

# Computer Organization Architecture

## Question:

Q.1) Explain logic gate with neat diagram.

## Answer:

- The logic gates are the main structural part of a digital system.
- Logic Gates are a block of hardware that produces signals of binary 1 or 0 when input logic requirements are satisfied.
- Each gate has a distinct graphic symbol, and its operation can be described by means of algebraic expressions.
- The seven basic logic gates includes: AND, OR, XOR, NOT, NAND, NOR, and XNOR.
- The relationship between the input-output binary variables for each gate can be represented in tabular form by a truth table.
- Each gate has one or two binary input variables designated by A and B and one binary output variable designated by x.

**AND Gate:**



Truth Table:

A	B	x
0	0	0
0	1	0
1	0	0
1	1	1

Algebraic Function:  $x = AB$

**OR Gate:**



Truth Table:

A	B	x
0	0	0
0	1	1
1	0	1
1	1	1

Algebraic Function:  $x = A + B$

**NAND Gate:**



Truth Table:

A	B	x
0	0	1
0	1	1
1	0	1
1	1	0

Algebraic Function:  $x = (AB)'$

**NOR Gate:**



Truth Table:

A	B	x
0	0	1
0	1	0
1	0	0
1	1	0

Algebraic Function:  $x = (A+B)'$

**XOR Gate:**



Truth Table:

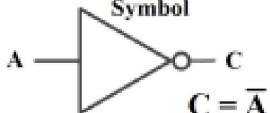
A	B	x
0	0	0
0	1	1
1	0	1
1	1	0

Algebraic Function:  $x = A \oplus B$   
or  
 $x = A'B + AB'$

**NOT Gate**

**NOT Gate**

Symbol



$C = \bar{A}$

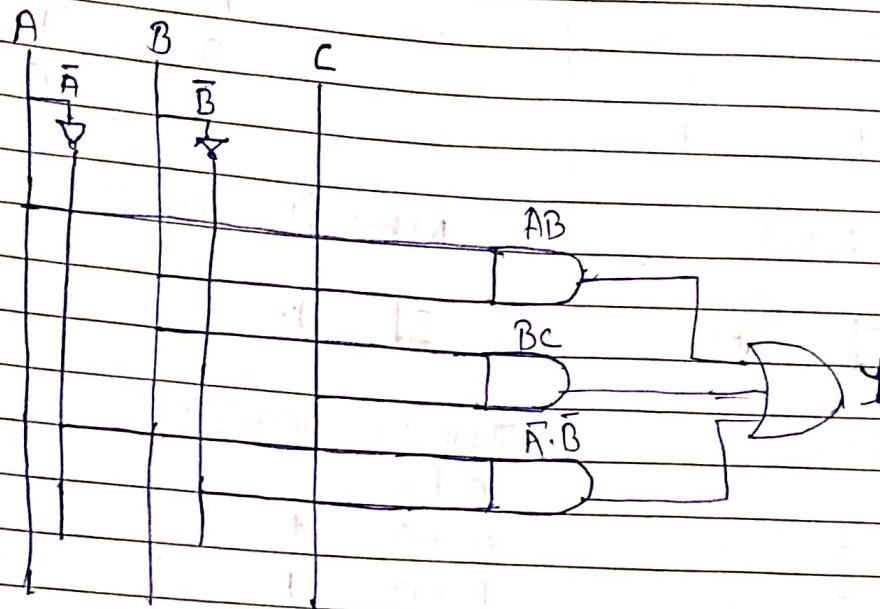
**Truth Table**

INPUT	OUTPUT
A	NOT A
0	1
1	0

Q.2

Draw the Circuit

$$Y = \bar{A}B + BC + \bar{A} \cdot \bar{B}$$



Q.3

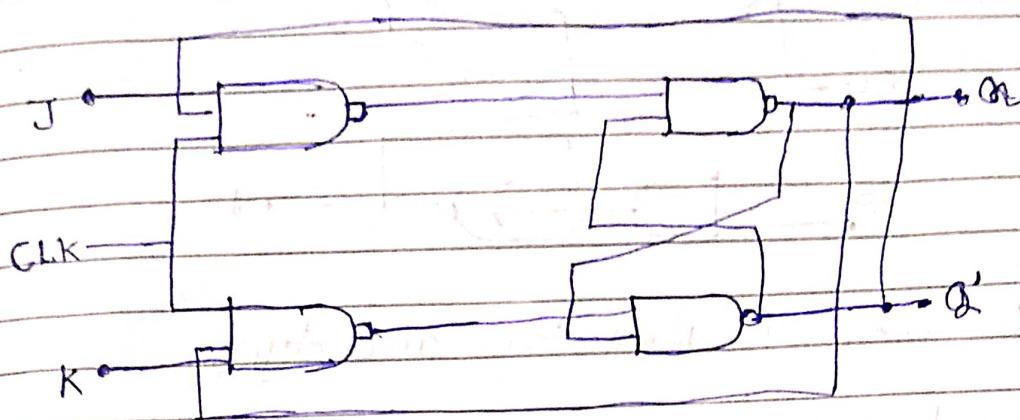
Explain SR &amp; JK flip-flop

\* JK

- It is one kind of Sequential Logic Circuit.
- Which stores binary information in bitwise manner.
- It consists of two inputs and two outputs.
- Input are Set(J) & Reset(K)
- JK flip flop has two modes of operation. Which are Synchronous mode and Asynchronous.

PAGE NO.	/ /
DATE	/ /

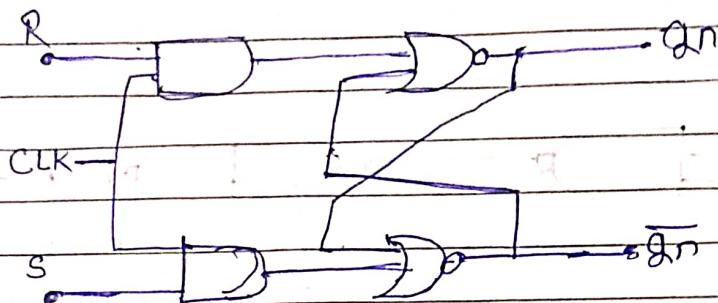
- The JK Flip flop diagram represents the basic structure which consists of clock (CLK), clear (CLR) and Preset (PR).



\* SR

- It is a Flip-Flop with two inputs.
- One is S and other is R.
- S here Stands for Set  
R here Stands for Reset
- Set basically indicates set the flip flop which means output 1
- Reset indicates resetting the flip flop which means output 0.

- Hence clock pulse is supplied to operate this flop flop, hence it is clocked flip flop.



#### Q.4 Explain Registers and types.

- Registers are used to quickly accept, store, and transfer data and instructions that are being immediately by the CPU. There are various types of Registers those are used for various purpose.

##### ① Accumulator Register (AR) :-

The Register is used for storing the results those are produced by the system. When the CPU will generate some results after the processing then all the results will be stored into the AC Register.

##### ② Memory Data Register (MDR) :-

\* MDR is the register of a computer's control unit that contains the data to be stored in the computer storage (e.g. RAM).

\* MDR hold the information before it goes to the

PAGE No.	
DATE	/ /

decoder.

### ③ Index Register (IR) :-

An index register in a computer's CPU is a processor register used for modifying operand addresses during the run of a program.

### ④ Memory Buffer Register :-

This register holds the contents of data or instruction read from, or written in memory.

### ⑤ Data Register (DR) :-

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

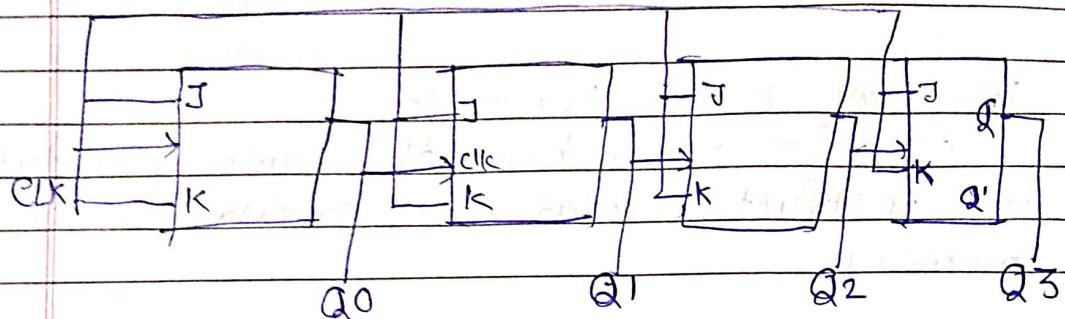
## Q.5 Explain Counters and types.

- Counter is a digital device and the output of the counter includes a predefined state based on the clock pulse applications. The output of the counter can be used to count the number of pulses.

PAGE NO.	
DATE	/ /

### ① Asynchronous Counter :-

In asynchronous counter, we don't use universal clock, only FF is driven by main clock and clock I/P of rest of the following FF is driven by O/P of previous FF.



