

Module 1

	Part A
1.	<p>What is the function of program counter in 8085 microprocessor ?</p> <p>PC is a 16-bit register. It contains a memory address. PC contains that very memory address from where the next instruction is to be fetched for execution. Suppose the PC contents are 8000H, then it means that the 8085 Desires to fetch the instruction Byte at 8000H. After fetching the Byte at 8000H, the PC is automatically incremented by 1.</p>
2.	<p>What is trap interrupt and its significance?</p> <p>It is non maskable edge and level triggered interrupt. TRAP has the highest priority and vectored interrupt. Edge and level triggered means that the TRAP must go high and remain high until it is acknowledged. In case of sudden power failure, it executes a ISR and send the data from main memory to backup memory.</p>
3.	<p>Define microprocessor.</p> <p>A microprocessor is a component that performs the instructions and tasks involved in computer processing. In a computer system, the microprocessor is the central unit that executes and manages the logical instructions passed to it.</p> <p>A microprocessor may also be called a processor or central processing unit, but it is actually more advanced in terms of architectural design and is built over a silicon microchip.</p>
4.	<p>Differentiate between maskable and non maskable interrupts.</p> <p>An interrupt is an event caused by a component other than the CPU. It indicates the CPU of an external event that requires immediate attention. Interrupts occur asynchronously. Maskable and non-maskable interrupts are two types of interrupts.</p> <p>1. Maskable Interrupt :</p> <p>An Interrupt that can be disabled or ignored by the instructions of CPU are called as Maskable Interrupt. The interrupts are either edge-triggered or level-triggered or level-triggered.</p> <p>Eg:</p> <p>RST6.5,RST7.5,RST5.5 of 8085</p> <p>2. Non-Maskable Interrupt :</p> <p>An interrupt that cannot be disabled or ignored by the instructions of CPU are called as Non-Maskable Interrupt. A Non-maskable interrupt is often used when response time is critical or when an interrupt should never be disable during normal system operation. Such uses include reporting non-recoverable hardware errors, system debugging and profiling and handling of species cases like system resets.</p> <p>Eg:Trap of 8085</p>

	<table border="1"> <thead> <tr> <th>SR.NO.</th><th>Maskable Interrupt</th><th>Non Maskable Interrupt</th></tr> </thead> <tbody> <tr> <td>1</td><td>Maskable interrupt is a hardware interrupt that can be disabled or ignored by the instructions of CPU.</td><td>A non-maskable interrupt is a hardware interrupt that cannot be disabled or ignored by the instructions of CPU.</td></tr> <tr> <td>2</td><td>When maskable interrupt occur, it can be handled after executing the current instruction.</td><td>When non-maskable interrupts occur, the current instructions and status are stored in stack for the CPU to handle the interrupt.</td></tr> <tr> <td>3</td><td>Maskable interrupts help to handle lower priority tasks.</td><td>Non-maskable interrupt help to handle higher priority tasks such as watchdog timer.</td></tr> <tr> <td>4</td><td>Maskable interrupts used to interface with peripheral device.</td><td>Non maskable interrupt used for emergency purpose e.g power failure, smoke detector etc .</td></tr> <tr> <td>5</td><td>In maskable interrupts, response time is high.</td><td>In non maskable interrupts, response time is low.</td></tr> <tr> <td>6</td><td>It may be vectored or non-vectored.</td><td>All are vectored interrupts.</td></tr> <tr> <td>7</td><td>Operation can be masked or made pending.</td><td>Operation Cannot be masked or made pending.</td></tr> <tr> <td>8</td><td>RST6.5, RST7.5, and RST5.5 of 8085 are some common examples of maskable Interrupts.</td><td>Trap of 8085 microprocessor is an example for non-maskable interrupt.</td></tr> </tbody> </table>	SR.NO.	Maskable Interrupt	Non Maskable Interrupt	1	Maskable interrupt is a hardware interrupt that can be disabled or ignored by the instructions of CPU.	A non-maskable interrupt is a hardware interrupt that cannot be disabled or ignored by the instructions of CPU.	2	When maskable interrupt occur, it can be handled after executing the current instruction.	When non-maskable interrupts occur, the current instructions and status are stored in stack for the CPU to handle the interrupt.	3	Maskable interrupts help to handle lower priority tasks.	Non-maskable interrupt help to handle higher priority tasks such as watchdog timer.	4	Maskable interrupts used to interface with peripheral device.	Non maskable interrupt used for emergency purpose e.g power failure, smoke detector etc .	5	In maskable interrupts, response time is high.	In non maskable interrupts, response time is low.	6	It may be vectored or non-vectored.	All are vectored interrupts.	7	Operation can be masked or made pending.	Operation Cannot be masked or made pending.	8	RST6.5, RST7.5, and RST5.5 of 8085 are some common examples of maskable Interrupts.	Trap of 8085 microprocessor is an example for non-maskable interrupt.
SR.NO.	Maskable Interrupt	Non Maskable Interrupt																										
1	Maskable interrupt is a hardware interrupt that can be disabled or ignored by the instructions of CPU.	A non-maskable interrupt is a hardware interrupt that cannot be disabled or ignored by the instructions of CPU.																										
2	When maskable interrupt occur, it can be handled after executing the current instruction.	When non-maskable interrupts occur, the current instructions and status are stored in stack for the CPU to handle the interrupt.																										
3	Maskable interrupts help to handle lower priority tasks.	Non-maskable interrupt help to handle higher priority tasks such as watchdog timer.																										
4	Maskable interrupts used to interface with peripheral device.	Non maskable interrupt used for emergency purpose e.g power failure, smoke detector etc .																										
5	In maskable interrupts, response time is high.	In non maskable interrupts, response time is low.																										
6	It may be vectored or non-vectored.	All are vectored interrupts.																										
7	Operation can be masked or made pending.	Operation Cannot be masked or made pending.																										
8	RST6.5, RST7.5, and RST5.5 of 8085 are some common examples of maskable Interrupts.	Trap of 8085 microprocessor is an example for non-maskable interrupt.																										
5.	<p>What are interrupts of Intel 8085?</p> <p>Interrupt is a mechanism by which an I/O or an instruction can suspend the normal execution of processor and get itself serviced. Generally, a particular task is assigned to that interrupt signal. In the microprocessor based system the interrupts are used for data transfer between the peripheral devices and the microprocessor.</p> <p>An interrupt is an external asynchronous input that informs the microprocessor to</p>																											

