computer provides an efficient mode of communication between the central system and the outside environment. Also it can handles all the input-output operation of the Computer System.

→ In a Computer system various input or output devices are connected for doing operations called peripheral devices.

→ BUS is used for data transfer between I/O and CPU for communications.

Peripheral Devices

→ Input or output devices that are connected to Computers are called peripheral devices.

→ These devices are designed to read information into or out of the memory unit by command from CPU.

→ These devices are also called peripherals.

→ Devices are the parts of Computer System.

Ex:- Keyboard, Display Units and printers etc. are common peripherals devices.

Types of peripherals

There are three types of pherals;-

1) Input peripherals:-

→ It allows user input from the outside devices to the computer.

Ex:- Keyboard, Mouse, Digital input etc.

2) Output peripherals:-

→ It allows information as output from the computer to outside devices.

Ex:- Printer, Monitor etc.

3) Input-Output peripherals:-

→ It allows both input as well as output for communication to the computer.

Ex:- Touch Screen etc.

5.1) Input-Output Interface

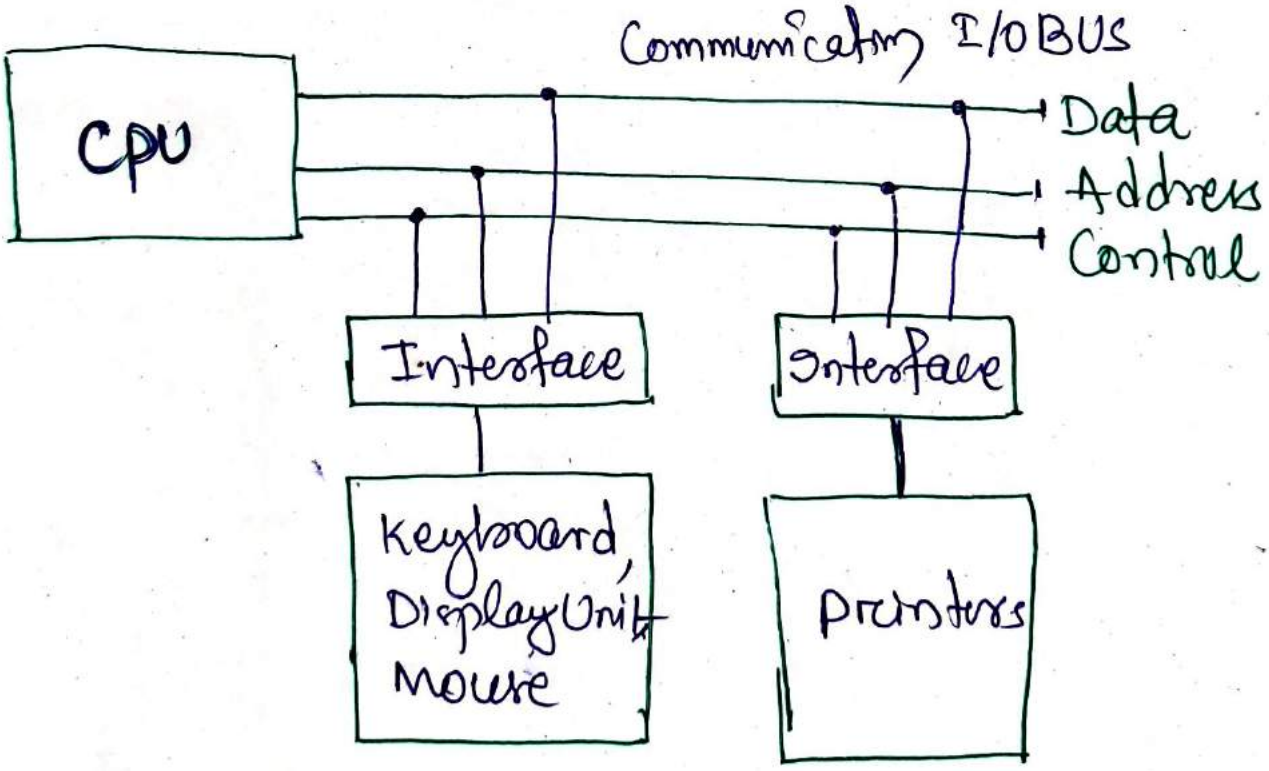
→ The method that is used to transfer information between internal storage and external input/out devices is known as input-output interface.

→ peripherals connected to a computer need a special communication links for interfacing them with the CPU.

→ In Computer System, there are special hardware Components between the CPU and peripherals to control or manage the Input - output transfers.

→ These Components are called Input-Output Interface units because they provide Communication link between processor bus and peripherals.

→ BUS are used for transferring information between internal system and input-output devices.



[Connecting of I/O BUS to I/O peripherals]

5.2) Modes of Data Transfer

→ The binary information that is received from an external device is usually stored in memory unit.

→ The information that is transferred from the CPU to the external devices is originated from the memory unit.

→ CPU process the data but it read and write is always the memory unit.

→ Data transfer between CPU and I/O devices can be handled by three types of data transfer modes such as:-

* Programmed I/O
* Interrupt driven I/O
* Direct Memory Access (DMA)

5.3) Programmed I/O Transfer

→ Programmed I/O transfer is one of the I/O technique other than interrupt drive I/O and DMA technique.

→ It is most simple type of I/O technique for exchange of data or any types of communication between the CPU and external devices.

→ With programmed I/O the data are exchanged between the processor (CPU) and the I/O module.

→ The processor executes a program that gives the direct control of the I/O operation including sensing device status, sending a read or write command, and transferring the data.

→ When a CPU provides a command to I/O module, than it must wait until the I/O operation is complete.

→ If the CPU is faster than the I/O module, than this is wasteful of processor time.

Operation of the programmed I/O as follows:-

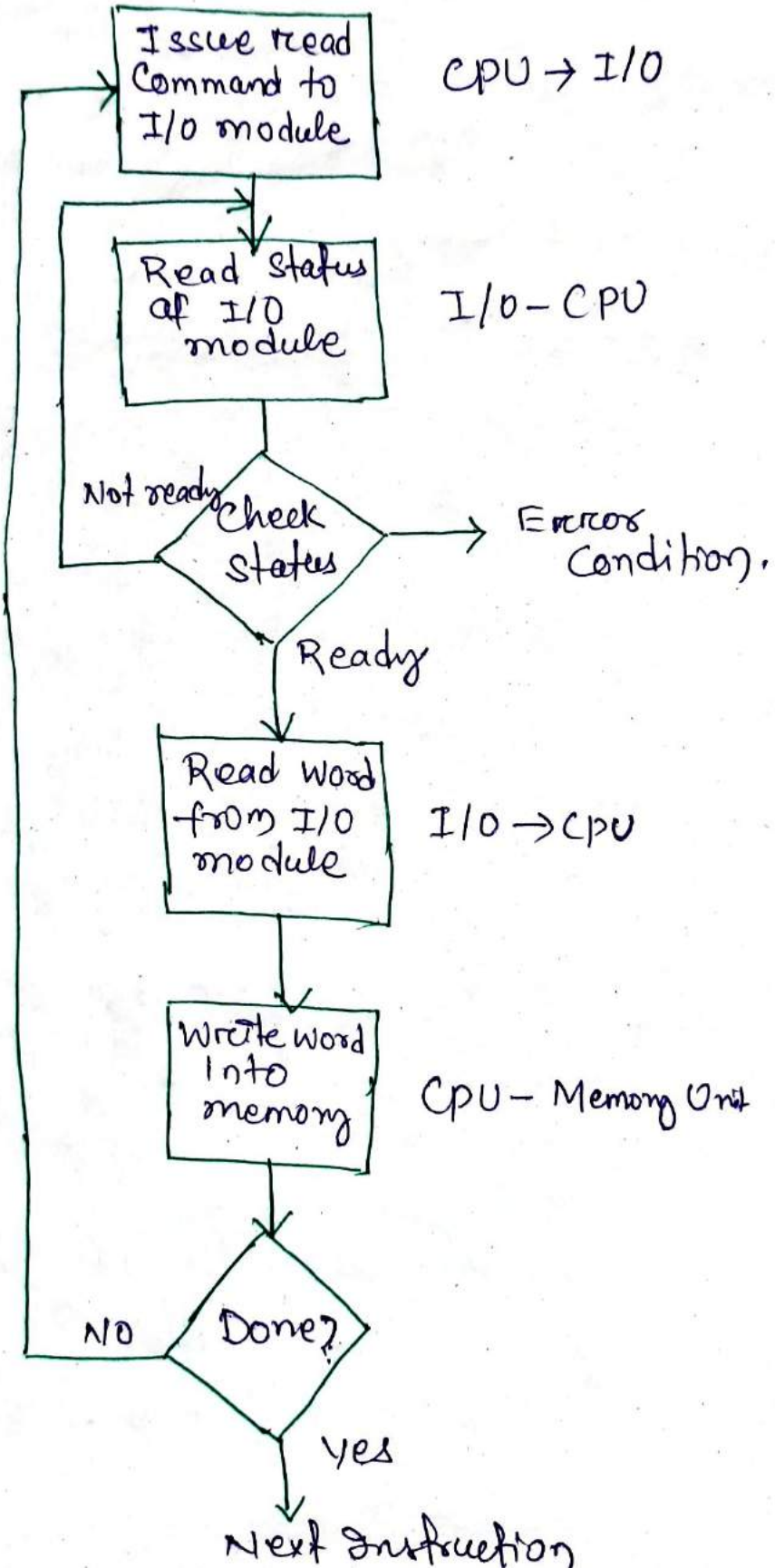
1) The processor is executing a program and found an instruction relating to I/O operation.