Clock

Q<sub>0</sub>Q<sub>1</sub>Q<sub>2</sub>Q<sub>3</sub>

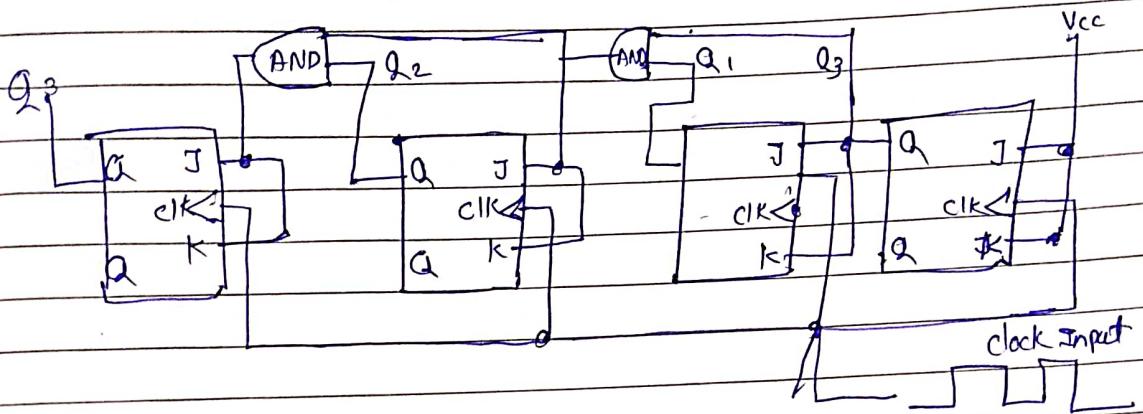
Q<sub>0</sub> is changing as soon as rising edge of clock pulse is encountered.

Q<sub>1</sub> is changing when rising edge of Q<sub>0</sub> is encountered.

## ② Synchronous Counter :-

It Can Operate on higher Frequency than ~~asynchronous~~ Counter.

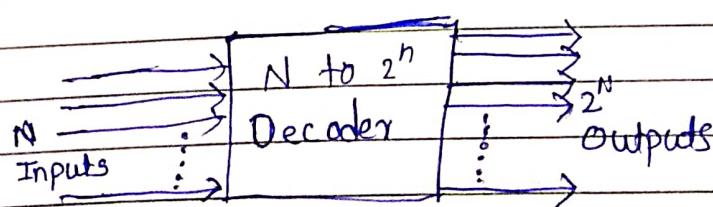
It does not have cumulative delay because of same clock is given to each Flip Flop. It is also called as parallel counter.



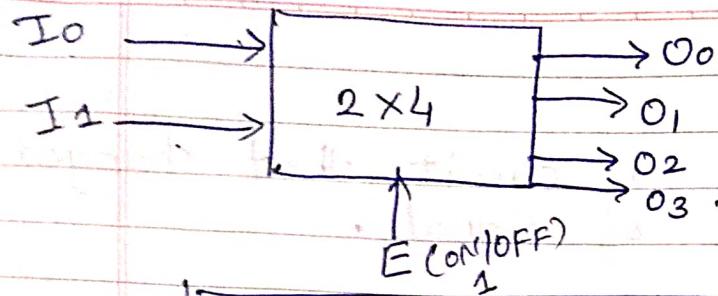
## Q.6. Define Decoder

A decoder is a circuit that changes code into set of signals.

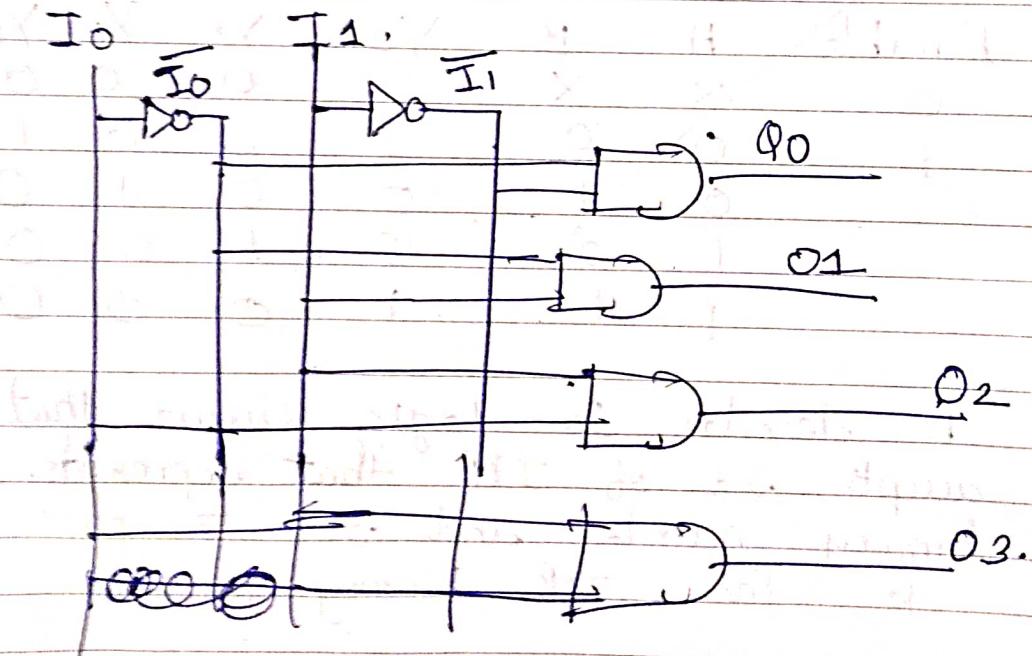
A decoder is logic circuit that accepts Set of I/P that represent a binary number and activate O/P to the I/P binary number has n Inputs and enable line 4 2<sup>n</sup> O/P line



$$2^{\frac{8}{2}} = 4/2 = 2$$

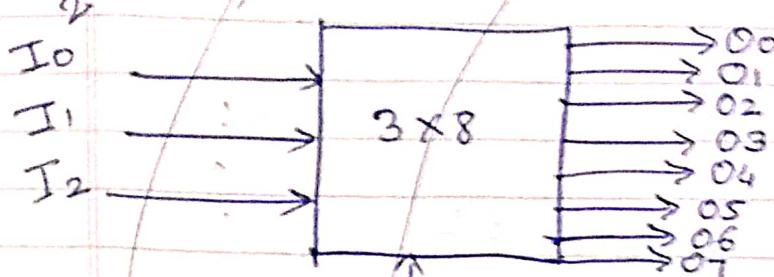


IIP	OIP				
Io	I1	O0	O1	O2	O3
0	0	10	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1



3 to 8 Decoder.

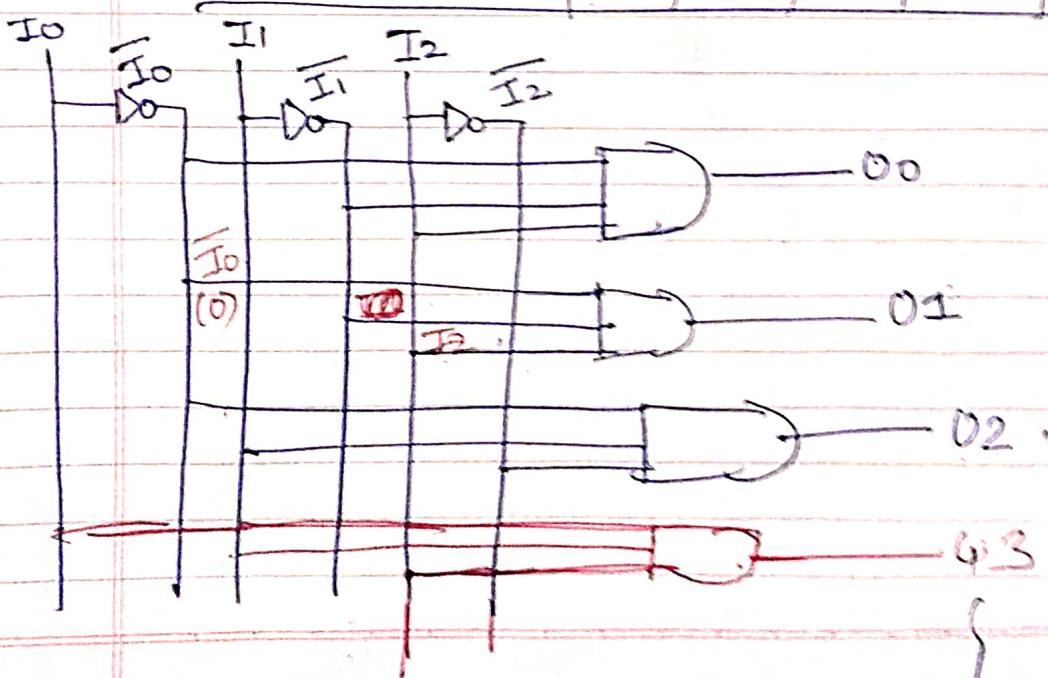
$$\begin{array}{l} 3 \times 2 = 4 \\ 4 \times 2 = 2 \\ 2 \times 2 = 1 \end{array}$$



3 : 8 line  
decoder also  
known as  
binary to octal  
decoder.