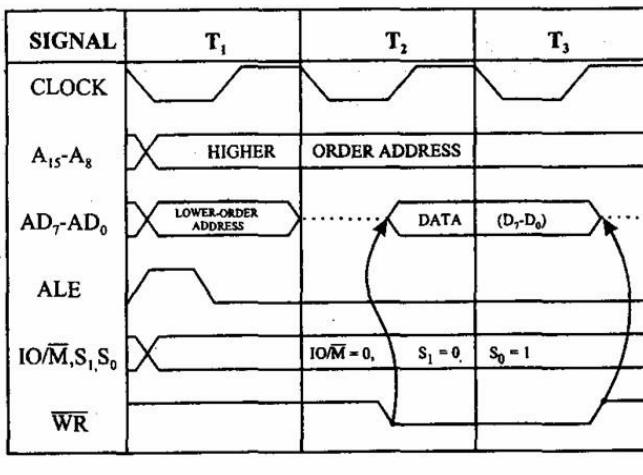
	complete the instruction that is currently executing and fetch a new routine in order to offer service to the I/O device. Once the I/O device is serviced, the microprocessor will continue with the execution of its normal program.
6.	<p>What are the features of 8085 microprocessor?</p> <ul style="list-style-type: none"> <input type="checkbox"/> It is invented in 1976. <input type="checkbox"/> It is an 8-bit microprocessor. <input type="checkbox"/> 8085 microprocessor provides 16 address lines, therefore it can access $2^{16} = 64K$ bytes of memory. <input type="checkbox"/> It has 8 Data lines and 16 address lines <input type="checkbox"/> It generates 8 bit I/O address, hence it can access $2^8 = 256$ input ports and 256 output ports. <input type="checkbox"/> 8085 microprocessor provides one Accumulator, one Flag register, 6 General Purpose Registers (B, C, D, E, H and L) and two special purpose registers (PC, SP). <input type="checkbox"/> The maximum clock frequency of 8085 microprocessor is 3MHz where as minimum clock frequency is 500 KHz. <input type="checkbox"/> 8085 microprocessor provides on chip clock generator, therefore there is no need of external clock generator, but it requires external tuned circuit like LC, RC or crystal. <input type="checkbox"/> It is available in 40 pin dual in line (DIP) package. <input type="checkbox"/> It requires a +5volts of power supply. <input type="checkbox"/> 8085 microprocessor has five hardware interrupts: TRAP, RST 5.5, RST 6.5, RST 7.5, and INTR. The hardware interrupt capability of 8085 microprocessor can be increased by providing external hardware. <input type="checkbox"/> 8085 microprocessor has capability to share its bus with external bus controller (Direct Memory Access controller); for transferring large amount of data from memory to I/O and vice versa. <input type="checkbox"/> 8085 microprocessor provides two serial I/O lines which are SOD and SID; it means, serial peripherals can be interfaced with 8085 microprocessor directly.
7.	<p>What is the function of timing and control unit?</p> <ul style="list-style-type: none"> • It provides timing and control signal to the microprocessor to perform the various operation.

- | | |
|--|---|
| | <ul style="list-style-type: none"> • It has three control signal. • It controls all external and internal circuits. |
|--|---|

We use Timing and Controlling unit in 8085 for the generation of timing signals and the signals to control. All the operations and functions both interior and exterior of a microprocessor are controlled by this unit.

	Part B
--	--------

- | | |
|----|---|
| 8. | Draw and explain the timing diagram of memory write cycle with example. |
|----|---|



These machine cycles have 3 T-states.

T1 state:

- The higher order address bus (A8-A15) and lower order address and data multiplexed (AD0-AD7) bus.
- ALE goes high so that the memory latches the (AD0-AD7) so that complete 16-bit address are available.
- The mp identifies the memory read machine cycle from the status signals IO/M'=0, S1=0, S0=1. This condition indicates the memory read cycle.

T2 state:

- Selected memory location is placed on the (D0-D7) of the A/D multiplexed bus. WR' goes LOW

T3 State:

- In the middle of the T3 state WR' goes high and disables the memory write operation. The data which was obtained from the memory is then decoded.

- | | |
|----|---|
| 9. | Explain the sequence of events during the execution of the CALL instruction by 8085 processor with the help of neat timing diagram. |
|----|---|

Time required to execute and fetch an entire instruction is called instruction cycle.