2) The processor then execute that instruction by providing a command to the appropiate I/O module.

3) The I/O module will perform the requested command by the processor (READ / WRITE) and set the bits in the I/O status register.

4) The processor periodically check the states of the I/O module until the operation is Complete.

Programmed I/O Mode Input Data Transfer



3rd Sem 25/09/2020 Sub:- TH1(CSA)

5.4) Interrupt Driven I/O

→ Interrupt driven I/O is an alternative technique for dealing with I/O operation effectively as compared to programmed I/O technique.

→ In programmed I/O technique we saw the CPU is kept busy unnecessarily, so that situation can be avoided by using interrupt driven method for data transfer.

→ Interrupt I/O is a way of controlling input/output operations by I/O devices by transfer a signal for received data or send data.

→ The signal will cause a program interrupt to be set.

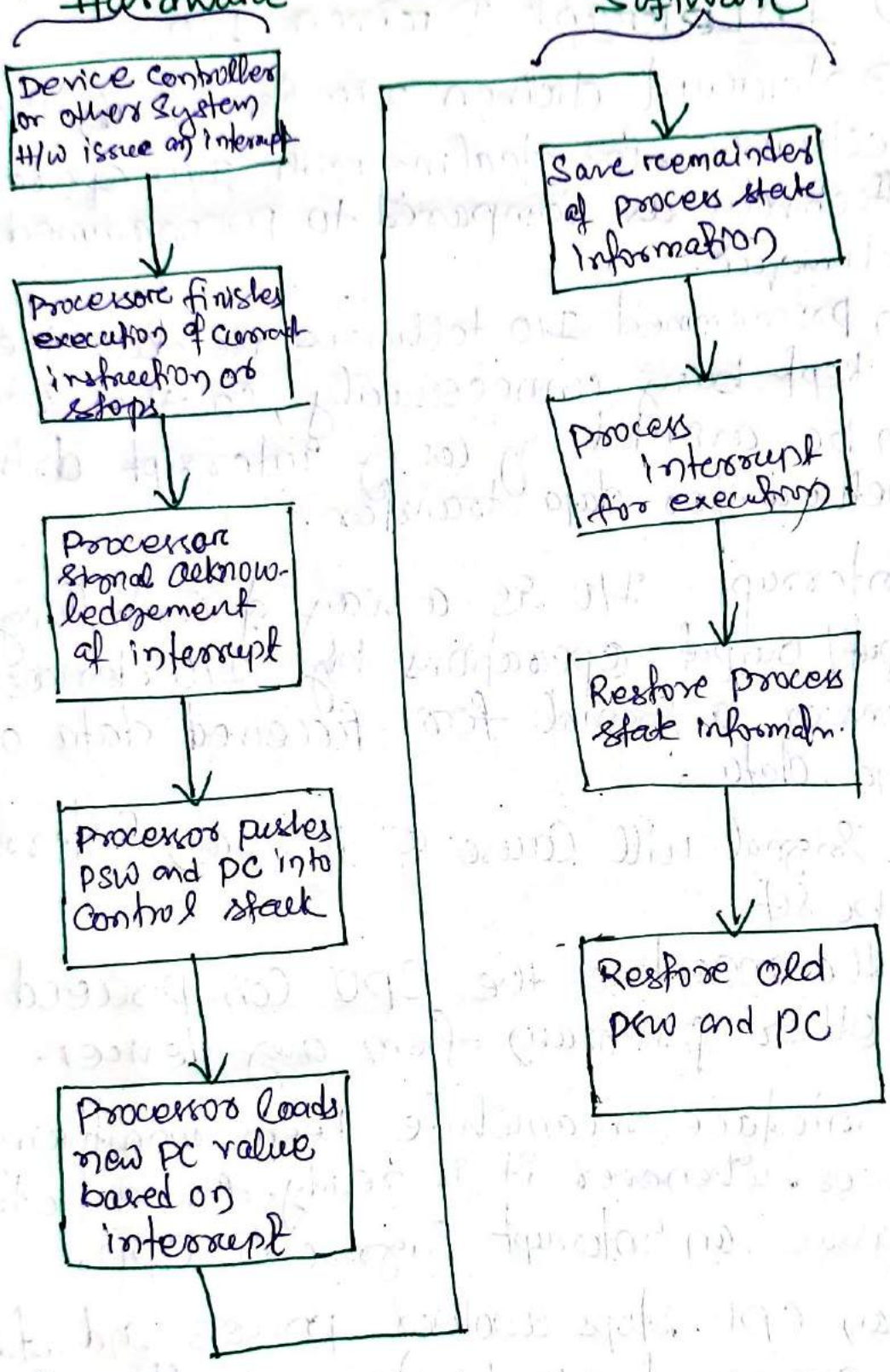
→ So the meantime the CPU can proceed for any other program from any devices.

→ The interface meanwhile keeps monitoring the devices, whenever it is ready for data tranfer it issue an interrupt signal to CPU.

→ Then CPU stops another process and start the process of I/O transfer and after completion then return to the task is was originally performed.

→ The interrupt technique requires more complex hardware and software, but it makes more efficient use of CPU time.

Simple interrupt processing as follows:-



Input

→ For input, the devices interrupts the CPU when new data has arrived and is ready to retrived by CPU.

→ Action was performed by I/O ports or memory mapping.

Output

-) for output, the device issue an interrupt for either is ready to accept new data or acknowledgement of a successful data transfer.

Basic Operation of Interrupt

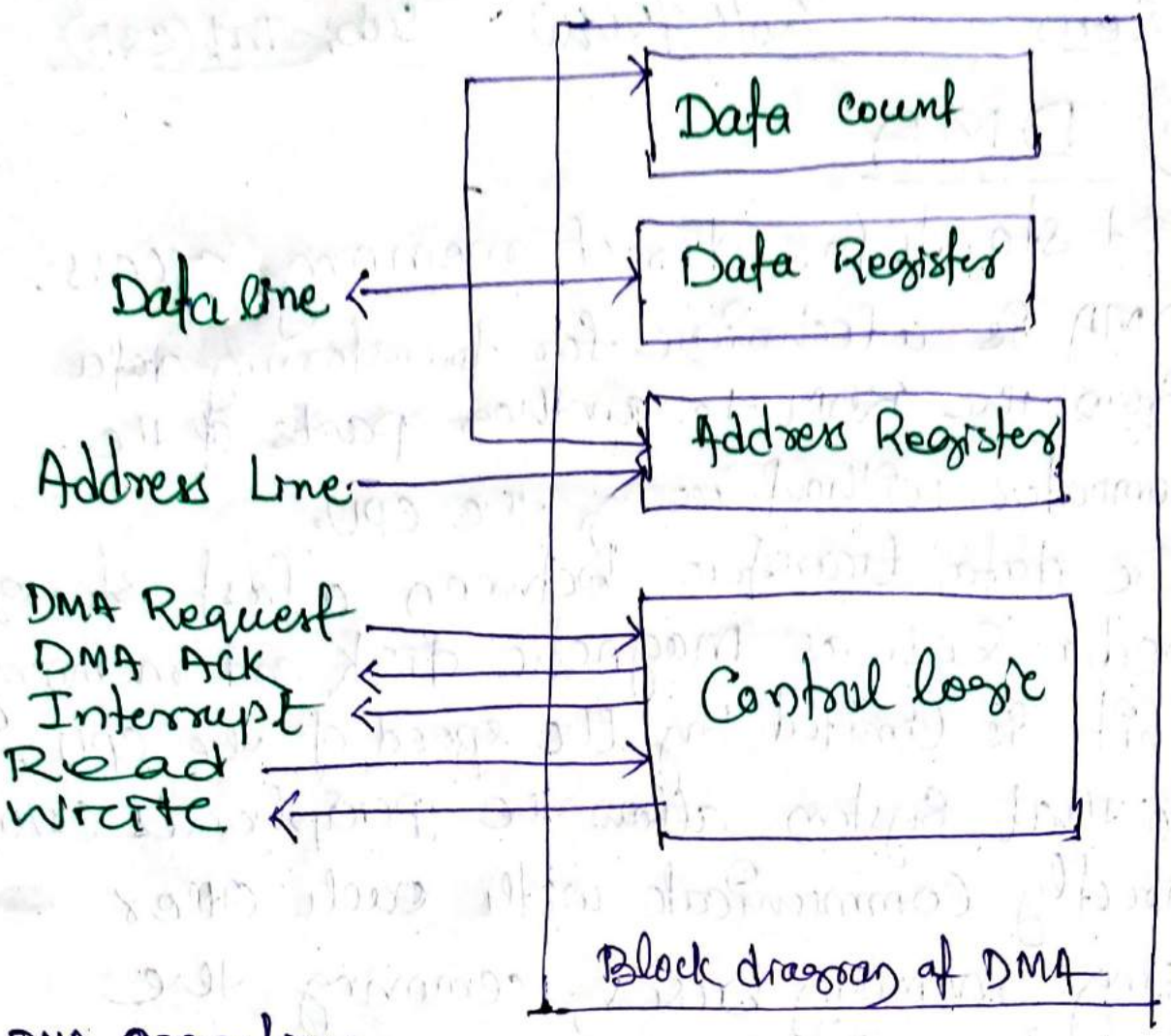
1) CPU issue read command.
2) I/O module (interface) get data from external peripheral, while CPU does other work.
3) I/O module interrupt CPU.
4) CPU process the requests data.
5) I/O module transfers data.