~~Explain~~  
Block Diagram

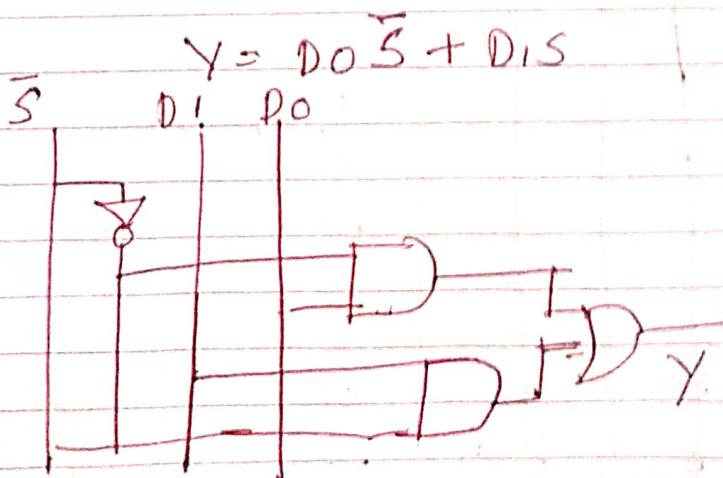
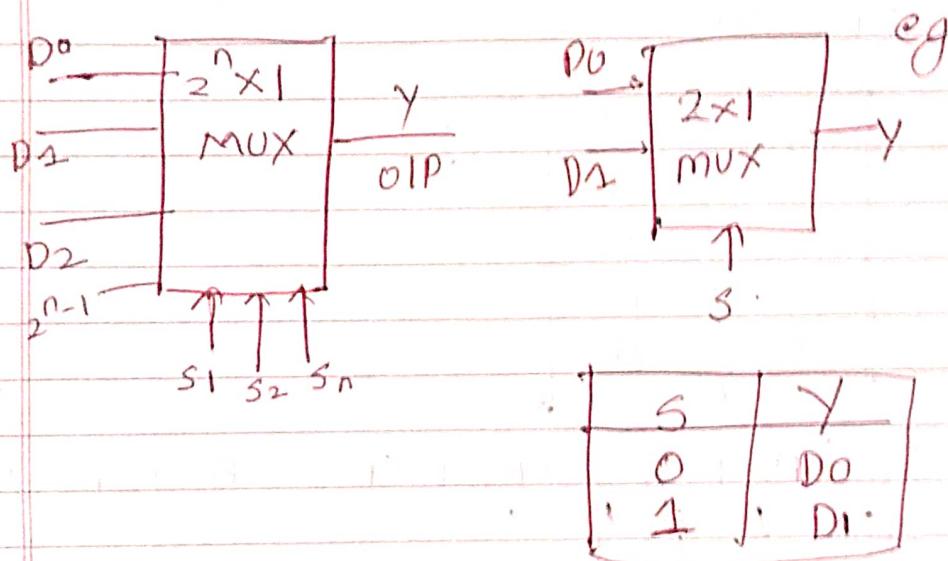
$I_0$	$I_1$	$I_2$	$O_0$	$O_1$	$O_2$	$O_3$	$O_4$	$O_5$	$O_6$	$O_7$
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	1	0	0	0	0	0	0	1





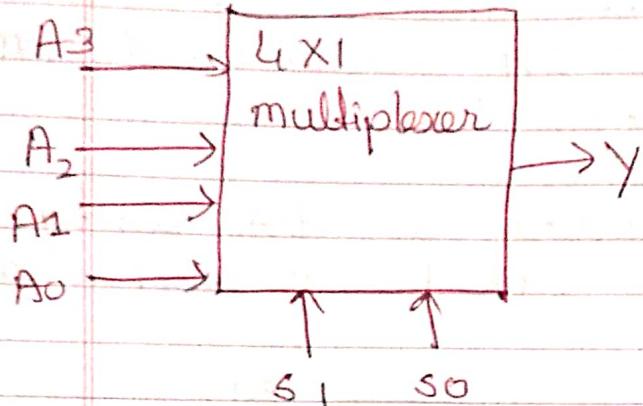
### A. Multiplexer

- combinational logic circuit used to select only one SIP among several SIP based on selection lines
- It is also called as data selector.
- For mux there can be n selection lines.





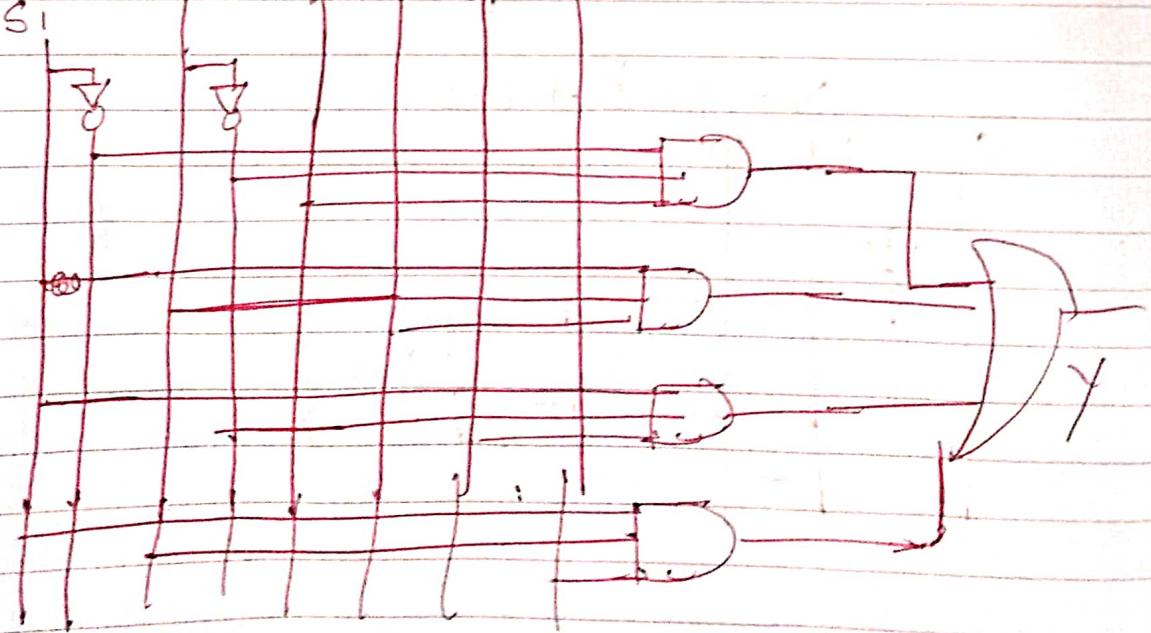
(2)

4:1 multiplexer

S1	S0	Y
0	0	A0
0	1	A1
1	0	A2
1	1	A3

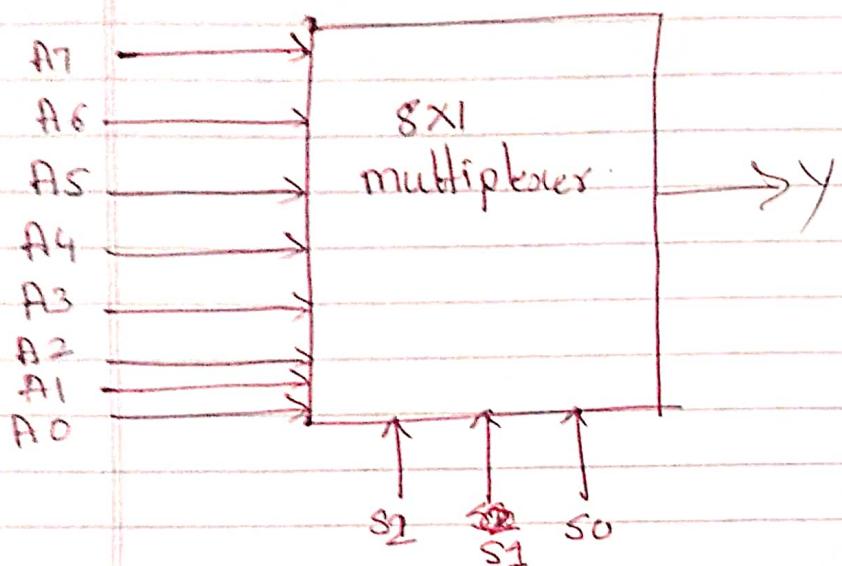
$$Y = \overline{S_1} \cdot \overline{S_0} \cdot A_0 + \overline{S_1} \cdot S_0 \cdot A_1 + S_1 \cdot \overline{S_0} \cdot A_2 + S_1 \cdot S_0 \cdot A_3$$

$$\overline{S_1} \quad \overline{S_0} \quad A_0 \quad A_1 \quad A_2 \quad A_3.$$





②  $8 \times 1$  multiplier.



$S_2$	$S_1$	$S_0$	$Y$
0	0	0	$A_0$
0	0	1	$A_1$
0	1	0	$A_2$
0	1	1	$A_3$
1	0	0	$A_4$
1	0	1	$A_5$
1	1	0	$A_6$
1	1	1	$A_7$

$$\begin{aligned}
 Y = & \bar{S}_2 \cdot \bar{S}_1 \cdot \bar{S}_0 \cdot A_0 + \\
 & \bar{S}_2 \cdot \bar{S}_1 \cdot S_0 \cdot A_1 + \\
 & \bar{S}_2 \cdot S_1 \cdot \bar{S}_0 \cdot A_2 + \\
 & \bar{S}_2 \cdot S_1 \cdot S_0 \cdot A_3 + \\
 & S_2 \cdot \bar{S}_1 \cdot \bar{S}_0 \cdot A_4 + \\
 & S_2 \cdot \bar{S}_1 \cdot S_0 \cdot A_5 + \\
 & S_2 \cdot S_1 \cdot \bar{S}_0 \cdot A_6 + \\
 & S_2 \cdot S_1 \cdot S_0 \cdot A_7
 \end{aligned}$$

## Assignment 2

**Q.1.** What are Data types in Computer Architecture?

- ① Data Types in CA is Binary information is sorted in memory or processor register.
- ② Registers containing either data or control information.
- ③ Data are numbers and other binary coded information.
- ④ Control information is a bit or group of bits used to specify sequence of command signals.
- ⑤ Data types found in registers of digital computers :

1. Numbers

2. Letters

3. Other discrete symbols

**Q.2.** Define complement and its types.

Complements are used in digital circuits, because it is faster to subtract by adding complements than by performing true subtractions. The binary complement of a number is created by reversing all bits and adding 1. The carry from the high-order position is eliminated.

Types of complement:

① one's complement

→ Reversing all bits

② 2's complement

→ Reversing all bits except the RHS  
bits upto first 1.