It consists:

Fetch cycle – The next instruction is fetched by the address stored in program counter (PC) and then stored in the instruction register.

Decode instruction – Decoder interprets the encoded instruction from instruction register.

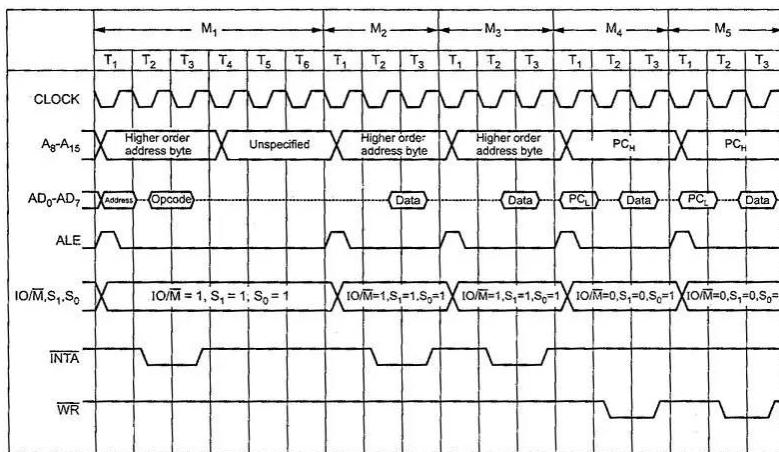
Reading effective address – The address given in instruction is read from main memory and required data is fetched. The effective address depends on direct addressing mode or indirect addressing mode.

Execution cycle – consists memory read (MR), memory write (MW), input output read (IOR) and input output write (IOW)

The time required by the microprocessor to complete an operation of accessing memory or input/output devices is called machine cycle. One time period of frequency of microprocessor is called t-state. A t-state is measured from the falling edge of one clock pulse to the falling edge of the next clock pulse.

Fetch cycle takes four t-states and execution cycle takes three t-states.

Timing diagram of CALL instruction



- The instruction cycle of CALL instruction consists of five machine cycles.
- The first three are the Interrupt Acknowledge Machine Cycle explained above. At the end of the IAMC, the instruction is decoded.
- In the second and third machine cycles, the device which caused the interruption gives the address of the location where the program location is supposed to jump after getting the interrupt signal.

	<ul style="list-style-type: none"> The fourth and fifth machine cycles are MWMC. During these machine cycles, the microprocessor saves the contents of the program counter into stack since it will be executing the interrupt service routine and will have to return to that location again. 										
10.	<p>Write short note on vectored interrupts of 8085 microprocessor.</p> <p><i>Vectored Interrupts</i> are those which have fixed vector address (starting address of sub-routine) and after executing these, program control is transferred to that address.</p> <p>Vector Addresses are calculated by the formula $8 * \text{TYPE}$</p> <table border="1"> <thead> <tr> <th>INTERRUPT</th> <th>VECTOR ADDRESS</th> </tr> </thead> <tbody> <tr> <td>TRAP (RST 4.5)</td> <td>24 H</td> </tr> <tr> <td>RST 5.5</td> <td>2C H</td> </tr> <tr> <td>RST 6.5</td> <td>34 H</td> </tr> <tr> <td>RST 7.5</td> <td>3C H</td> </tr> </tbody> </table>	INTERRUPT	VECTOR ADDRESS	TRAP (RST 4.5)	24 H	RST 5.5	2C H	RST 6.5	34 H	RST 7.5	3C H
INTERRUPT	VECTOR ADDRESS										
TRAP (RST 4.5)	24 H										
RST 5.5	2C H										
RST 6.5	34 H										
RST 7.5	3C H										
11.	<p>What is meant by an interrupt? Explain with examples.</p> <p>Interrupt is a mechanism by which an I/O or an instruction can suspend the normal execution of processor and get itself serviced. Generally, a particular task is assigned to that interrupt signal. In the microprocessor based system the interrupts are used for data transfer between the peripheral devices and the microprocessor.</p> <p>An interrupt is a signal emitted by a device attached to a computer or from a program within the computer. It requires the operating system (OS) to stop and figure out what to do next. An interrupt temporarily stops or terminates a service or a current process. Most I/O devices have a bus control line called Interrupt Service Routine (ISR) for this purpose.</p> <p>An interrupt signal might be planned (i.e., specifically requested by a program) or it may be unplanned (i.e., caused by an event that may not be related to a program that's currently running on the system).</p> <p>Maskable/Non-Maskable Interrupt</p> <p>An interrupt that can be disabled by writing some instruction is known as Maskable Interrupt otherwise it is called Non-Maskable Interrupt.</p>										

There are 6 pins available in 8085 for interrupt

1. TRAP
2. RST 7.5
3. RST6.5
4. RST5.5
5. INTR
6. INTA

1. Software Interrupts

A software interrupt is a particular instruction that can be inserted into the desired location in the program. There are eight Software interrupts in 8085 Microprocessor. From RST0 to RST7.

1. RST0
2. RST1
3. RST2
4. RST3
5. RST4
6. RST5
7. RST6
8. RST7

They allow the microprocessor to transfer program control from the main program to the subroutine program. After completing the subroutine program, the program control returns back to the main program.