3rd Sem 26/09/2020 Sub:- Th1 (CSA)

5.5) DMA

→ It stands for direct memory access.
→ DMA is a technique for transferring data from the RAM to another parts of the Computer without using the CPU.
→ The data transfer between a fast storage media such as magnetic disk and memory unit is limited by the speed of the CPU.
→ So that System allow the peripherals (I/O) directly communicate with each other using memory buses, removing the intervation of the CPU.
→ This type of data tranfer technique is known as DMA (Direct Memory Access).
→ During DMA the CPU is idel and it has no control over the memory buses.
→ The DMA controller takes over the buses to manage the data transfer directly between I/O devices and the memory unit.
→ Many hardware system use DMA such as disk drive Controllers, graphics Cards, network Cards and Socund Cards etc.

DMA Block Diagram



Block diagram of DMA

DMA Operations

→ the CPU issue a command to DMA module to read or write I/O device address using data lines.

→ Then the starting of data lines is stored the address register.

→ The number of words to be transfered using data lines that stored in data register.

→ Then the processor continues their works.

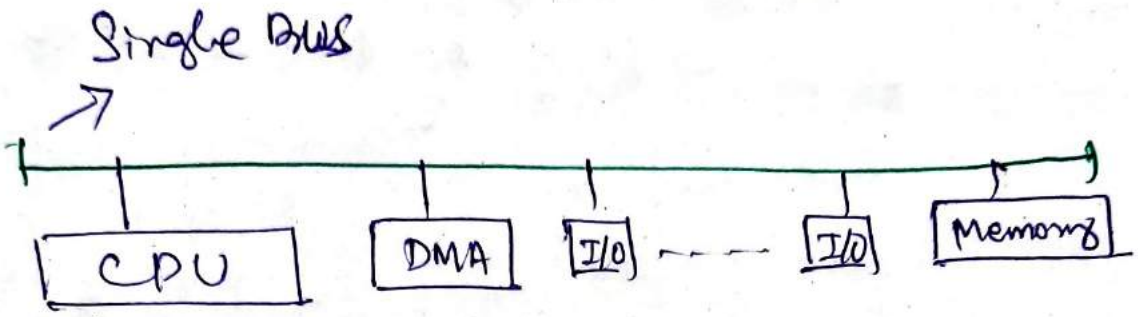
→ DMA module transfer the entire block of data - by one word at a time directly without using CPU.

→ DMA module sends an interrupt to the CPU when I/O operation is complete.

DMA Configuration

→ It is two types :-

1) Single BUS detached DMA module



→ This module supports more than one device, but each transfer of data uses BUS twice i.e I/O to DMA and DMA to memory unit.

2) Single BUS Integrated DMA module

→ It also supports more than one device but each transfer of data uses BUS only once i.e: DMA to Memory Unit.

→ CPU suspended only once.

Applications of DMA

* Disk Drive Controller
* Device Drivers
* Graphics Card
* Network Interface Card.
* Sound Cards, etc.

5.6) Input/Output Processore

→ An input-output processor (IOP) is a processore with direct memory access capability.

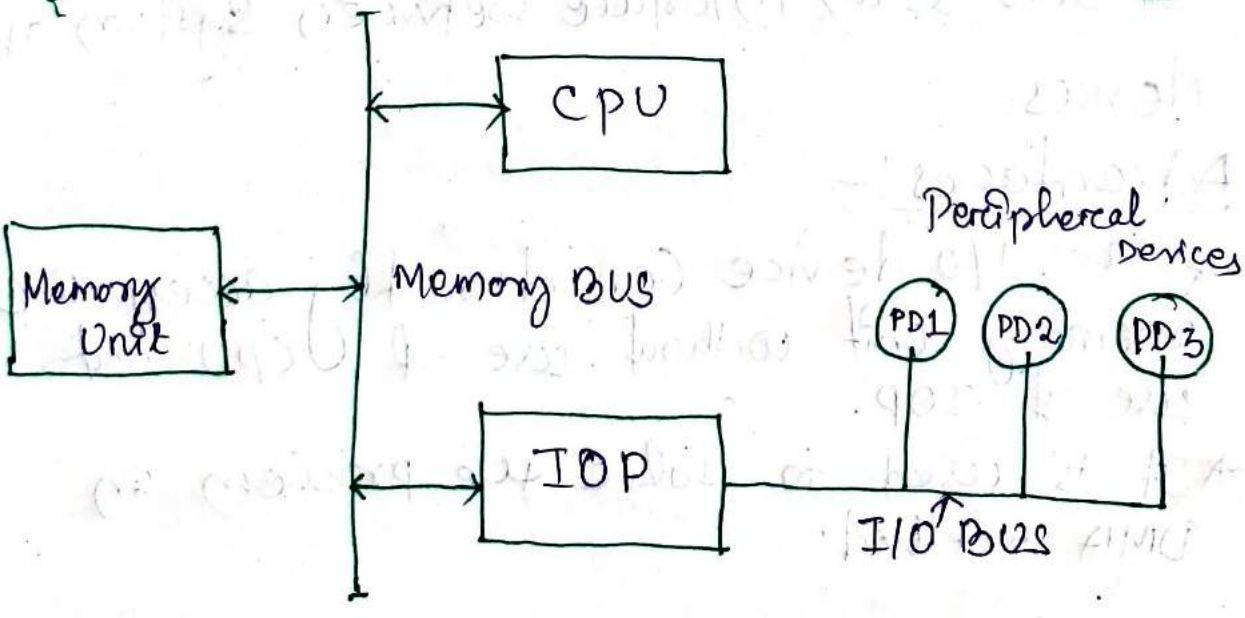
→ In I/O processor the computer system is divided into a memory unit and number of processors.

→ Each IOP controls and manage the input-output task or operations.

→ The IOP is similar to CPU but it handles only the operation of I/O processing.

→ So IOP can fetch and execute its own instructions. only, these IOP instruction one designed to manage I/O data transfers only.

Block Diagram of I/O processor



-> In the above diagram of I/O processors, the memory unit occupies the centeral positions and can communicate with CPU and IOP processors.

-> The CPU process the data required for solving the computational task.

-> Then the IOP provides a path for tranfers data between peripherals and memory unit.

-> The CPU assign the task of initiating the I/O program.

-> The IOP operates indep from CPU and transfer data from peripherals and memory unit.

-> The IOP is a specialized processor which loads and stores data into memory along with the execution of I/O instructions.

-> It acts as an interface between system and devices

Advantages:-

* The I/O device can directly access the memory unit without use of CPU but use of IOP.

* It is used to solve the problem in DMA method.

Functions of IOP

→ The communicatn between IOP and peripheral is similar to program control method of transfer and it is similar to DMA.

→ The CPU can act as master and IOP act as slave processor.

→ The CPU assign the task to the IOP, then IOP executes own instructn for I/O operatn without involving of CPU.

→ The CPU provide instructn to start I/O operatn and the IOP sends interrupt to CPU after I/O operatn is completed.

→ Instructions are read from memory by IOP are called Commands.

→ Commands are prepared by programmers and stored in to the memory.

→ CPU informs the IOP where to find the commands in memory then tranfers data between peripheral and memory unit.

Unit-6 :- I/O Interface and BUS structure

Introduction

→ Input Output Interface provides a method for transferring information between internal storage and external I/O devices.

→ Peripheral Connected to a Computer need a Special Communication links for interfacing them with CPU.

→ The Communication link (BUS) is used to resolve the problem that exist between the CPU and each peripheral devices.

The Major Differences are:—

1) peripherals are electromechanical and electromagnetic devices and CPU, RAM are electronic devices. So Conversion signal value may be needed.

2) The data rate of peripheral is usually slower than the rate of transfer of CPU. So a synchronization mechanism is used.

3) Data Codes and formats in the peripherals differ from the word format in the CPU and memory.

4) The operating mode of peripheral are different from each other and must be controlled, so it not distrubs to other devices connected to CPU.

→ To resolve these problems or differences, the Computer system include special hardware components between CPU and peripherals to proper communicate of I/O transfer is called I/O module or interface.

→ These components are called Interface Units Ex:- PCI, SCSI, USB etc.

I/O BUS and Interface Module

→ Bus defines the link between the CPU and peripheral.

→ The I/O Bus consists of data lines, address lines and control lines

→ The I/O Bus from CPU is attached to all peripherals to communicate.

→ To communicate with a device, the CPU provides device address on address lines.

→ Each Interface decodes the address and control received from I/O Bus, then

Signal provided for peripheral controller.

⇒ Then the data flow from peripheral and CPU.

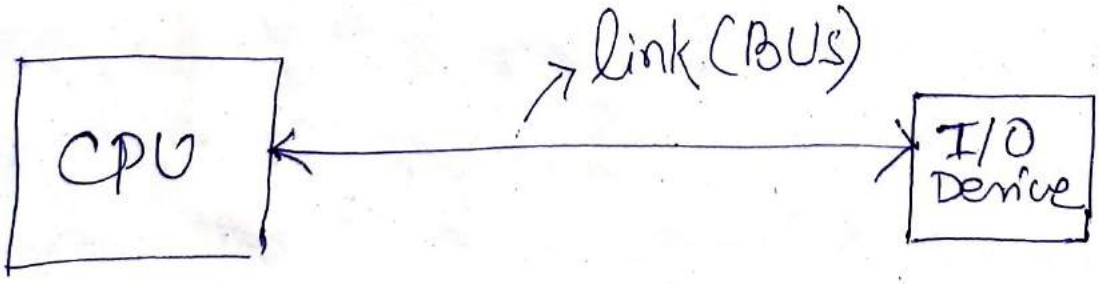
BUS Structure

→ A group of lines that serves as a connecting path for several devices is called Bus.

→ It used to carry data, for I/O operation.

→ The bus must have lines for address and control purposes.

→ In Computer architecture, a bus is a sub-system that transfer data between computer components inside the computer or between computers.