Q. 3. Store -40 in 16-bit memory location using 2's complement representation.

① Change to binary ~~101000~~

② Add 10 zero's to 16-bit

~~000000000000101000~~

③ Make every 0 on LHS to 1.

Because the sign bit is 'negative'.

And keep the entire bits as it is still the first 1.

④ Result is ~~11111111011000~~

Q. 4. How to convert decimal to binary

$(169.9)_{10}$

① Divide  $(169)_{10}$  successively by 2 until the quotient is 0

2	169.
2	84 1
2	42 0
2	21 0
2	10 1
2	5 0
2	2 1
2	1 0
0	1

② Read from bottom to top

③ Result  $\rightarrow 10101001$

#### Q.5. Explain Binary codes.

Binary code is a system of writing numerals that assign 4-digit binary code into each digit 0 through 9 in decimal (base 10) numerals.

	8	4	2	1
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	0	1	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0

Binary code classification:

- (1) weighted code
- (2) non-weighted code
- (3) reflective code
- (4) sequential code
- (5) Alphanumeric code
- (6) Error detecting and correcting  
→ Hamming code

Q.6. Explain Error detection code with suitable examples.

- (1) Error detection code can be applied to data units of any length.
- (2) A Error detection code is a binary code & detects digital error during transmission.
- (3) The binary information is transferred from one location to another location through some communication medium.
- (4) To detect error in the received message we add some extra bits to actual data.
- (5) It is used to detect single bit error hamming code structure.

- ⑥ All bits positions that are power of 2 are marked as parity bits (1, 2, 4, 8) and other bits for data.

## Error Detection Techniques

1. Parity check
2. checksum
3. cyclic Redundancy check (CRC)

## Hamming code structure

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	P <sub>4</sub>	D <sub>3</sub>	P <sub>2</sub>	P <sub>1</sub>
----------------	----------------	----------------	----------------	----------------	----------------	----------------

Example:

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	P <sub>4</sub>	D <sub>3</sub>	P <sub>2</sub>	P <sub>1</sub>
1	0	1	0	1	1	0

At receiver end bit

(1, 3, 5, 7) (2, 3, 6, 7) and (4, 5, 6)  
are checked for even parity.

P <sub>1</sub>	D <sub>3</sub>	D <sub>5</sub>	D <sub>7</sub>	P <sub>2</sub>	D <sub>3</sub>	D <sub>6</sub>	D <sub>7</sub>	P <sub>4</sub>	D <sub>5</sub>	D <sub>6</sub>	D <sub>1</sub>
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

$$P_1 = 0 \quad P_2 = 1 \quad P_4 = 0$$

Q.7. Explain Register Transfer Language.

Very fast computer memory.  
It is used to store data instruction in execution.  
It is a group of flipflops with each flip flop capable of storing one-bit of information.

Registers

Accumulators
General Purpose Registers
Special Purpose Registers

- MAR (Memory address register)
- MBR (Memory buffer register)
- PC (Program counter)
- IR (Index register)

Microoperations  
The operations executed on data stored in registers.

Register Transfer language is the symbolic notation used to describe the microoperation transfers among registers and memory.

It is not executed by computer. Register transfer means availability of hardware logic circuit that can perform state microoperation & transfer the result of operation to some other register.

# Basic symbols for Register Transfer language

Symbols	Description	Examples
1. Capital letters	denotes Registers and Numerals	MAR, R2
2. Parenthesis ()	denotes the port of register	R2(0-9), R2(2)
3. Arrow	denotes transfer of information	R2 $\leftarrow$ R1
4. colon	denotes terms and control function	P :-
5. comma	separation of micro operations	AEP, BFX
Q.8.	Explain types of micro operations	
	there are 4 types of micro operations:	
1.	Register Transfer micro operation	
2.	Arithmetic Micro operation	
3.	Logic micro operation	
4.	Shift micro operation	

## 1. Register Transfer Micro operation.

The symbol notation used to describe micro operation transfer among registers are called Register Transfer Language.

Transfer binary information from one register to another. Doesn't change the information content when the binary information moves from source register to destination register.

Other three types of micro operations change the information content during the transfer.

## 2. Arithmetic micro operations.

Perf

performs arithmetic operation on numeric data stored in registers.

### Type of Arithmetic micro operations

① Addition

$$R_3 \leftarrow R_1 + R_2$$

② Subtraction

$$R_3 \leftarrow R_1 - R_2 + 1$$

③ Increment

$$0111 \rightarrow 0110$$

#### ④ Decrement

$$\textcircled{a} \quad 001 \rightarrow 110$$

#### 5. Logical micro operation

These are binary micro-operations carried out on the register bits. These procedures treat each bit as a binary variable and consider it separately.

Consider the X-OR micro operation with the contents of R<sub>1</sub> and R<sub>2</sub> registers.

$$P: R_1 \leftarrow R_1 \oplus R_2$$

#### 4. Shift micro operation

The serial shifts are the important different types of micro-operations that means we can move the register's contents to the left or right. The serial input shifts a bit to the rightmost position in the shift left operation, and a bit to the leftmost position in the shift right action.

$R \leftarrow \text{shl } R$  } shift right  
 $R \leftarrow \text{shr } R$  }

$R \leftarrow \text{rl } R$  } circular right  
 $R \leftarrow \text{rl } L$  }

$R \leftarrow \text{ashl } R$  } Arithmetic shift  
 $R \leftarrow \text{ashr } R$  } Right register

Q.9. Determine the value of parity bits with code word 1101.

The value of parity bits is determined by a sequence of bits that are alternatively checks and skip.

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	= P <sub>1</sub>	D <sub>3</sub>	P <sub>2</sub>	= P <sub>1</sub>
----------------	----------------	----------------	------------------	----------------	----------------	------------------

1	1	0	-	1	-	-
---	---	---	---	---	---	---

Step 1: P<sub>1</sub> 2 bit check

checks 1 bit

skip 1 bit and did = 1 010

Step 2: P<sub>2</sub> 2 bit check

skip 1 bit and did = 1 010

Step 3: P<sub>3</sub> 2 bit check

skip 2 bit and did = 1 010

Step 4: P<sub>4</sub> 2 bit check

skip 2 bit

check 2 bit

skip 2 bit

(2, 3, 6, 7, 10, 11)

Step 3: P<sub>4,9</sub>

check 4 bit

skip 4 bit

(4, 5, 6, 7, 12, 13, 14, 15, ...)

P <sub>1</sub>	D <sub>3</sub>	D <sub>5</sub>	D <sub>7</sub>	P <sub>2</sub>	D <sub>3</sub>	D <sub>6</sub>	D <sub>7</sub>	P <sub>4</sub>	D <sub>5</sub>	D <sub>6</sub>	D <sub>7</sub>
1	0	1	1	0	1	0	1	1	0	1	1

$P_1 = 0$      $P_2 = 1$      $P_4 = 0$

1	1	0	0	1	1	1	0
D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	P <sub>4</sub>	D <sub>3</sub>	P <sub>2</sub>	P <sub>1</sub>	

Q.10. 7-bit hamming code is received as 1011011. Assume even parity and state whether the received code is parity or odd.

1	0	1	1	0	1	1	0
D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	P <sub>4</sub>	D <sub>3</sub>	P <sub>2</sub>	P <sub>1</sub>	

## Detecting error

P <sub>1</sub>	P <sub>3</sub>	D <sub>5</sub>	D <sub>7</sub>
1	0	1	1

P<sub>1</sub> = odd Parity Error

P <sub>2</sub>	P <sub>3</sub>	D <sub>6</sub>	D <sub>7</sub>
1	0	0	1

P<sub>2</sub> = Even Parity  
No Error

P <sub>4</sub>	D <sub>5</sub>	D <sub>6</sub>	D <sub>7</sub>
1	1	0	1

P<sub>4</sub> = odd Parity Error

$$\text{Bit Error} = 1 + 4 = 5$$

The correct data is

1	0	0	1	0	1	1	
D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	P <sub>4</sub>	D <sub>3</sub>	P <sub>2</sub>	P <sub>1</sub>	

## Assignment - 03

**Q. 1. Define memory hierarchy?**

- In Computer Organisation, the memory hierarchy separates Computer Storage into a hierarchy based on response time. Since response time, complexity, and capacity are related, the levels may be distinguished by their performance and controlling technologies.

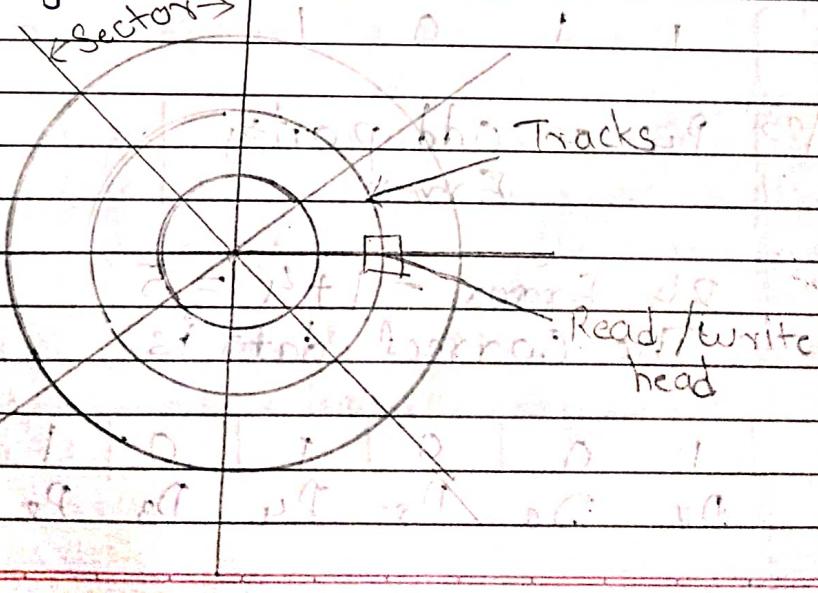
**Q. 2. Explain the term auxiliary memory?**

- Auxiliary memory is known as lowest cost, highest capacity and slowest access storage in computer system.