2. Hardware Interrupt

As already discussed that there are 6 interrupt pins in the microprocessor used as Hardware Interrupts given below:

1. TRAP
2. RST7.5
3. RST6.5
4. RST5.5
5. INTR

TRAP

- It is non maskable edge and level triggered interrupt.
- TRAP has the highest priority and vectored interrupt.
- Edge and level triggered means that the TRAP must go high and remain high until

it is acknowledged.

- In case of sudden power failure, it executes a ISR and send the data from main memory to backup memory.
- As we know that TRAP can not be masked but it can be delayed using HOLD signal. This interrupt transfers the microprocessor's control to location 0024H.

RST7.5

- It has the second highest priority. It is maskable and edge level triggered interrupt. The vector address of this interrupt is 003CH. Edge sensitive means input goes high and no need to maintain high state until it is recognized.

RST6.5 and RST5.5

These are level triggered and maskable interrupts. When RST6.5 pin is at logic 1, INTE flip-flop is set. RST 6.5 has third highest priority and RST 5.5 has fourth highest priority.

INTR

It is level triggered and maskable interrupt. The following sequence of events occurs when INTR signal goes high:

1. The 8085 checks the status of INTR signal during execution of each instruction.
2. If INTR signal is high, then 8085 complete its current instruction and sends active low interrupt acknowledge signal, if the interrupt is enabled.
3. On receiving the instruction, the 8085 save the address of next instruction on stack and execute received instruction.

3. Vectored and Non-Vectored Interrupts –

Vectored Interrupts are those which have fixed vector address (starting address of subroutine) and after executing these, program control is transferred to that address.

INTERRUPT	VECTOR ADDRESS
TRAP (RST 4.5)	24 H
RST 5.5	2C H
RST 6.5	34 H

	RST 7.5	3C H
	<p><u>Non-Vectored Interrupts</u> are those in which vector address is not predefined. The interrupting device gives the address of sub-routine for these interrupts. <i>INTR</i> is the only non-vectored interrupt in 8085 microprocessor.</p> <p><u>4. Maskable and Non-Maskable Interrupts –</u></p> <p>Maskable Interrupts are those which can be disabled or ignored by the microprocessor. These interrupts are either edge-triggered or level-triggered, so they can be disabled. <i>INTR</i>, <i>RST 7.5</i>, <i>RST 6.5</i>, <i>RST 5.5</i> are maskable interrupts in 8085 microprocessor.</p> <p><u>Non-Maskable</u> Interrupts are those which cannot be disabled or ignored by microprocessor. <i>TRAP</i> is a non-maskable interrupt. It consists of both level as well as edge triggering and is used in critical power failure conditions.</p>	
12.	<p>Briefly explain the instruction set of Intel 8085</p> <h2 style="text-align: center;">Instruction Set of 8085</h2> <ul style="list-style-type: none"> • An instruction is a binary pattern designed inside a microprocessor to perform a specific function. • The entire group of instructions that a microprocessor supports is called Instruction Set. • 8085 has 246 instructions. • Each instruction is represented by an 8-bit binary value. • These 8-bits of binary value is called Op-Code or Instruction Byte. <h2 style="text-align: center;">Classification of Instruction Set</h2> <ul style="list-style-type: none"> • Data Transfer Instruction • Arithmetic Instructions • Logical Instructions • Branching Instructions • Control Instructions 	

Data Transfer Instructions

- These instructions move data between registers, or between memory and registers.
- These instructions copy data from source to destination.
- While copying, the contents of source are not modified.

Data Transfer Instructions

Opcode	Operand	Description
MOV	Rd, Rs M, Rs Rd, M	Copy from source to destination.

- This instruction copies the contents of the source register into the destination register.
- The contents of the source register are not altered.
- If one of the operands is a memory location, its location is specified by the contents of the HL registers.
- **Example:** MOV B, C or MOV B, M

Data Transfer Instructions

Opcode	Operand	Description
MVI	Rd, Data M, Data	Move immediate 8-bit

- The 8-bit data is stored in the destination register or memory.
- If the operand is a memory location, its location is specified by the contents of the H-L registers.
- **Example:** MVI B, 57H or MVI M, 57H

Clip slide

BEFORE EXECUTION

A	F
B	C
D	E
H	L

AFTER EXECUTION

A	F
B	60
D	E
H	L

MVI B,60H

BEFORE EXECUTION

204FH
HL=2050H
2051H

AFTER EXECUTION

204FH
HL=2050H
2051H

MVI M,40H

3

Data Transfer Instructions

Opcode	Operand	Description
LDA	16-bit address	Load Accumulator

- The contents of a memory location, specified by a 16-bit address in the operand, are copied to the accumulator.
- The contents of the source are not altered.
- Example:** LDA 2034H