6.1) Bus and System Bus

→ In Computer architecture, a bus is a Communication System that transfers data inside the Computer or between Computers.

→ In early Computer buses were parallel electrical wires with multiple hardware connections.

→ Now the term bus means we arrange all lines within a single board called mother board which is combination of logical path and electrical bus.

→ Modern Computer uses SATA cable for data transmission, data send either parallel and serial bit by bit.

→ A bus can Communicates inside the Computer called internal buses and Communicate with peripherals called external buses.

Internal Buses

→ The internal bus is also known as internal data bus, memory bus, system bus or front side bus.

→ It Connects all the internal Components of a Computer such as CPU and memory to the motherboard.

→ This type of bus having quick response of execution of any program or instruction.

External buses

→ The external bus also called as expansion bus

→ This the electronic pathways which connects various external devices such as:- Printer, Audio cards, Graphics card, NIC etc. to the computer.

System Bus

→ A system bus is a single computer bus that connects the major components of computer system.

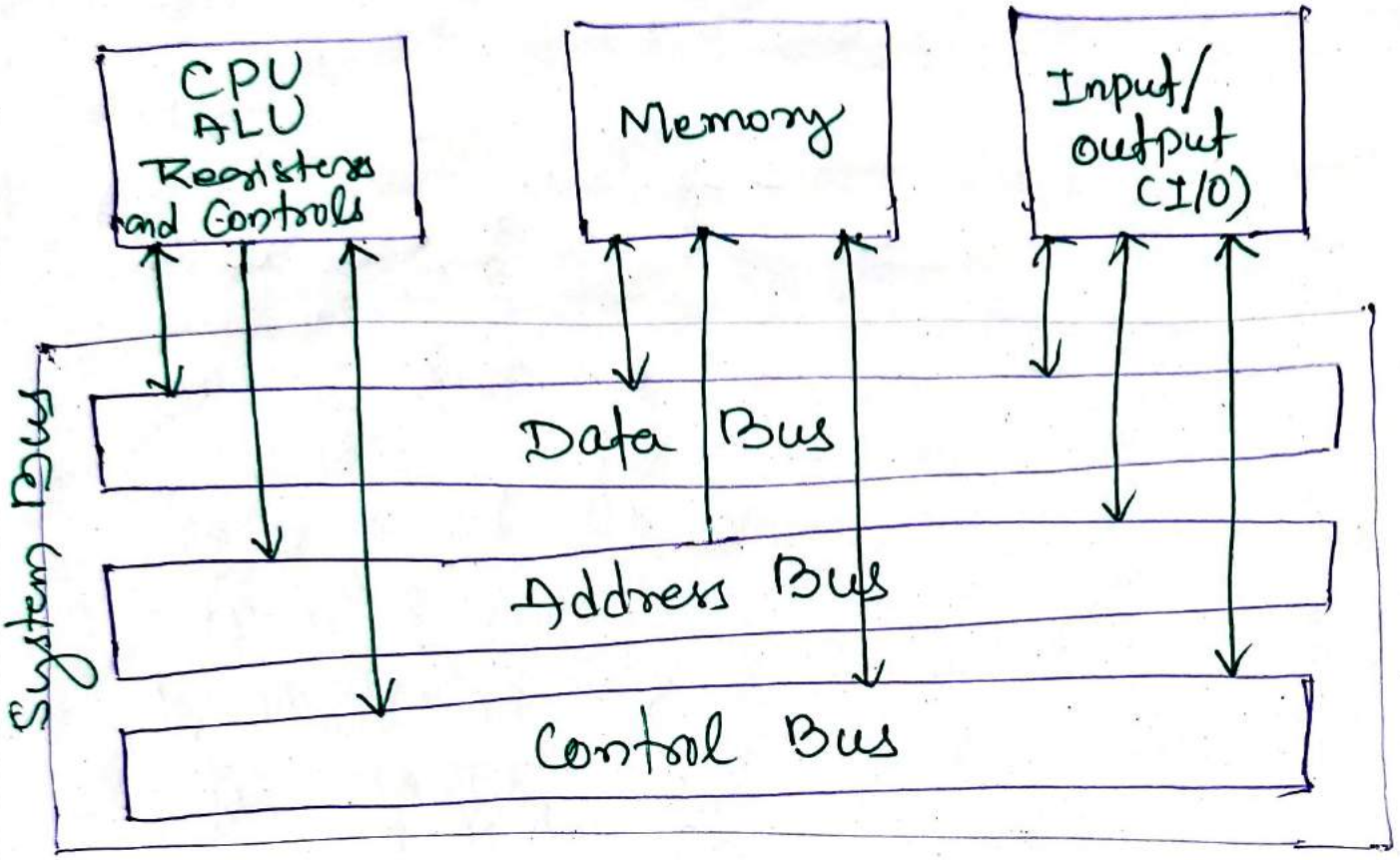
→ A system bus by combining the functions of data bus which is used to carry information, an address bus is used to determine where it should be sent, and control bus is used to determine its operation.

→ System bus contains about hundred number of lines separately, each lines assigns some functions.

→ A system bus usually seprated into 3 functional groups such as;-

* Data Bus, * Address Bus & Control Bus

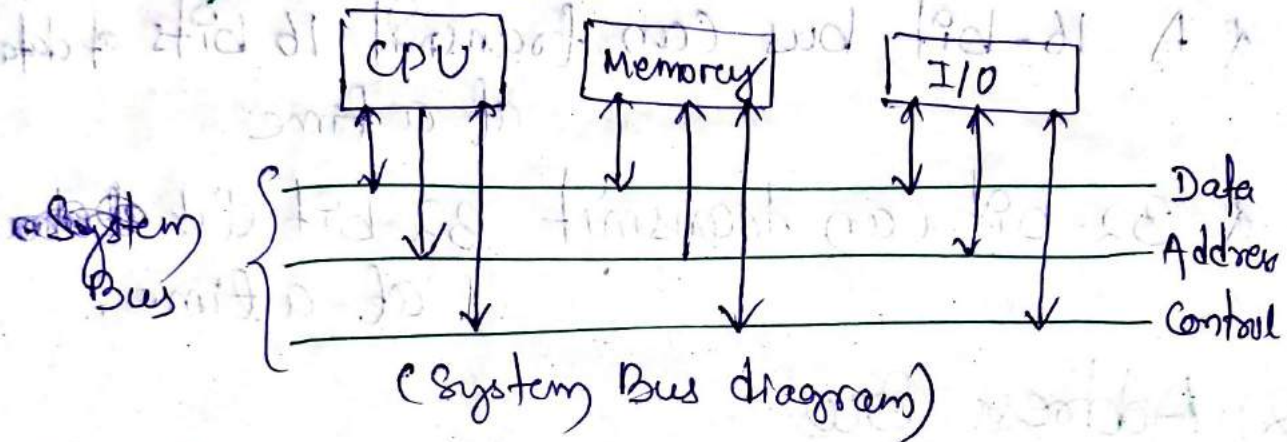
Diagram of System Bus :-



6.2) Types of System Bus

The System bus usally separated by three fenctional groups such as:-

1) Data Bus
2) Address Bus
3) Control Bus



(System Bus diagram)

1) Data Bus

→ A collection of wires through which data is transmitted from one part of Computer to another is called data bus.

→ It Consists of 8, 16, 32, 64, 128 --- more Separate lines for data transfer.

→ The total number of lines reffered as width of data bus which determines how much data can be transmitted at one time that is called size of the bus.

→ This bus connects all the peripherals to CPU and main memory for data transfered between them.

→ The data bus lines are bi-directional, which means CPU can read and write from memory using these lines to memory locations.

→ This bus connects all the components of a Computer to CPU and main memory.

Ex:-

* A 16-bit bus can transmit 16 bits of data at a time
* 32-bit can transmit 32-bit data at a time.

2) Address Bus

→ A collection of wires used to identify the particular location of main-memory is called address bus.

→ This bus is used to identify the source or destination of data.

→ This bus shared the address of data from one another over the bus.

→ The bus with (size) is determines the maximum capacity of system (processor)

Ex:- * A system with 4-bit address bus can address $2^4 = 16$ bytes of memory.

* A system with 16-bit address bus can address $2^{16} = 64$ KB of memory

3) Control Bus

→ The connections that carry control information between the CPU and other devices within the computer is called Control Bus.

→ This bus monitors data and address bus.

→ The control bus carries signals that report the status of various devices (I/O).

→ It can controls various signals such as:-

Memory Read / Memory Write.
I/O Read / I/O Write etc.
CPU Read / CPU Write etc.

Eg:-
This bus is used to indicate whether the CPU is reading from memory or writing to memory.

6.3) Bus Structure

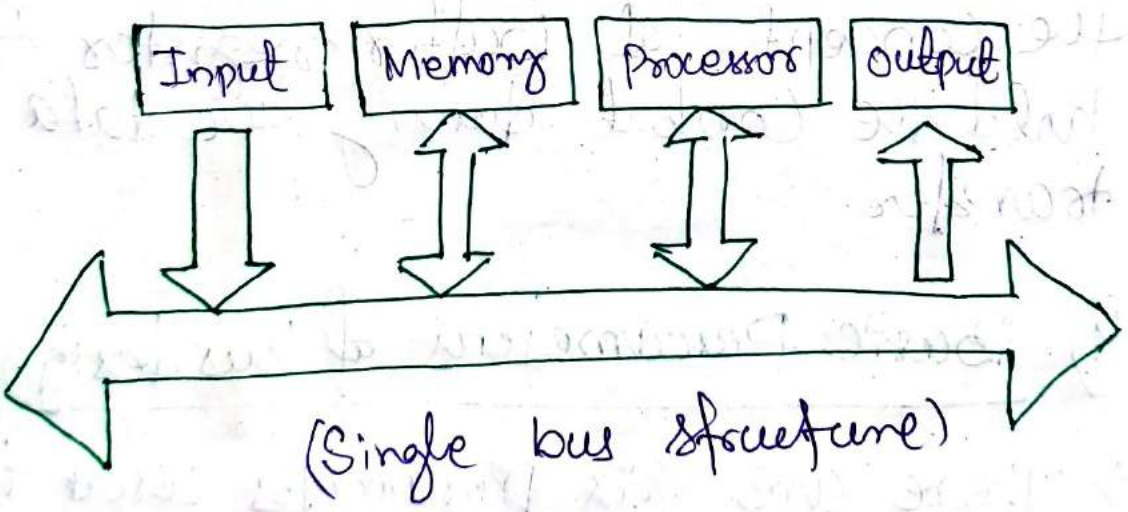
→ It is the simplest and most common way of interconnecting various parts of computer.