It is where program and data are kept for long term storage or when not in immediate.

Eg. ① Magnetic Tapes

② Magnetic Disks



PAGE NO.	
DATE	/ /

Q.3. Explain the term of Main Memory?

→ The main memory is central storage unit in a computer system.

It is large and fast memory which is used to store programs & data during run time applications.

The integrated circuit for main memory are

- ① RAM integrated circuit chips.
- ② ROM integrated circuit chips.

Q.4. Explain the term Cache Memory?

→ The data or content of main memory that are used frequently by CPU are stored in Cache memory so that processor can easily access that data in shorter time.

Whenever CPU needs to access memory it first checks the Cache Memory

If data is not found in Cache memory then CPU moves into main memory

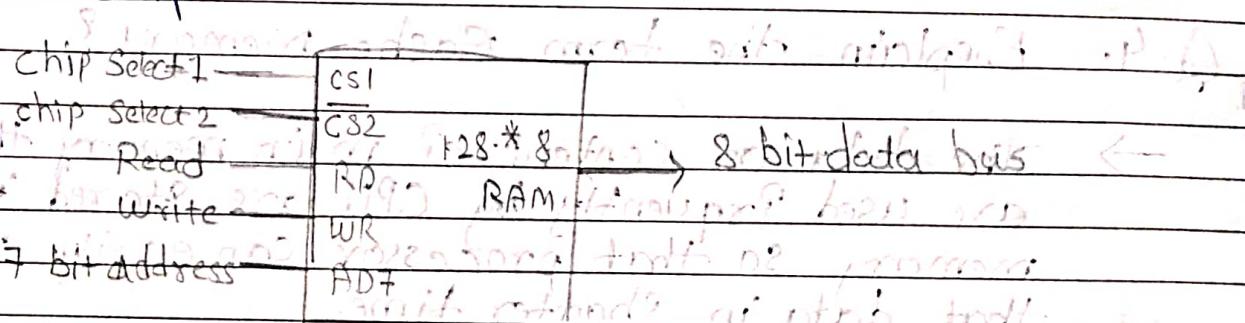
Page No.	
Date	/ /

Q.S. Define RAM & ROM integrated circuit chips?

→ ① The RAM integrated circuit chips are further classified into two possible operating modes. Static and Dynamic.

The static RAM is easy to use and takes less time performing read and write operations as compared to dynamic RAM.

The following block diagram demonstrates the chip interconnection in a  $128 \times 8$  RAM chip.



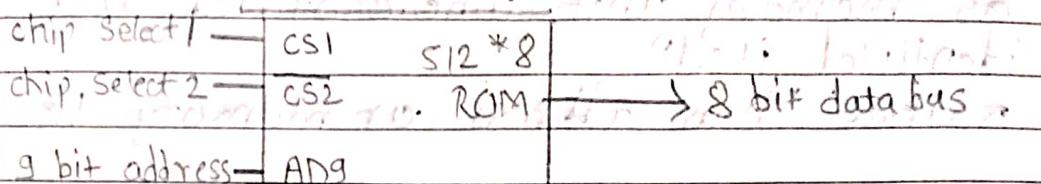
② ROM integrated Circuit chips.

The primary function of the bootstrap loader program is to start the computer software operating when power is turned on.

The primary component of the main memory is RAM integrated circuit chips.

PAGE NO.	/ /
DATE	

The following block diagram demonstrates the chip interconnection in a  $512 \times 8$  ROM chip.



Q.6. Explain auxiliary memory in detail.

→ An Auxiliary memory is known as the lowest-cost, highest-capacity and slowest-access storage in a computer system. It is where programs and data are kept for long-term storage or when not in immediate use. The most common examples of auxiliary memories are magnetic tapes and magnetic disks.

\* Magnetic Tape :- Magnetic tape is a storage medium that allows data archiving, collection, and backup for different kind of data. The magnetic tape is constructed using a plastic strip coated with a magnetic recording medium.

\* Magnetic Disks :- A magnetic disk is a type of memory constructed using a circular plate of metal or plastic coated with magnetized materials. Usually, both sides of the disks are used to carry out read/write operations.

Page No.	/ /
Date	/ /

Q.7. Explain Associative memory in detail?

- Associative memory can be considered as memory unit where stored data can be identified itself rather than address or memory location.

When a write operation is performed on associative memory no address or memory location is given to the word.

The memory itself is capable of finding an empty unused location to store the word.

When word is to be read the content of word is specified and matched word are located in memory.

Q.8 Explain in brief what main memory and its types?

- The main memory is the fundamental storage unit in a computer system. It is a sequentially large and quick memory and saves programs and information during computer operations.

The technology that makes the main memory work is based on Semiconductor integrated circuits.

There are two types

- ① Random Access Memory

PAGE NO.	
DATE	11

The term Random Access Memory or RAM is typically used to refer to memory that is easily read from and written to by the microprocessor. For a memory to be called random access, it should be possible to access such as tapes or hard drives where the data is accessed sequentially.

### Read-Only Memory

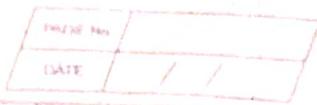
In each computer system, there should be a segment of memory that is fixed and unaffected by power failure. This type of memory is known as Read-Only Memory or ROM.

Q.9 Write Advantages & Disadvantages of Cache memory.

#### Advantages

#### Disadvantages

- |  |   |
|--|---|
| (1) It makes the computer system more speedy.                                  | (1) It is a volatile memory.                                  |
| (2) Cache memory is high speed semiconductor memory that can speed up the CPU. | (2) Cache data is stored as long as the computer power is on. |
| (3) It stores data, instructions & information for a limited period.           | (3) The increased chip  |



- (4) It consumes less access time as compared to main memory.
- (4) Cache Stores the data temporarily for faster access.
- (5) It is generally placed between the primary cache and the rest of the memory.
- (5) The storage capacity is completely in use.

Q.10. Explain Virtual Memory in brief?

→ Virtual memory is the partition of digital memory from physical memory. This partition supports large virtual memory for programmers when only limited physical memory is available.

Virtual memory can give programmers the illusion that they have a very high memory although the computer has a very small main memory.

Ans: Characteristics of Cache:

- 1. Cache stores frequently used data.
- 2. Cache is fast and small.
- 3. Cache is located between processor and main memory.

# Assignment 4

## Unit - III

DATE : / /  
PAGE NO. :

Q.1. What are computer registers.

→ Computer registers are a type of computer memory used to quickly accept, store and transfer data and instructions that the CPU needs immediately.

The register used by CPU is called processor register.

A processor register may hold an instruction or storage address, or any data (such as bit sequence or individual characters).

Q.2. Define Operands & Operands.

→ There are two parts of the instruction codes.

1]

Opcode is the first part of an instruction that tells the computer what function to perform and is also called Operation codes.

2] Operand

Second part of an instruction is the data to be operated on is called operand.

memory

instruction
Program
Operand
data

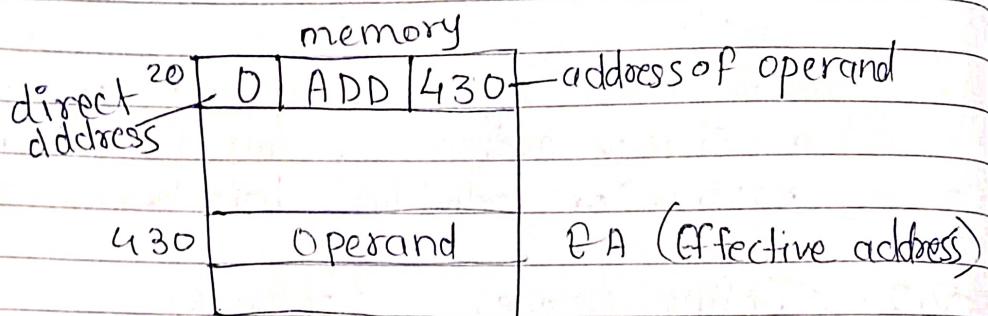
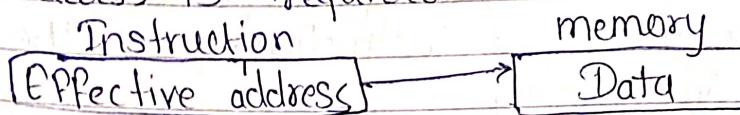
Operand is another second part of instruction, which indicates the computer system where to find the data or instructions or where to store the data or instructions.

Q.3. Define direct address and indirect address.

→ Direct address - is the address of operand (effective address). In this the address field

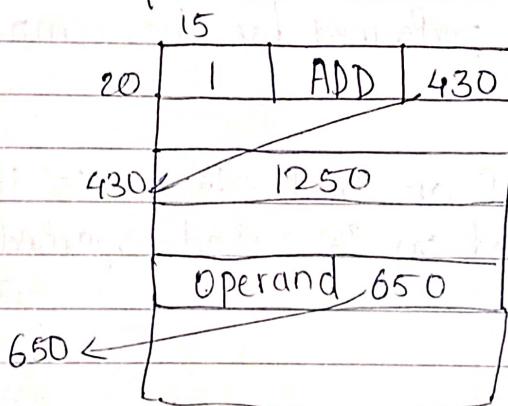
DATE: / /  
PAGE NO.:

in the instruction contains the effective address of the operand and no intermediate memory access is required.