BEFORE EXECUTION

A	
2000H	30

AFTER EXECUTION

A	30
2000H	30

LDA 2000H

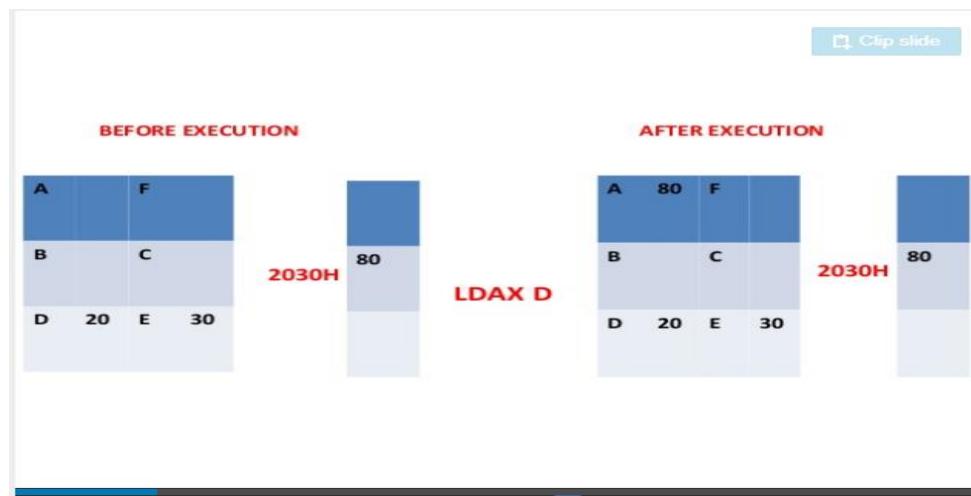
In 8085 Instruction set, **LDAX** is a mnemonic that stands for LoAD Accumulator from memory pointed by eXtended register pair denoted as “rp” in the instruction. This instruction uses register indirect addressing for specifying the data. It occupies only 1-Byte in the memory.

4

Data Transfer Instructions

Opcode	Operand	Description
LDAX	B/D Register Pair	Load accumulator indirect

- The contents of the designated register pair point to a memory location.
- This instruction copies the contents of that memory location into the accumulator.
- The contents of either the register pair or the memory location are not altered.
- Example:** LDAX B



5

Data Transfer Instructions

Opcode	Operand	Description
LXI	Reg. pair, 16-bit data	Load register pair immediate

- This instruction loads 16-bit data in the register pair.
- **Example:** LXI H, 2034 H

6

Data Transfer Instructions

Opcode	Operand	Description
LHLD	16-bit address	Load H-L registers direct

- This instruction copies the contents of memory location pointed out by 16-bit address into register L.
- It copies the contents of next memory location into register H.
- **Example:** LHLD 2040 H

1

Arithmetic Instructions

Opcode	Operand	Description
ADD	R M	Add register or memory to accumulator

- The contents of register or memory are added to the contents of accumulator.
- The result is stored in accumulator.
- If the operand is memory location, its address is specified by H-L pair.
- All flags are modified to reflect the result of the addition.
- **Example:** ADD B or ADD M

BEFORE EXECUTION

A	20
B	C
D	E
H	L

ADD C
A=A+R

AFTER EXECUTION Clip slide

A	50
B	C
D	E
H	L

BEFORE EXECUTION

A	20
B	C
D	E
H	L

ADD M
A=A+M

10

2050

AFTER EXECUTION

A	30
B	C
D	E
H	L

10

2050

Addition

- Any 8-bit number, or the contents of register, or the contents of memory location can be added to the contents of accumulator.
- The result (sum) is stored in the accumulator.
- No two other 8-bit registers can be added directly.
- Example:** The contents of register B cannot be added directly to the contents of register C.

Subtraction

- Any 8-bit number, or the contents of register, or the contents of memory location can be subtracted from the contents of accumulator.
- The result is stored in the accumulator.
- Subtraction is performed in 2's complement form.
- If the result is negative, it is stored in 2's complement form.
- No two other 8-bit registers can be subtracted directly.

Increment / Decrement

- The 8-bit contents of a register or a memory location can be incremented or decremented by 1.
- The 16-bit contents of a register pair can be incremented or decremented by 1.
- Increment or decrement can be performed on any register or a memory location.

2

Arithmetic Instructions

Opcode	Operand	Description
ADC	R M	Add register or memory to accumulator with carry

- The contents of register or memory and Carry Flag (CY) are added to the contents of accumulator.
- The result is stored in accumulator.
- If the operand is memory location, its address is specified by H-L pair.
- All flags are modified to reflect the result of the addition.
- **Example:** ADC B or ADC M