→ A group of lines that serve as a connecting port for several devices is called a bus.

→ The individual lines or path for exchange data, address and control information.

→ The data lines carry data, the bus must have address lines and control purpose.

→ The simplest way to interconnect is to use of single bus such as:-



(Single bus structure)

→ The bus can be used for transfer one bit at a time.

→ Single bus structure is very low cost and very flexible for attached peripheral devices.

→ Multiple bus structure increases performance but also increases the cost.

→ All the interconnected devices are not of same speed and time which creats a problem.

→ That problem can be solved by using cache registers (buffer registers).

→ Instructions from the processor at one are loaded to the buffer then the complet transfer of data as fast as possible for execution.

Ex:- Processor to printer shared a common bus for communication.

→ So A common approach is to use the concept of buffer registers to hold the content during the data transfer.

6.4) Basic Parameters of Bus design

→ There are six parameters used to design system bus.

→ There parameters are very essential for design the bus such as:-

1) Bus Types
   a) Dedicated
   b) Multiplexed
   c) Physical Dedication
2) Method of Arbitration
   a) Centercalized
   b) Distributed
3) Timing
   a) Synchronous
   b) Asynchronous
4) Bus width
5) Data transfer type
   a) Read - Modify-Write
   b) Read - After-Write
6) Block data transfer.

3rd Sem 17/10/2020 Sub:- TH1(CSA)

Continue from basic parameters to design bus:-

1) Bus Types

⇒ It is divided by three types such as:-

a) Dedicated:

→ A line is Permanently assigned either one function or other.

Ex:- So Data line and address lines are dedicated lines.

b) Multiplexed:

→ A single line is used for multiple purpose is called multiplexed.

Ex:- In single bus structure the data and address share information over the same set of lines.

→ If a data transfer to address bus then it activate signal after that it removes by transferring data.

c) Physical Dedication:

→ The use of multiple buses each of which connects to only subset mode. is either, a data module or address module.

2) Method of Arbitration

→ It is of two types such as! -

a) Centeralized Arbitration:

→ A single hardware device called the bus controller or arbiter allocate time on the bus.

→ The HW device may be a separate or apart from the processor (CPU)

b) Distributed Arbitration

→ There is no centeralized controller for bus.

→ Each module contain access control logic and modules act together.

3) Timing

a) Synchronous Timing:-

→ The system bus includes a clock time upon which a clock transmits a regular sequence of alternating data 0's and 1's.

→ A single bit data transfer is referred as the clock cycle or bus cycle.

-> All other devices on the bus can read the Clock cycle.

b) Asynchronous Timing

-> The occurance of program execution of one event on the bus following is depends on the previous event.

Ex:- A printer has a lot of job, some delay for a lot of job.

4) Bus width

-> The width of data bus has an impact on data bus has an impact on the System bus performance.

-> A wider bus, the greater number of bus was transfer data with wide range so it increase performance.

5) Data Transfer Type:-

-> It is of two types such as:-

a) Read-Modify-Write:- A read instruction followed by immediately write operation to the same address.

b) Read-After-Write:- It consisting to write operation followed by immediately by a read from the same address.

6) Block Data transfer:-

→ One address cycle followed by 'n' data cycle.
→ The first data item to or from the specific address.
→ Remaining data items to or from the subsequent address.
→ Multi address (lines) transferred a block of data directly.

PCI (Peripheral Component Interconnect)

→ PCI stands for peripheral Component Interconnect.

→ PCI is a hardware bus used to adding internal Component to the desktop Computer.

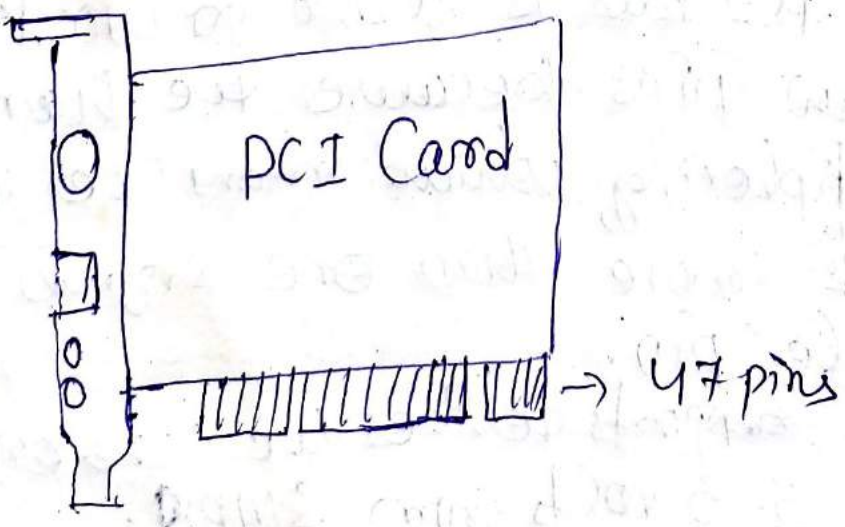
→ A PCI bus Connects the CPU and expansion boards such as:- modem, Sound, graphics Cards etc.

For Example:-

A PCI card can be inserted into a PCI slot on a motherboard, providing additional I/O ports on the back of a Computer.

→ The PCI local bus is the general standard for a PC expansion bus.

→ The PCI architecture also known as "Conventional PCI" was designed by Intel in the year 1992.



→ Many desktoppe ceres two to five PCI Cards.

→ PCI Originally operated at 33 MHz using 32-bit wide path. It supports 133 Mbps for data transfer rate.

→ An upgrade 64-bit, 66 MHz standard was created for faster datatransfer rate.

→ In 1998 IBM, HP, and Compaq introduced PCI-X (PCI extended) which was backward Compatible with PCI

→ the 133 MHz PCI-X interface supports data rate upto 1064 MBps.

→ So Both PCI and PCI-X were Combined by PCI Express Card was introduced in 2004.

→ PCI Cards use 47 pins to Connect the PCI bus without intervention of cpu.

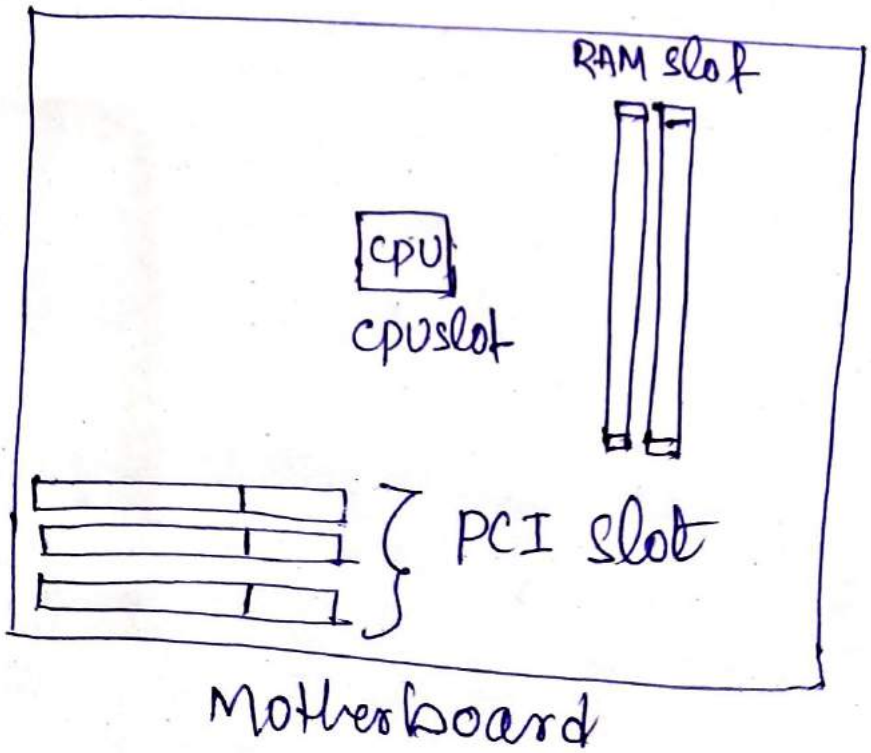
→ the PCI bus is able to work with so few pins because the hardware multiplexing, which means the device sends more than one signal over a single pin.

→ PCI supports devices the use either 5 volt or 3.3 volts from SMPS.

How it works?

→ During the startup the Operating System searches for all PCI bus. to maintain information about resources needed for each devices.

→ the operating system communicates with each devices and assign the system resources including memory, interrupt request and alloted Input/Output space etc.



6.5) SCSI

→ It stands for Small Computer System Interface.

→ It is a Computer interface used primarily for high-speed hard drives.

→ SCSI supports faster data transfer rates than the commonly used IDE storage interface.

→ It also supports daisy-chaining device which means several SCSI hard drives can be connected to a single SCSI interface with little no decreare in performance.