Indirect address - address of memory word in which address of operand is found.

It requires two memory access.



- Q. 4) List the type of computer registers.
- 1) Program counters - it holds address of instructions.
- PC
- 2) Arithmetic registers, It addresses from memory
- AR

DATE : / /
PAGE NO. :

3] Instruction register - used to hold instructions

15 0

**[IR]**

4] Temporary register - holds temporary data.

15 0

**[TR]**

5] Outer register - holds input character.

7 0

**[ONPR]**

6] Inner register - holds output character

7 0

**[INPR]**

7] Data register - holds operands.

15 0

**[DR]**

8] Process register - Accumulator

15 0

**[AC]**

Q.5. Perform computer arithmetic:

1]  $1010 + 0111$

2]  $1000 - 0111$

3]  $101 \times 11$

4]  $101 \div 10100$

$$\begin{array}{r} \rightarrow 1] \begin{array}{r} 1010 \\ + 0111 \\ \hline 10001 \end{array} \Rightarrow 10 + 0 + 0 + 1 + 1 \\ \text{carry bit} \end{array}$$

$$\begin{array}{r} \rightarrow 2] \begin{array}{r} 10 \\ - 0111 \\ \hline 0001 \end{array} \Rightarrow 8 - 0 - 1 - 1 - 1 = 1 \\ \text{0 = borrow} \end{array}$$

3) 
$$\begin{array}{r} & \overset{8}{\cancel{4}} \overset{2}{\cancel{1}} \\ \times & 101 \Rightarrow 5 \\ & \underline{11} \Rightarrow 3 \\ & 101 \quad \underline{15} \\ & \underline{\underline{101x}} \\ & 1111 \end{array} \quad \begin{array}{l} |x| = 1 \\ |x0| = 0 \\ 0x1 = 0 \\ 0x0 = 0 \end{array}$$

4) 
$$\begin{array}{r} 1000 \\ \textcircled{101} \quad 101000 \\ - \\ 101 \\ 00 \\ - \\ 00 \\ 00 \end{array}$$

Q.1. Explain in brief microprogrammed control unit.

→ Sequence of micro instruction in microprogramm language.

⇒ Midway between Hardware & Software.

- It generates set of control signals.
- It is easy to design test & implement.
- It is flexible to modify.

Microcontrol unit consist of.

1) Control signals - Group of bits used to select multiplexer.

2) Control variables - Binary variables specify microoperation.

3) Control word - Strings of 1's and 0's representation.

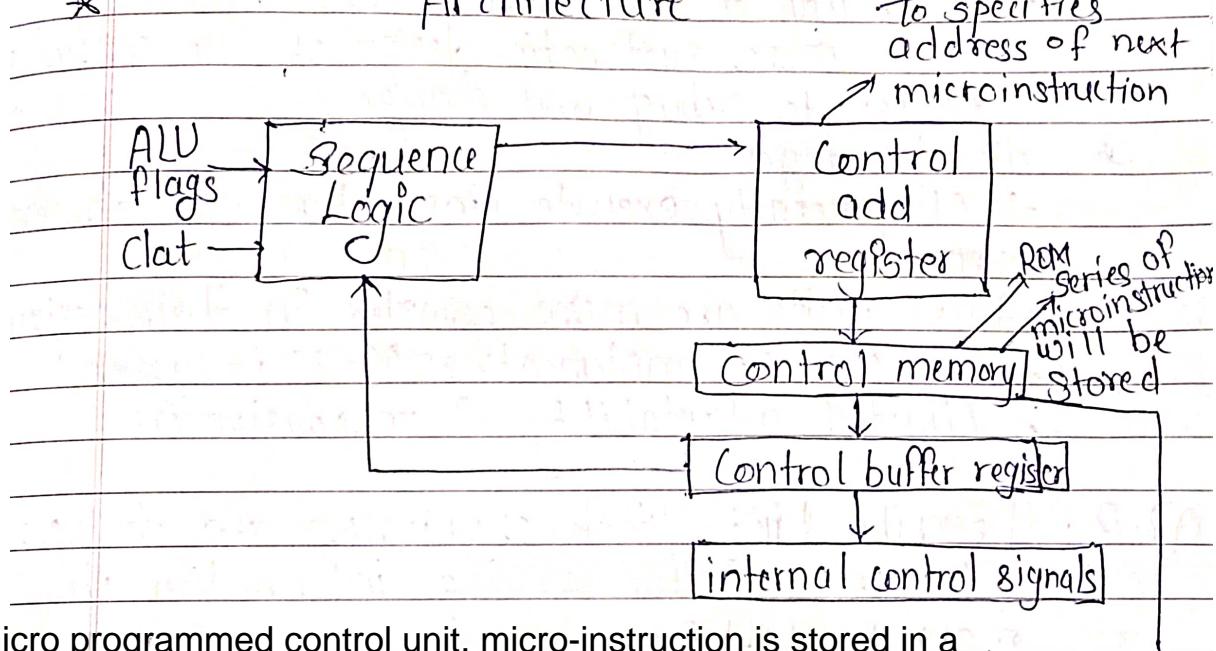
4) Control memory - Memory contains control word.

DATE: / /  
PAGE NO.:

- 5) Micro instructions - Control words stored in control memory, specify control signals for executions of micro operation.

- 6) Micro program - Sequence of microinstruction architecture.

### \* Architecture



\* In micro programmed control unit, micro-instruction is stored in a special memory called control memory.

\* Implemented using programming approach.

\* Sequence is carried out by executing a program consisting of micro-instruction

Microprogrammed Control unit (MCU) is a type of control unit in a computers central processing unit (CPU). Unlike hardwired control units a microprogrammed control unit uses microinstruction stored in memory to control the execution of instructions.

In microprogrammed control unit the control information is stored in the control memory

DATE: / /  
PAGE NO.:

and is programmed to initiate the required sequence of microoperations.

A control unit whose binary control variables are stored in memory is known as a microprogram control unit.

### \* Advantages

- Flexibility to add new instructions.
- It is more systematic design of the control unit.
- easier to debug and change.

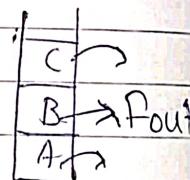
### \* disadvantages-

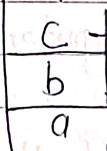
- MCU generally execute instructions at slower speed.
- These units are more complex in their design.
- The cost of implementing MCU is higher.
- Limited adaptability & responsiveness.

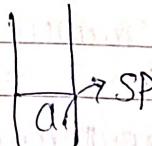
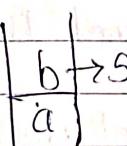
Q. 2. Describe brief Stack organization and its types.

→ Storage device for storing information in manner LIFO.

Stack is memory unit with an address register called stack pointer (SP)



 Stack pointer indicate at top of stack



DATE: / /  
PAGE NO.:

push - insertion  $\Rightarrow$  increment SP  
 pop - deletion  $\Rightarrow$  decrement SP.

Two types of stacks.

- ① Register Stack (Stack depth is limited)
- ② memory Stack (Stack depth is flexible)

I] Register Stack.

Push Operation.

$SP \leftarrow SP + 1$  : increment SP

$m[SP] \leftarrow DR$  : write the stack

$TF(SP=0)$  then ( $full \leftarrow 1$ ) : check if stack is full

$Empty \leftarrow 0$  : mark not empty.

POP

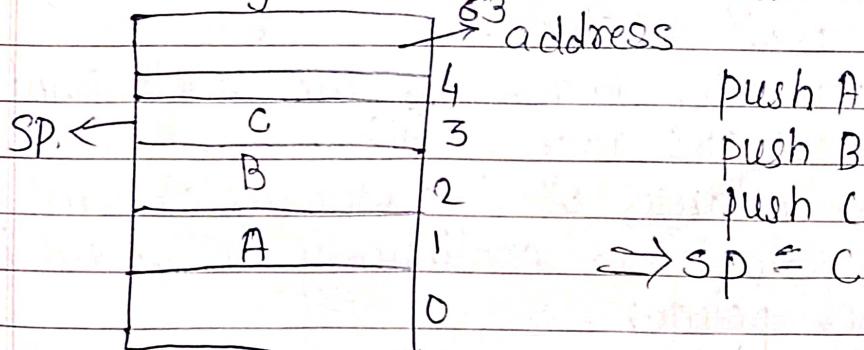
$DR \leftarrow m(SP)$  : Read item from stack