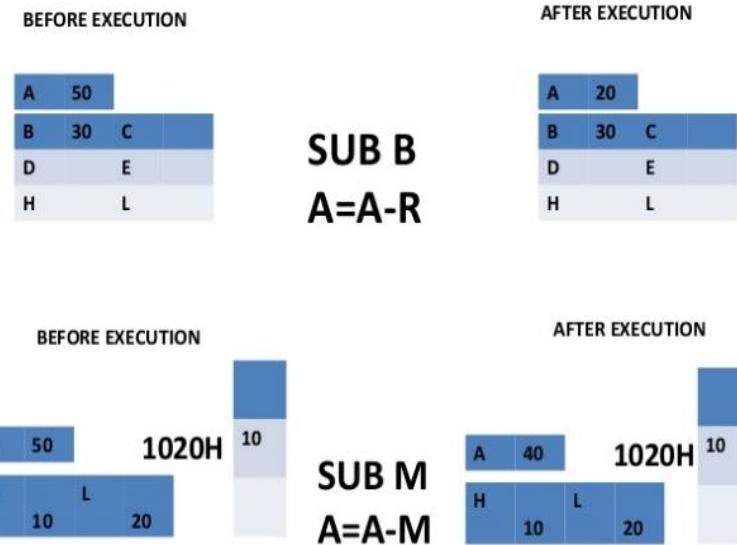
	BEFORE EXECUTION	AFTER EXECUTION																																	
	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>CY</td><td>1</td></tr> <tr><td>A</td><td>50</td></tr> <tr><td>B</td><td></td><td>C</td><td>20</td></tr> <tr><td>D</td><td></td><td>E</td><td></td></tr> <tr><td>H</td><td></td><td>L</td><td></td></tr> </table>	CY	1	A	50	B		C	20	D		E		H		L		<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>CY</td><td>0</td></tr> <tr><td>A</td><td>71</td></tr> <tr><td>B</td><td></td><td>C</td><td>20</td></tr> <tr><td>D</td><td></td><td>E</td><td></td></tr> <tr><td>H</td><td></td><td>L</td><td></td></tr> </table>	CY	0	A	71	B		C	20	D		E		H		L		
CY	1																																		
A	50																																		
B		C	20																																
D		E																																	
H		L																																	
CY	0																																		
A	71																																		
B		C	20																																
D		E																																	
H		L																																	
		ADC C A=A+R+CY																																	
	BEFORE EXECUTION	AFTER EXECUTION																																	
	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>CY</td><td>1</td></tr> <tr><td>A</td><td>20</td></tr> <tr><td>2050H</td><td>30</td></tr> <tr><td>H</td><td>20</td><td>L</td><td>50</td></tr> </table>	CY	1	A	20	2050H	30	H	20	L	50	<table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>CY</td><td>0</td></tr> <tr><td>A</td><td>51</td></tr> <tr><td>2050H</td><td>30</td></tr> <tr><td>H</td><td>20</td><td>L</td><td>50</td></tr> </table>	CY	0	A	51	2050H	30	H	20	L	50													
CY	1																																		
A	20																																		
2050H	30																																		
H	20	L	50																																
CY	0																																		
A	51																																		
2050H	30																																		
H	20	L	50																																
		ADC M A=A+M+CY																																	

3

Arithmetic Instructions

Opcode	Operand	Description
SUB	R M	Subtract register or memory from accumulator

- The contents of the register or memory location are subtracted from the contents of the accumulator.
- The result is stored in accumulator.
- If the operand is memory location, its address is specified by H-L pair.
- All flags are modified to reflect the result of subtraction.
- Example:** SUB B or SUB M

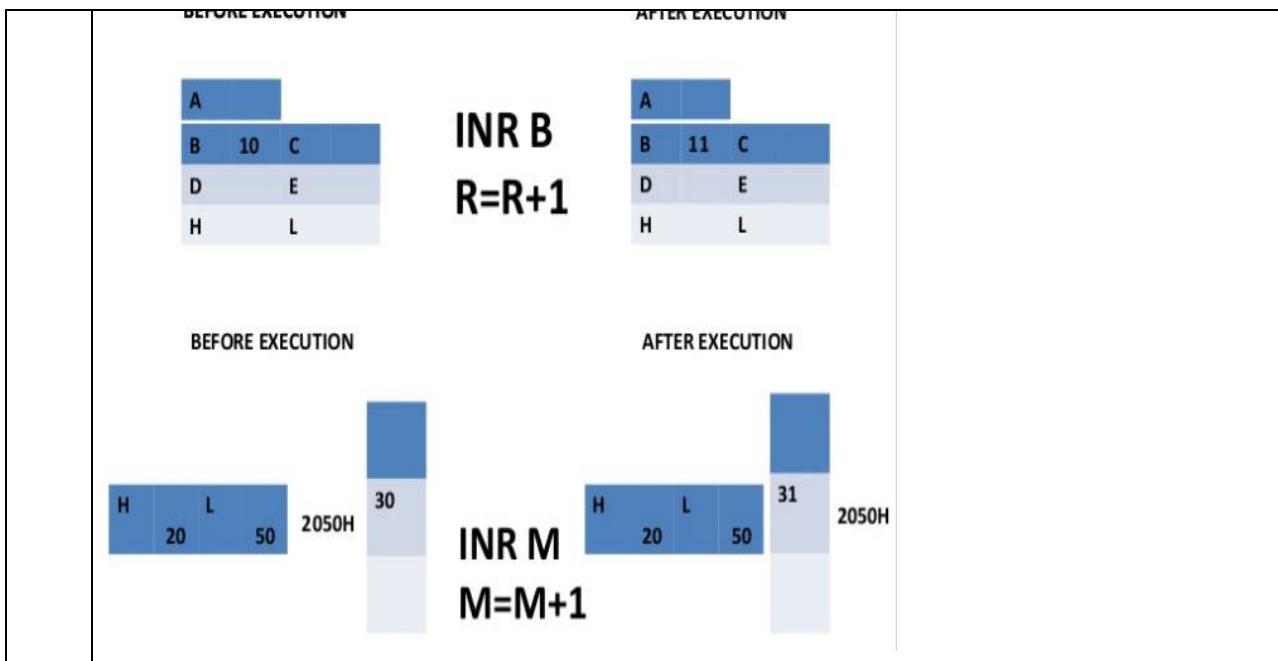


4

Arithmetic Instructions

Opcode	Operand	Description
INR	R M	Increment register or memory by 1

- The contents of register or memory location are incremented by 1.
- The result is stored in the same place.
- If the operand is a memory location, its address is specified by the contents of H-L pair.
- Example:** INR B or INR M



5

Arithmetic Instructions

Opcode	Operand	Description
DCR	R M	Decrement register or memory by 1

- The contents of register or memory location are decremented by 1.
- The result is stored in the same place.
- If the operand is a memory location, its address is specified by the contents of H-L pair.
- **Example:** DCR B or DCR M

Logical Instructions

- These instructions perform logical operations on data stored in registers, memory and status flags.
- The logical operations are:
 - AND
 - OR
 - XOR
 - Rotate
 - Compare
 - Complement

AND, OR, XOR

- Any 8-bit data, or the contents of register, or memory location can logically have
 - AND operation
 - OR operation
 - XOR operation
- with the contents of accumulator.
- The result is stored in accumulator.