Types of SCSI

There are 12 types of SCSI interface Sudhasi-

1) SCSI-1: → It is uses on 8-bit bus,
→ It supports 4 Mbps data transfer speed.

2) SCSI-2: → It uses 50-pins and supports multiple devices.
→ It is a most commonly used SCSI device.
→ It supports 5Mbps datatransfer speed.

3) Wide SCSI: → It uses wider cable (68 pins) to supports 16-bit data transfer.
→ Data transfer speed is around 6 Mbps.

4) Fast SCSI: → It uses 8-bit pin but doubles the clock rate to support data transfer speeds of 10Mbps.

5) fast wide SCSI: → It uses a 16-bit bus and supports data transfer speeds of 20 Mbps.

6) Ultra SCSI: → It uses 8-bit bus.
→ It supports data rate of 20 Mbps

7) SCSI-3: → It uses 16-bit bus.
→ It supports data rates of 40 Mbps.
→ It also called Ultra Wide SCSI.

8) Ultra 2 SCSI: → It uses 8-bit bus.
→ It supports data transfer rate of 40 Mbps.

9) Wide Ultra 2 SCSI: → It uses 16-bit bus.
→ It supports data rates of 80 mbps.
→ It is also known as Ultra-80

10) Ultra 3 SCSI: → It uses 16-bit bus, it supports data ~~rate~~ transfer rate of 160 Mbps.
→ It is also known as Ultra-160

11) Ultra-320 SCSI: → It uses 16-bit bus.
→ It supports data transfer speeds of 320 Mbps.

12) Ultra-640 SCSI: → It uses 16-bit bus and supports data transfer of 640 Mbps.

Now SCSI is still used for some high performance equipment, newer interface have large replaced SCSI on certain application.

Ex:- firewire and USB2.0 has commonly used for connecting external hard drives.

→ Serial ATA or SATA is now used as fast interface for internal hard drives.

6.6) USB

→ It stands for universal Serial bus.
→ It is a plug and play interface that allows a computer to communicate with peripherals and other devices.
→ USB is a common interface that enables communication between peripheral devices and a host Controller such as personal Computer.
→ USB connects peripheral devices such as:-
* Digital Camera
* External drive.
* iPod or other MP3 players
* keyboard
* keypad
* Microphone
* Mouse
* printer
* joystick
* Scanner
* Smartphone
* Webcams
* Tablet
* Pendrive (flash drive) etc
→ USB interface are found on every Computer devices such as PC, Laptop, Smart phones, Cameras, flash memory etc.

→ USB provides a very simple and effective way for connecting devices as gives a good results.

→ USB is also used to connect via Controller for Convenient use, specially to OTG (on-the-go).

USB Connector types

→ USB Connectors comes in different shapes and sizes such as:-

1) Normal-USB
2) Mini-USB
3) Micro-USB
4) USB Type-C

Normal-USB

→ It is used with pendrive, keyboard, mouse.

Mini-USB

→ It is also known as mini-B is used with digital Cameras and Computer peripherals.

Micro-USB

→ It is replace of mini-USB.

→ It has two varieties such as Micro-A and Micro-B.

→ It is used to connect computer peripherals video games and Charging for Smartphones.

USB Type-C

→ It is the upgrading from Micro-USB named as USB type-C.

→ Now a days used to Charged smartphone and data transfer to PC.

→ Android Smartphone uses USB type-C for connecting USB connected devices.

USB transfer Speed

It is 4 types such as:-

USB 1.x

→ It supports data transfer rate of 12 Mbps
→ It is introduced in the year 2000.

USB 2.0

→ It is also known as Hi-speed USB.
→ It supports data transfer rate of 60 Mbps.
→ It is introduced in the year 2001

USB 3.0

→ It is also known as SuperSpeed USB.
→ It supports data transfer rate of 640 Mbps.
→ It is introduced in the year 2011.

USB 3.1

→ It is also known as SuperSpeed+.
→ It was made available on July-31, 2013.
→ It supported data transfer rate of 10 Gbps.

3rd Sem 16/11/2020 Sub:- TH1 (CSA)

Unit-7 Parallel Processing

7.1) Parallel Processing

→ Instead of processing each instruction sequentially, we use a different technique is called Parallel Processing.

→ What is parallel processing?

Parallel processing is a technique which enables the system to perform concurrent data processing to achive faster execution time.

→ It is a method in computing environment of running two or more processor (CPU's) to handle each separate parts of computer to perform overall task.

→ The whole program is breaking up different different parts of task called blocks, so multiple processors will help to reduce the amount of time to run a program.

→ All modern Computer System has more than one processor can perform parallel processing.

→ Todays Computers has works on multi-Core processors ~~Comp~~ was concurrently running.

→ Parallel Processing System may have two or more ALU's and should be able to execute two or more instructions at the same time.

→ The main purpose of parallel processing is to speed up the Computer processing capability and increase its throughput.

N:B

Throughput:- It is the number of instructions that can be executed in a unit of time.

→ Multi-Core processor are the Silicon IC chip that Contain two or more processor for better performance of Computer.

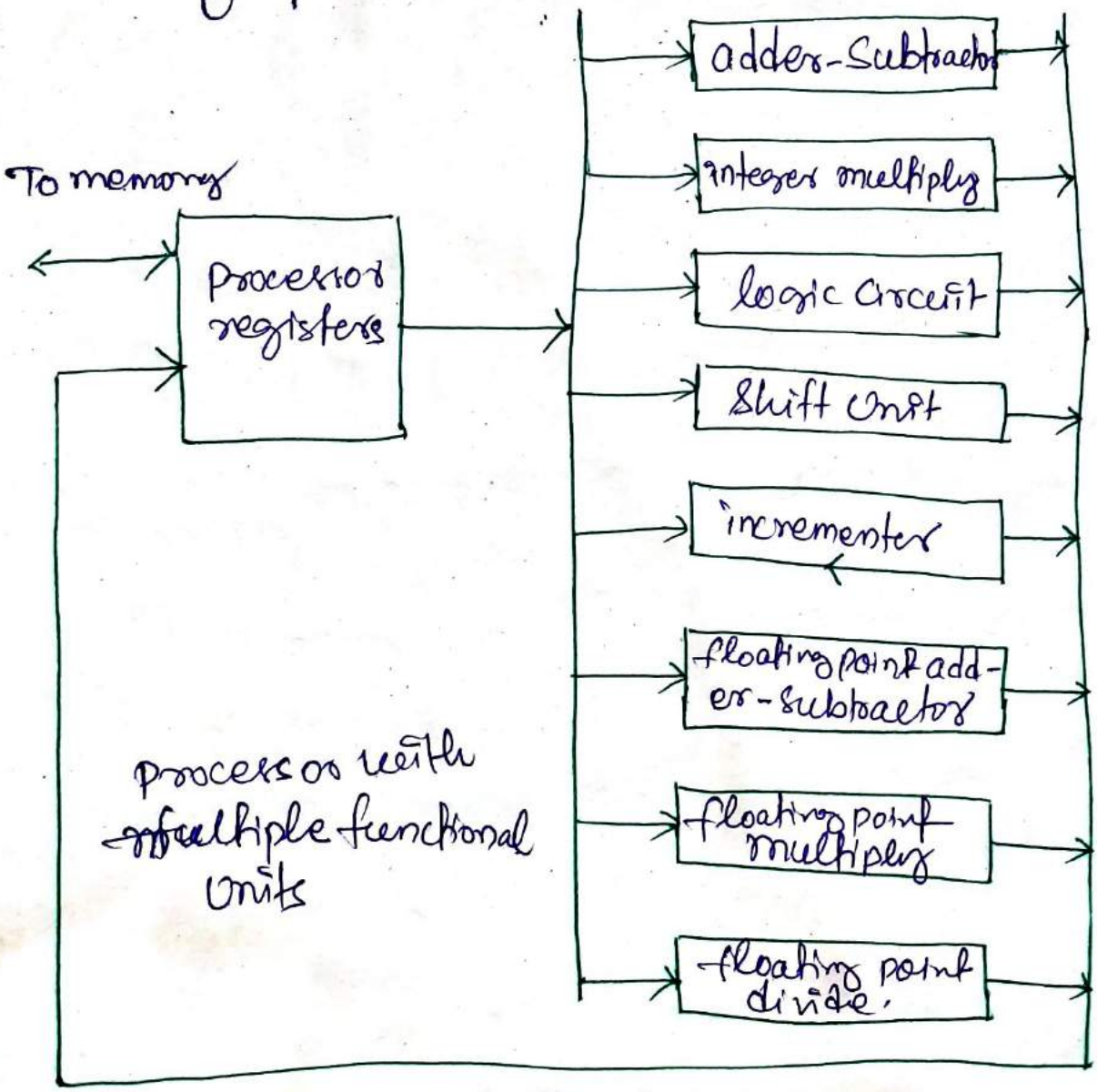
→ Most Computers may have two-four Cores which can be increase upto 12 Cores for faster executing of Computer.

→ Parallel processing is Commonly used to perform complex task and enhance the Computation speed up of Computer to achive the fast processing.

→ Parallel processing can be viewed from various levels of complexity.

→ At the lowest level, we distinguish between parallel and serial operations by the type of registers used.

→ At higher levels of complexity, parallel processing can be achieved by using multiple functional units that perform many operations simultaneously.



3rd Sem 18/11/2020 Sub:- TH1 (CSA)

Continue from Parallel processing....

→ Parallel Computing is breaking up a task into smaller pieces and executing those pieces at the same time.

→ Each pieces executing on their own processors or on a set of computers that have been networked together.