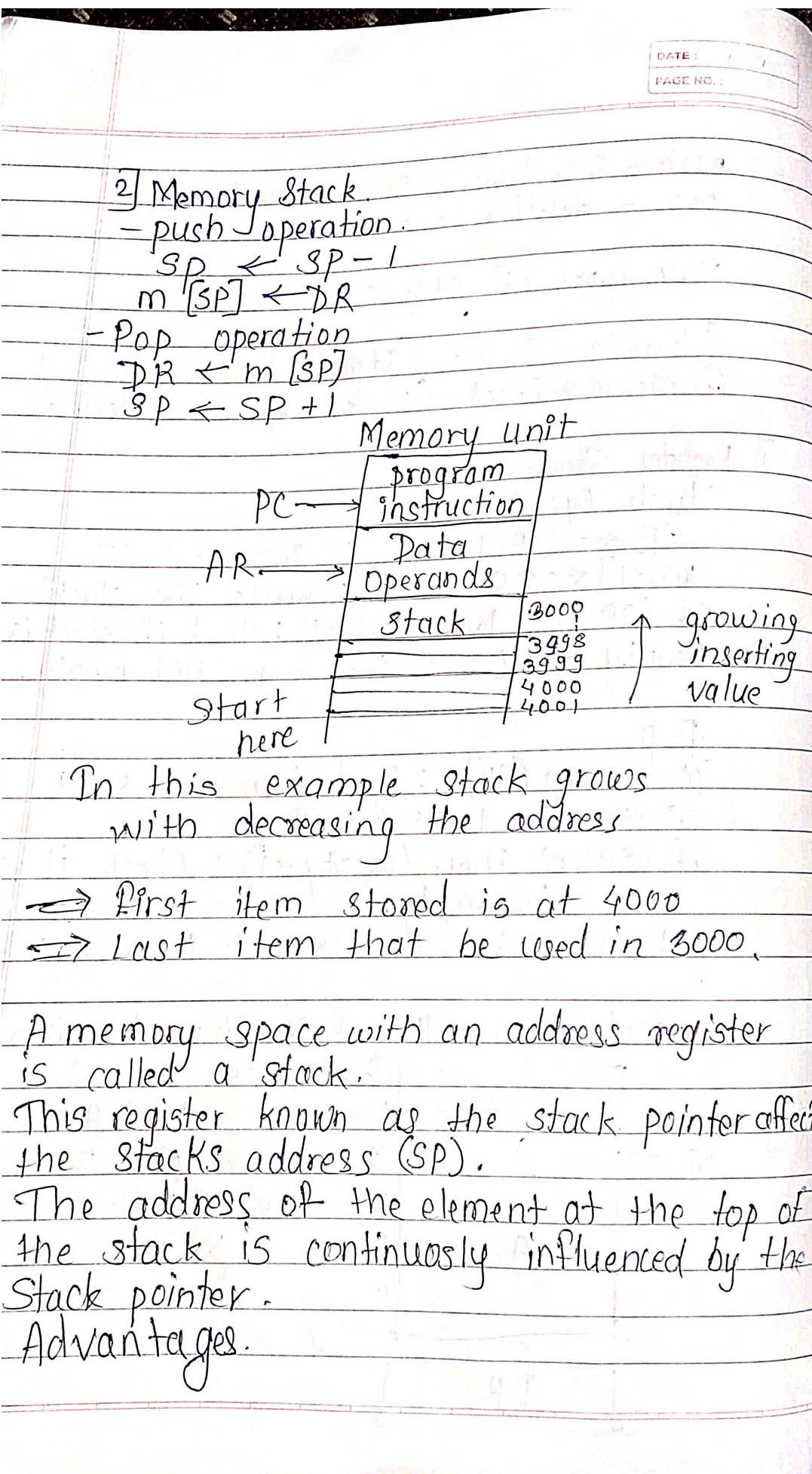
$SP \leftarrow SP - 1$  : dec. SP

$if(SP=0)$  then ( $empty \leftarrow 1$ ) : check if stack is empty.

$full \leftarrow 0$  Stack is not full.

Stack organization of 64-bit word register stack.





- Complex arithmetic statements may be rapidly calculated.

- Instruction execution is rapid because operand data is stored in consecutive memory areas.

- The instructions are minimal since they don't contain an address field.

### \* Disadvantages.

- The size of the program increases when we use a stack.

- Its in memory and memory is slower in several ways than CPU registers.

- It generally has a lesser bandwidth and a longer latency.

- Memory accesses are more difficult to accelerate.

Q.3. Difference between hardwired control and microprogrammed control.

→ Hardwired Control Unit	Microprogrammed Control unit
--------------------------	------------------------------

1) Hardwired control unit generates the control signals needed for the processor using logic circuits. - Microprogrammed control unit generates the control signals with the help of micro instructions stored in control memory.

2) Hardwired control unit is faster when compared to microprogrammed control unit as the required control signals are generated with the help of hardware. - This is slower than the other as micro instructions are used for generating signals here.

3]	Difficult to modify as the control signals that need to be generated are hard wired.	- Easy to modify as the modification need to be done only at the instruction level.
4]	More, costlier as everything has to be realized in terms of logic gates	- Less costlier than hardware control as only micro instructions are used for generating control signals.
5]	It cannot handle complex instructions as the circuit design for it becomes complex	- It can handle complex instructions. - Control signals for many instructions can be generated.
6]	Only limited number of instructions are used due to the hardware implementation	- Used in computer that makes use of Complex Instruction Set computers (CISC)
7]	Used in computer that makes use of reduced instruction set computers (RISC)	- Used in computer that makes use of Complex Instruction Set computers (CISC)
	- Hard to modify	- Easy to modify.
8]	It is more expensive as compared to MCU	- Affordable as compared to HCV.
9]	It faces difficulty in managing the complex instructions because the design of the circuit is also complex	- It can easily manage complex instructions.
10]	It can use limited instructions	- It can generate control signals for many instructions.

DATE: / /  
PAGE NO.:

Q.4. Explain Computer arithmetic with suitable examples.

→ Computer arithmetic is a field of computer science that investigates how computers should represent numbers and perform operations on them.

It includes integer arithmetic, fixed-point arithmetic and the arithmetic this book focuses on: floating point (FP) arithmetic which will be more thoroughly described.

Fast computer arithmetic is important because it has a major effect on computer performance. Addition is used to compute addresses which are usually specified as a base and displacement. An index value incremented in a register is added in addressing arrays.

Types

1] Binary Addition

Rules -

$0$	$0$	$1$	$1$
$+ 0$	$+ 1$	$+ 0$	$+ 1$
$\hline$	$\hline$	$\hline$	$\hline$
$0$	$1$	$1$	$0$

(carry bit)

ex. 1 1

$$\begin{array}{r} 1 0 1 0 \\ + 0 1 1 1 \\ \hline 1 0 0 0 1 \end{array} \Rightarrow 10 + 7 = 17$$

$$\begin{array}{r} 0 1 1 1 \\ - 1 0 1 0 \\ \hline 1 1 1 1 \end{array} \Rightarrow 11 - 10 = 1$$

2] Binary Subtraction

Rules =  $\begin{array}{r} 0 1 1 0 \\ - 0 1 0 1 \\ \hline 1 0 0 1 \end{array} \Rightarrow \text{borrow } 1 0$

$$\begin{array}{r} 0 1 1 0 \\ - 0 1 0 1 \\ \hline 1 0 0 1 \end{array}$$

DATE: / /  
PAGE NO.:

ex.

9	4	2	1
		+	0
- 0	1	0	1
<hr/>			
1 0 0 1			

$\Rightarrow 14$

$\Rightarrow -5$

$\Rightarrow 9$

## 3] Binary multiplication

Rules ↴

1 0 1	$\Rightarrow 5$			
$\times$	1 1	$\Rightarrow 3$		
<hr/>		1 X 1 = 1		
<hr/>		1 0 1	15	1 X 0 = 0
<hr/>		1 0 1 X	15	0 X 1 = 0
<hr/>		1 1 1 1	15	0 X 0 = 0

## 4] Binary division

1 1		
(100)	1 1 0 0	
<hr/>		-
<hr/>		1 0 0
<hr/>		1 0 0
<hr/>		0 0 0

Q.5. Explain RISC architecture with suitable diagram.

→ RISC stands for →

Reduced Instruction Set Computing.

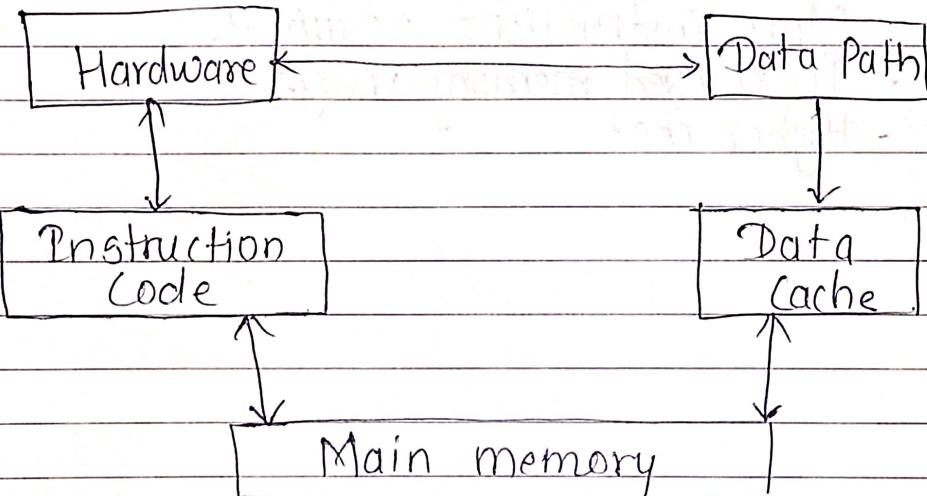
It is a type of microprocessor architecture that utilizes a small highly-optimized set of instructions rather than the highly-specialized set of instructions typically found in other architecture.

It is the way to make hardware simpler

Characteristics -

- Simpler Instruction hence simple instruction decoding.
- Instruction comes undersize of one word.
- Instruction takes a single clock cycle to get executed.
- More general-purpose registers.
- Simple addressing modes.
- Fewer data types.
- A pipeline can be achieved.

Architecture



Hardware - is to execute instruction quickly because more precise & smaller numbers of instruction & large number of register.

Data path - Used to store & manipulate data in computer.

- responsible for managing data within processor.

DATE : / /

PAGE NO. :

Cache - Processor uses cache to reduce access time to the main memory.

- instruction cache is used for retrieving & storing data frequently used instructions.

Data cache - Provides storage for frequently used data from main memory.

Advantages -

- Simpler instructions.
- Faster execution.
- Lower power consumption.

Disadvantages -

- More instructions required.
- Increased memory usage.
- Higher cost.