Rotate

- Each bit in the accumulator can be shifted either left or right to the next position.

1

Compare

- Any 8-bit data, or the contents of register, or memory location can be compared for:
 - Equality
 - Greater Than
 - Less Than
- with the contents of accumulator.
- The result is reflected in status flags.

Complement

- The contents of accumulator can be complemented.
- Each 0 is replaced by 1 and each 1 is replaced by 0.

Logical Instructions

Opcode	Operand	Description
CMP	R M	Compare register or memory with accumulator

- The contents of the operand (register or memory) are compared with the contents of the accumulator.
- Both contents are preserved .
- The result of the comparison is shown by setting the flags of the PSW as follows:

Logical Instructions

Opcode	Operand	Description
CMP	R M	Compare register or memory with accumulator

- if $(A) < (\text{reg}/\text{mem})$: carry flag is set
- if $(A) = (\text{reg}/\text{mem})$: zero flag is set
- if $(A) > (\text{reg}/\text{mem})$: carry and zero flags are reset.
- **Example:** CMP B or CMP M

2

Logical Instructions

Opcode	Operand	Description
CPI	8-bit data	Compare immediate with accumulator

- The 8-bit data is compared with the contents of accumulator.
- The values being compared remain unchanged.
- The result of the comparison is shown by setting the flags of the PSW as follows:

3

Logical Instructions

Opcode	Operand	Description
CPI	8-bit data	Compare immediate with accumulator

- if $(A) < \text{data}$: carry flag is set
- if $(A) = \text{data}$: zero flag is set
- if $(A) > \text{data}$: carry and zero flags are reset
- **Example:** CPI 89H

Logical Instructions

Opcode	Operand	Description
ORA	R M	Logical OR register or memory with accumulator

- The contents of the accumulator are logically ORed with the contents of the register or memory.
- The result is placed in the accumulator.
- If the operand is a memory location, its address is specified by the contents of H-L pair.
- S, Z, P are modified to reflect the result.
- CY and  are reset.
- **Example:** ORA B or ORA M.

Opcode	Operand	Description
RLC	None	Rotate accumulator left

- Each binary bit of the accumulator is rotated left by one position.
- Bit D₇ is placed in the position of D₀ as well as in the Carry flag.
- CY is modified according to bit D₇.
- S, Z, P, AC are not affected.
- **Example:** RLC.

		<p>The diagram illustrates the RRC (Rotate Right Circular) instruction. It shows a 9-bit register with bits B7 to B0. Bit B7 is labeled CY. Four blue arrows above the register indicate a rightward shift of one position for each bit. A green arrow points from the rightmost bit B0 back to the leftmost bit B7, representing the carry of bit B0 into bit B7. Below the register, a 9-bit binary number is shown with its bits labeled B7 through B0.</p>						
6	<table border="1"> <thead> <tr> <th>Opcode</th> <th>Operand</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>RRC</td> <td>None</td> <td>Rotate accumulator right</td> </tr> </tbody> </table>	Opcode	Operand	Description	RRC	None	Rotate accumulator right	
Opcode	Operand	Description						
RRC	None	Rotate accumulator right						
<ul style="list-style-type: none"> • Each binary bit of the accumulator is rotated right by one position. • Bit Do is placed in the position of D7 as well as in the Carry flag. • CY is modified according to bit Do. • S, Z, P, AC are not affected. • Example: RRC. 								
<h2>Branching Instructions</h2> <ul style="list-style-type: none"> • The branching instruction alter the normal sequential flow. • These instructions alter either unconditionally or conditionally. 								

1

Branching Instructions

Opcode	Operand	Description
JMP	16-bit address	Jump unconditionally

- The program sequence is transferred to the memory location specified by the 16-bit address given in the operand.
- Example:** JMP 2034 H.

Conditional Jumps

Instruction Code	Description	Condition For Jump
JC	Jump on carry	CY=1
JNC	Jump on not carry	CY=0
JP	Jump on positive	S=0
JM	Jump on minus	S=1
JPE	Jump on parity even	P=1
JPO	Jump on parity odd	P=0
JZ	Jump on zero	Z=1
JNZ	Jump on not zero	Z=0

2

Branching Instructions

Opcode	Operand	Description
Jx	16-bit address	Jump conditionally

- The program sequence is transferred to the memory location specified by the 16-bit address given in the operand based on the specified flag of the PSW.
- Example:** JZ 2034 H.