Example for Single processor & parallel process,

Equation

$$x = (4 \times 5) + (1 \times 6) + (5 \times 3)$$

On a single processor, the steps needed to calculate a value of x might look like:

Step1: $x = 20 + (1 \times 6) + (5 \times 3)$

Step2: $x = 20 + 6 + (5 \times 3)$

Step3: $x = 20 + 6 + 15$

Step4: $x = 41$

In a parallel processing, with 3-processors

Step 1: $x = 20 + 6 + 15$

Step 2: $x = 41$

→ It breaking the task down into pieces and

execute those pieces simultaneously.

Performance characteristics

→ Parallel processing is increase the speed of Computer.

→ If we use single processor for solve the equation to calculate the value of x, then it required four steps.

→ If we use parallel CPU for that equation to calculate value of x, then it required only two steps.

→ So parallel processing increase the performance of CPU.

Types of Parallel Processing

→ there are four types of parallel processing, two of the most commonly used types such as SIMD and MIMD.

1) SIMD

→ It stands for Single Instruction Multiple Data stream.

→ It is a type of parallel processing in which a Computer will have two or more processor follow the same instruction set.

→ Each processor handles different data.

2) MIMD

→ It stands for Multiple Instruction Multiple Data.

→ It is another type of Common parallel Processing in which Computer will have two or more of its own processor will get data from separate data streams.

3) SISD:

→ It stands for Single Instruction Single Data.

→ A Computer can Containing a control unit, processor Unit and a memory unit.

→ Instructions are executed sequentially

→ It can be achived by pipelining or multiple functional units.

4) MISD

→ It stands for Multiple Instruction Single Data Stream.

→ It consists of Single Computer Containing multiple processors Connected with multiple Control units and a Common memory unit.

→ It can process several instructions over Single data stream Simultaneously.

3rd Sem 20/11/20 & 21/11/2020 Sub:- THI(CSA)

Pipelining

→ Pipelining is a technology where multiple instructions are overlapped during execution.

→ It is a process of arrangement of hardware elements of the CPU such that its overall performance is increased.

→ Pipeline process can execute more than one instruction simultaneously in a pipelined processor.

→ Pipelining technology makes the hardware utilization and provides maximizing throughput of Computer.

→ Pipeline is divided into stages and these stages are connected with one another to form a pipe like structure.

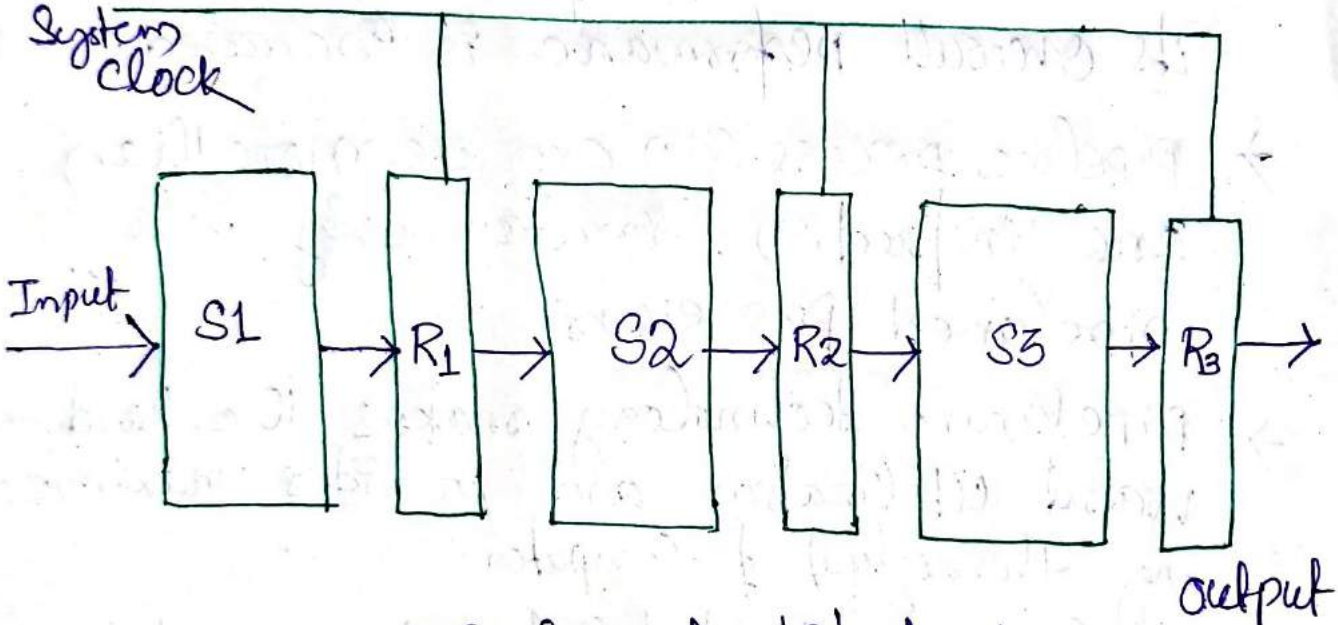
→ Instructions enter from one end and exit from another end.

→ Pipelining increases the overall instruction throughput makes a better Computer for faster execution.

→ 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 for performs operations on it.

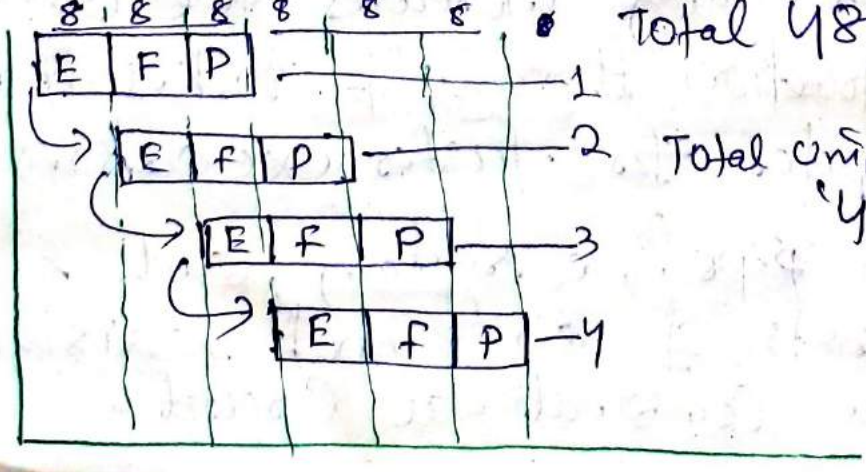
→ The output of Combinational Circuit is applied to the input register of the next segment



(Pipeline Architecture)

Ex:- In a Car manufacturing industry all assembly parts can perform by robotic arm one stage by another.

Stage 1:- Engine setup (E)
Stage 2:- Fitting parts (F)
Stage 3:- Paint Car (P)



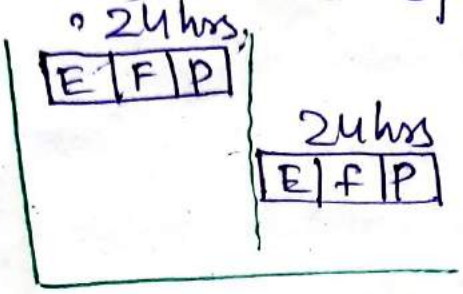
Total 48 hrs working

Total units of car '4'

→ Let us Consider stage-1(E), stage-2(F) & stage-3(P) and each stage take 8 hrs to complet its operation

→ In non-pipeline method one operation is Completed then next operation started.

(Non-pipeline)



Total car unit = "2"
Total 48 hrs working.

→ In pipeline method all operation Continue as it required timeline. Such as:-

→ first stage-1 takes 8 hrs for Complet its operation then they doesnot wait again, they started stage-1 for next unit of car, and so on.

→ So in same time taken by pipeline & non-pipeline which is 48 hrs.

→ In 48 hrs pipeline method manufactured Car unit is 4 but non-pipeline it is 2.