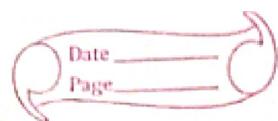
Assignment

Q] Explain peripheral devices?

Ans A peripheral device is an internal or external device that connects directly to a computer or other digital device but does not contribute to the computer's primary function, such as computing. It helps end users access and use the functionalities of a computer. A peripheral device is also called a peripheral, computer peripheral, Input-output device.

2.] Define Interfaces & its types.

Ans Interfaces are tools & concepts that technology developers use as points of interaction between hardware and software components. They help all components within a system communicate with each other via an



I/O system & detailed protocols while also allowing them to function independently.

These are 2 types of Interfaces -

- Hardware Interface - Hardware Interface help various hardware devices connect & communicate with each other. Some common hardware interfaces in computing are -
  - USB
  - Serial interface
  - Parallel Interface
  - PCI & PCI-express interface
  - IEEE 1394 Interface

- Software Interface - It is also called programming interface. They typically control a system's resources, such as its memory, storage space & CPU. Some common software interfaces are -
  - Operating system
  - Simple Mail Transfer Protocol email
  - IP network protocol
  - Software drivers

Q) What are modes of transfer

Ans • Binary information - Binary information received from an external device is usually stored in memory for later processing

• CPU - It merely executes the I/O instructions & may accept the data temporarily, but the ultimate source or destination is in the memory unit.

• Data transfer bet'n the central computer & I/O device may be handled in a variety of modes,

• Some modes use the CPU as an intermediate path, others transfer the data directly to & from the memory unit.

• Data transfer to & from peripherals may be handled in one of three modes -

- Arithmetic Pipeline
- Instruction pipeline
- RISC pipeline

4.] Write features of I/O processor.

Ans 1) The I/O processor is a specialized processor which loads & stores data in memory along with the execution of I/O instruction.

2.) It acts as an interface bet' the system & device.

3.) It involves a sequence of events to execute I/O operations & then store the results in memory.

5.] Explain arithmetic pipeline.

Ans Arithmetic pipeline are mostly used in high speed computers. They are used to implement floating point operations, multiplication of fixed point numbers & similar computations encountered in scientific problems.

6.] Explain in brief Instruction pipeline.

Ans Pipeline processing can occur not only in the data stream but in the instruction stream as well. Most of the digital computers with complex instructions require instruction pipeline to carry out operations like fetch, decode & execute instructions. In general, the computer needs to process each instruction with the following sequence of steps.

1.) Fetch instruction from memory.

2.) Decode the instruction

3.) Calculate the effective address

4.) Fetch the operands from memory

5.) Execute the instruction

6.) Store the result in the proper place.

Each step is executed in a particular segment & 3 are

times when diff<sup>n</sup> segments may take diff<sup>n</sup> time to operate on the incoming information. The organization of an instruction pipeline will be more efficient if the instruction cycle is divided into segments of equal duration. One of the most common example of this type of organization is a four-segment instruction pipeline.

7.) Describe brief about various arithmetic pipeline.

Ans Arithmetic pipeline are mostly used in high speed computer. They are used to implement floating point operations, multiplication of fixed point numbers & similar computations encountered in scientific problems.

To understand the concept of arithmetic pipeline is a more convenient way, let us consider an ex- of a pipeline unit for floating-point addition & subtraction. The inputs to the floating-point adder pipeline are 2 numbers normalized floating-point binary numbers defined as -

$$x = A * 2^a = 0.1100 * 10^3$$

$$y = B * 2^b = 0.1010 * 10^2$$

where A & B are 2 fractions that represent the mantissa & a & b are the exponents. The combined operation of floating-point addition & subtraction is divided into 4 segments, each segment contains the corresponding sub-operation that are shown in the 4 segments are-

1.] Compare the exponents by subtraction

2.] Align the mantissas.

3.] Add or subtract the mantissas.

4.] Normalize the result.

8) Explain vector processing in COT.

Ans Vector processing is a central processing unit that can perform the computer's vector input in individual instruction. It is a complete unit of hardware resources that implements a sequential set of similar data elements in the memory using individual instruction. The scientific & research computations involve many computations which require extensive & high power computers.

Features of Vector Processing -

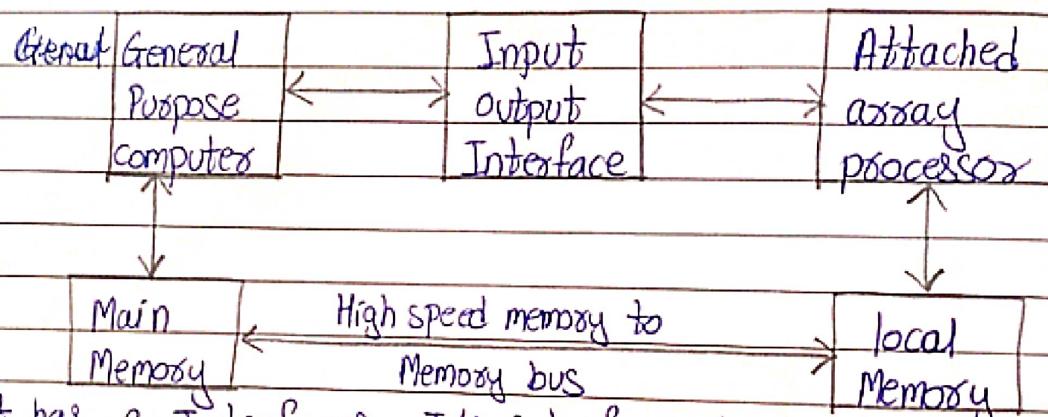
- A vector is a structured set of elements. The elements in a vector are scalar quantity. A vector operand includes an ordered set of  $n$  elements, where  $n$  is known as the length of the vector.
- A vector processor implements better with higher vector because of its foundation delay in a pipeline. Vector processing decrease the overhead related to maintenance of the loop-control variables which creates it more efficient than scalar processing.
- In parallel vector processing more than 2 results are generated per clock cycle, the parallel vector operation are automatically started under the following two circumstances -
  - 1) When successive vector instructions facilitate diff functional units & multiple vector registers.
  - 2) When successive vector instructions use the resulting flow from one vector register as the operand of another operation utilizing a diff' functional unit. This phase is known as chaining.



Q) Describe briefly about various types of array processor with suitable diagram.

Ans Array Processor performs computation on large array of data. There are two types of Array Processors: Attached array Processors and SIMD array Processors.

1) Attached Array Processor - This To improve the performance of the host computer in numerical computational tasks auxiliary processor is attached to it.

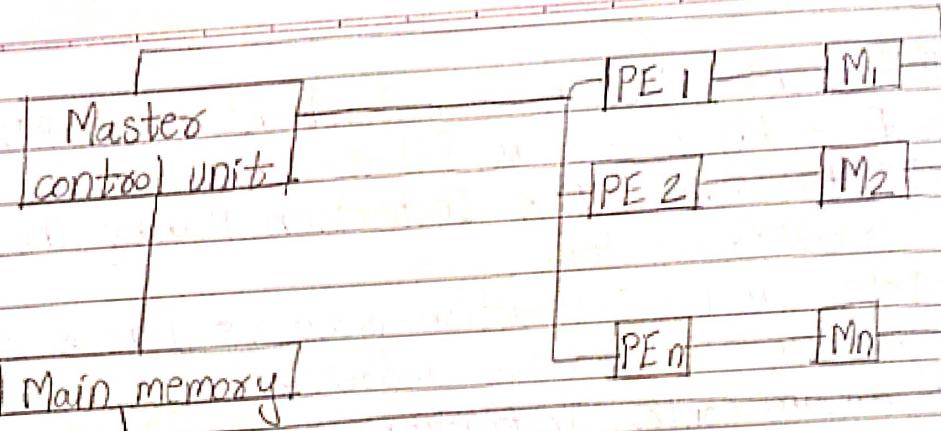


It has 2 Interfaces - I/O interface to a common processor  
• Interface with a local memory -

Here local memory interconnects main memory. Host computer is general purpose computer. Attached processor is back end machine driven by the host computer.

2) SIMD array processor - This is computer with multiple processing unit operating in parallel both type of array processors, manipulate vectors but their internal organization is diffn.

SIMD is a computer with multiple processing units operating in parallel. The processing units are synchronized to perform the same operation under the control of a common control unit. Thus providing a single instruction stream, multiple data stream [SIMD] organization.



Q.1) Explain RISC pipeline in COT.

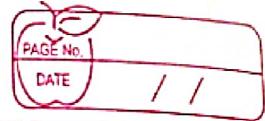
Ans. RISC stands for Reduced Instruction set computers. It was introduced to execute as fast as one instruction per clock cycle. This RISC pipeline helps to simplify the computer architecture design.

It relates to what is known as the semantic gaps, this is the difference & between the operations provided in the high level languages & those provided in computer architecture. The main benefit of RISC to implement instructions at the cost of one per clock cycle is continually not applicable because each instruction cannot be fetched from memory & implemented in one clock cycle correctly under all circumstances.

Principles of RISC pipeline -

- Keep the most frequently accepted operand in CPU register.
- It can minimize the register to memory operation.
- It can use a simplified instruction set & leave out those complex & unnecessary instructions.
- It can optimize the design of instruction pipeline such that minimum compiler code generation can be achieved.
- It can use a high number of registers to enhance operand referring & decrease the processor memory traffic.

Teacher's Signature:.....



A frequent collection of instructions for a RISC processor, is of 3 types are as follows-

- Data manipulation Instruction- Manage the data in processor register.
- Data transfer Instruction- These are load & store instruction that use an effective address.
- Program control Instruction- It use register values & a constant to evaluate the branch address.