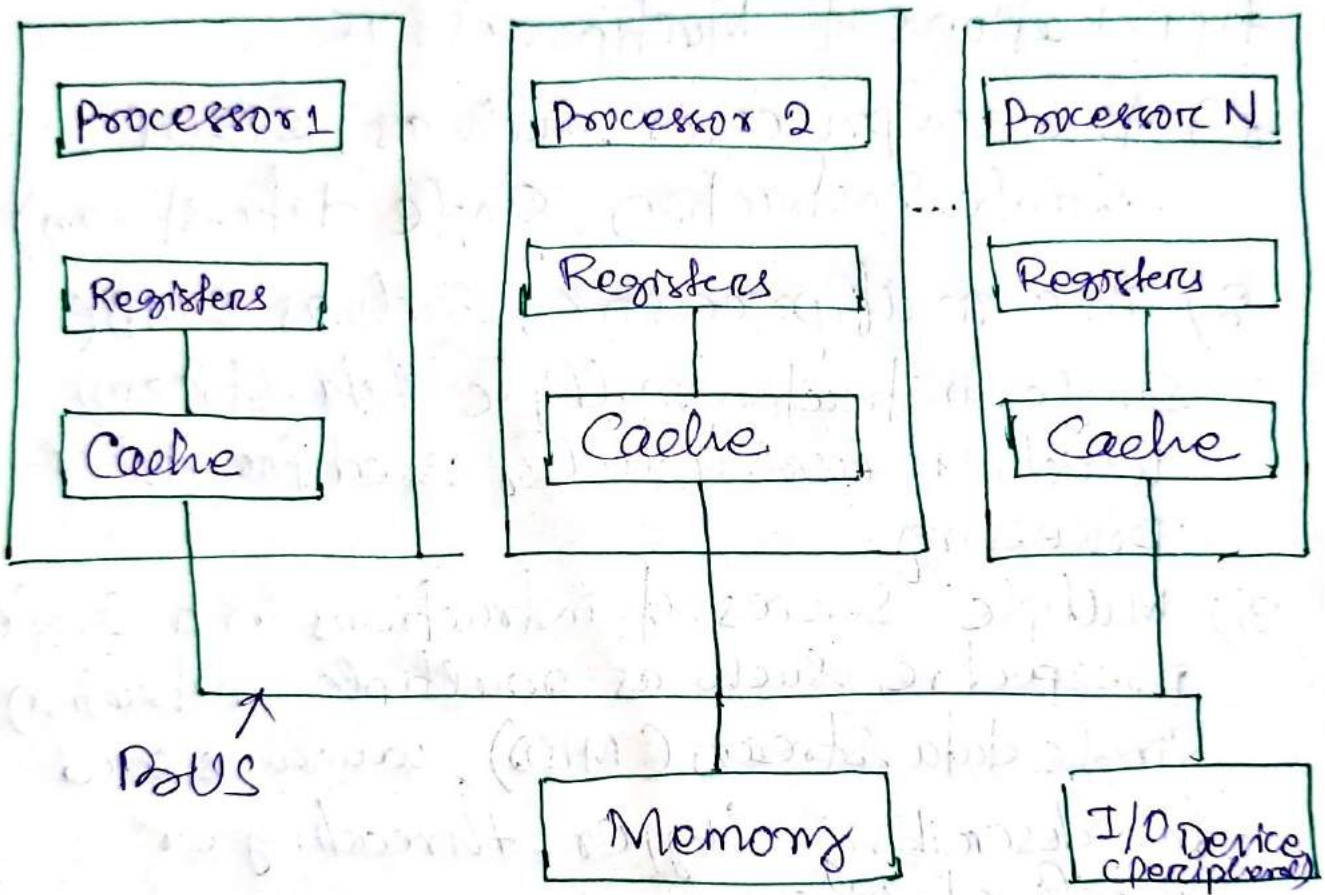
→ So pipeline method in computer makes faster.

Speed of pipeline = number of units

$$S(P) = n$$

7.3) Multiprocessor

→ A multiprocessor is a Computer System with two or more Central processing Units (CPUs), with each one sharing the common memory as well as peripherals. for simultaneous processing of programs.



(Multiprocessor System)

→ The main aim of using multiprocessor is to boost the execution speed of Computer System.

→ A multiprocessor System improve Computing speeds, performance and cost-effectiveness as well as to provide enhanced availability and reliability.

→ Most Computer Systems are Single processor System i.e they only have one processor, however multiprocessor or parallel Systems are multiple processor working parallel that share the Computer Clock, memory, bus & peripheral devices etc.

Applications of Multiprocessor

1) As a uniprocessor, such as SISD (Single Instruction, Single data stream)
2) As a multiprocessor, such as SIMD (Single instruction multiple data stream) which is usually used for vector processing.
3) Multiple Series of instructions in a Single perspective, such as multiple instruction Single data stream (MISD), which is used for describing hyper-threading or pipelined processor.
4) Inside a Single System for executing multiple instructions in multiple prespective such as (MIMD), multiple instruction, multiple data streams.

Benifits of a Multiprocessor

* Enhanced performance.
* Multiple applications.
* Multi-tasking inside an application.
* High throughput and responsiveness.
* Hardware sharing among CPU's

3rd Sem 23/12/2020 Sub:- M1(CSE)

Continue from Multiprocessor...

Types of Multiprocessor

→ There are mainly two types of multiprocessor i.e ① Symmetric ② Asymmetric

1) Symmetric Multiprocessor

→ In this type of multiprocessor System, each processor contains a similar copy of the operating System and they all communicate with each other.

→ All processors are in peer-to-peer (P2P) relationship so there is no master-slave relationship.

Ex:- Unix for Multimax Computer.

2) Asymmetric Multiprocessor

→ In this type of multiprocessor System each processor is given a predefined task.

→ It has a master processor that gives instruction to all other processor for execution.

→ Asymmetric multiprocessor System contains a master-slave relationship

Ex:- Intel Multiprocessor.

Advantages of Multiprocessor System

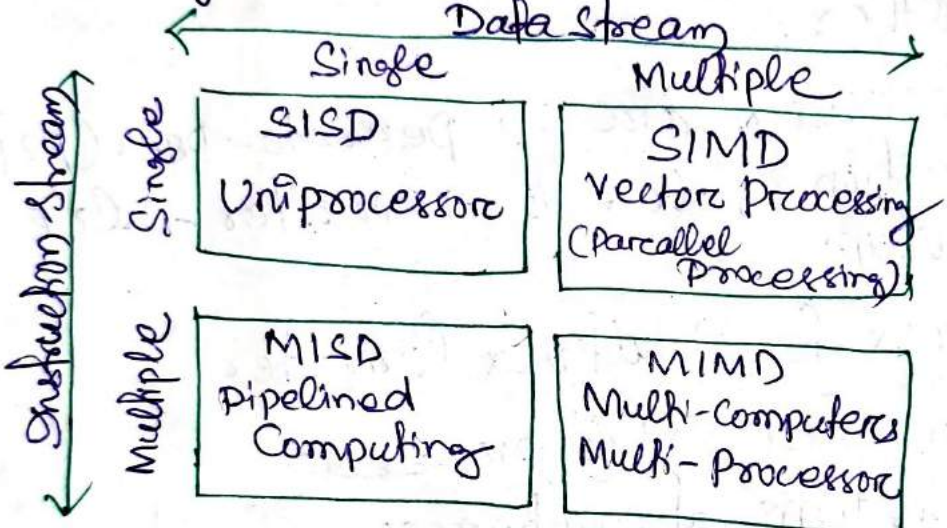
* More reliable System.
* Enhanced Throughput
* More Economic System (low cost)

Disadvantages of Multiprocessor System

* Increased Expense
* Complicated Operating System Needed.

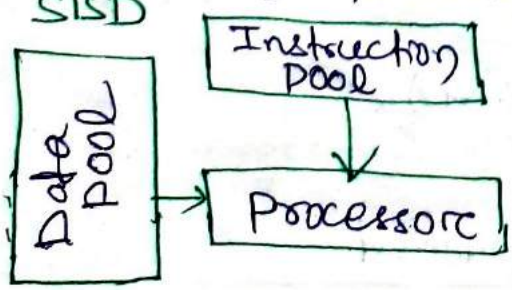
7.4) Flynn's Classification

→ Flynn's classification is also called flynn's taxonomy.
→ It is based on parallel Computing.
→ Based on the number of instructions and data stream that can be processed simultaneously, Computing system are classified into four major Categories.
→ Flynn's classification divides Computer into four major categories such as; -



1) Single-Instruction, Single-data (SISD) System: -

→ It is a uniprocessor system which is capable of executing a single instruction operating on a single data stream.
→ In SISD system instructions are processed in a sequential basis and all instructions stored in primary memory.



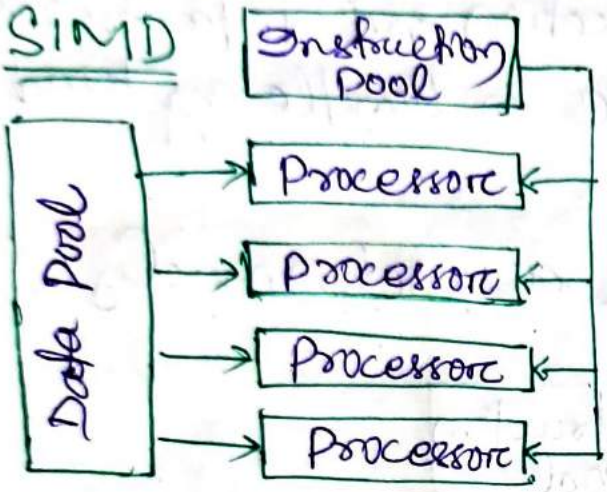
Ex: - IBM PC, Workstation

2) Single-Instruction, Multiple-data (SIMD) system

→ SIMD system is capable of executing the same instruction on all the CPUs but operating on different data streams.

→ SIMD model involves in Scientific Computing for vector and matrix operations.

SIMD

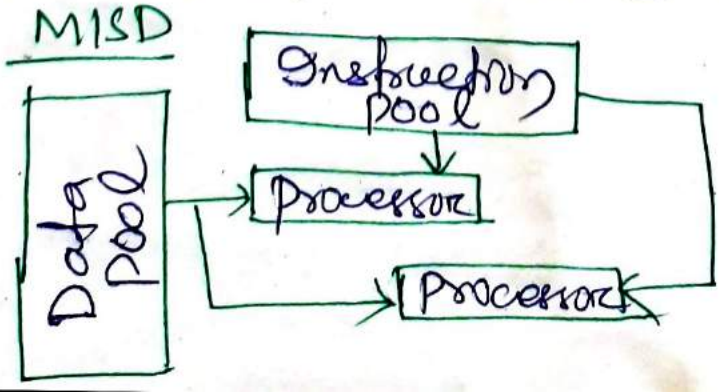


Ex:- SIMD system is Cray's vector processing machine.

3) Multi-Instruction, Single-data (MISD) System:-

→ An MISD Computing system is a multiprocessor system capable of executing different instructions on different processors for execution but all of them perform on the same dataset.

MISD



4) Multiple-Instruction, multiple-data (MIMD) System:-

→ It is a multiprocessor system.

→ MIMD system enables of executing multiple instruction on multiple datasets.

→ Each program execution in this model has separate instruction and data streams So MIMD is capable to handle any kind of application.

→ MIMD machine work asynchronously.

MIMD