Jump Conditionally

Opcde	Description	Status Flags
JC	Jump if Carry	CY = 1
JNC	Jump if No Carry	CY = 0
JP	Jump if Positive	S = 0
JM	Jump if Minus	S = 1
JZ	Jump if Zero	Z = 1
JNZ	Jump if No Zero	Z = 0
JPE	Jump if Parity Even	P = 1
JPO	Jump if Parity Odd	P = 0

3

Branching Instructions

Opcde	Operand	Description
CALL	16-bit address	Call unconditionally

- The program sequence is transferred to the memory location specified by the 16-bit address given in the operand.
- Before the transfer, the address of the next instruction after CALL (the contents of the program counter) is pushed onto the stack.
- Example:** CALL 2034 H.

4

Branching Instructions

Opcde	Operand	Description
RET	None	Return unconditionally

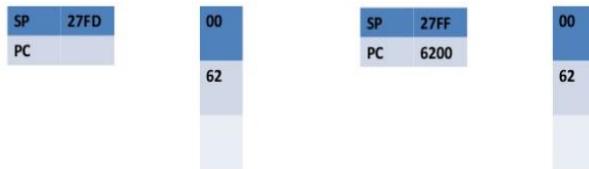
- The program sequence is transferred from the subroutine to the calling program.
- The two bytes from the top of the stack are copied into the program counter, and program execution begins at the new address.
- Example:** RET.

1

Control Instructions

Opcode	Operand	Description
NOP	None	No operation

- No operation is performed.
- The instruction is fetched and decoded but no operation is executed.
- **Example:** NOP



2

Control Instructions

Opcode	Operand	Description
HLT	None	Halt

- The CPU finishes executing the current instruction and halts any further execution.
- An interrupt or reset is necessary to exit from the halt state.
- **Example:** HLT

3

Control Instructions

Opcode	Operand	Description
DI	None	Disable interrupt

- The interrupt enable flip-flop is reset and all the interrupts except the TRAP are disabled.
- No flags are affected.
- **Example:** DI

4

Control Instructions

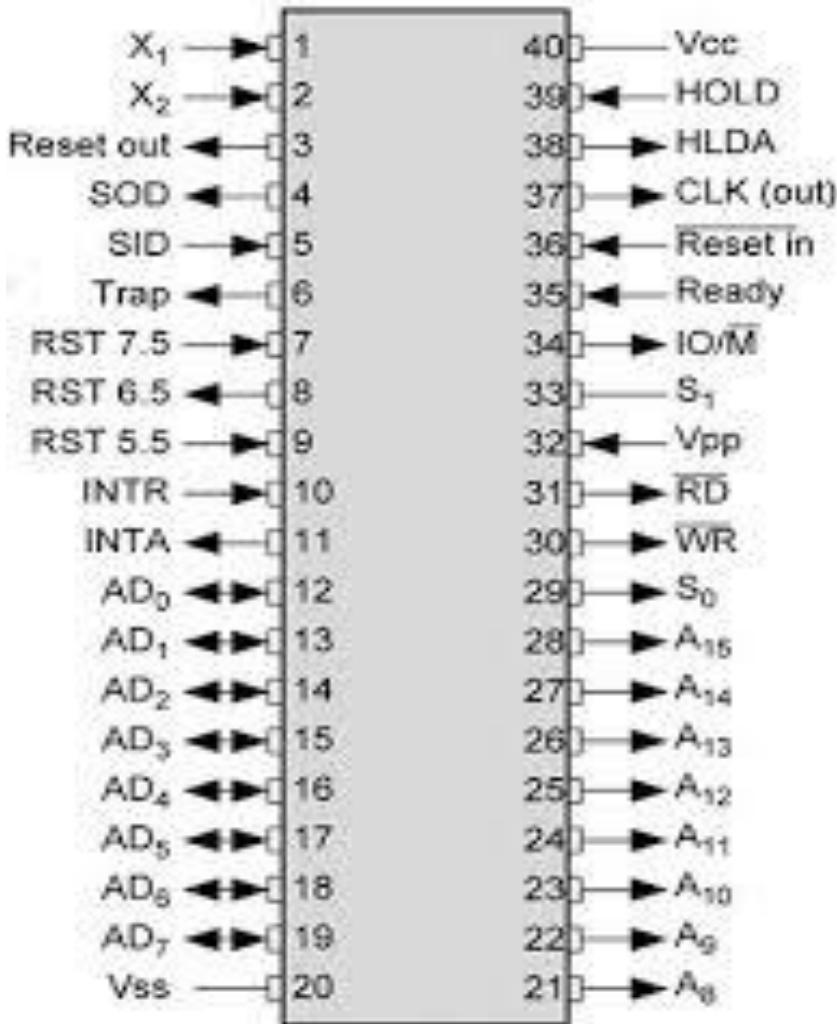
Opcode	Operand	Description
EI	None	Enable interrupt

- The interrupt enable flip-flop is set and all interrupts are enabled.
- No flags are affected.
- This instruction is necessary to re-enable the interrupts (except TRAP).
- **Example:** EI

Part C

13. With the help of pin diagram explain the operations of 8085 microprocessor.

PIN DIAGRAM OF 8086 MICROPROCESSOR



1.Address Bus

- These pins carry the higher order of address bus.
- The address is sent from microprocessor to memory.
- **A8 – A15.** It carries the most significant 8-bit of memory I/O address.

2.Data Bus

- Data bus is of 8 Bit.
- It is used to transfer data between microprocessor and memory.
- **AD0 – AD7.** It carries the least significant 8-bit address and data bus.

3.Control Signals

- **ALE(Address Latch Enable)** –Microprocessor is handling information and data. ALE selects the address or data to be transferred. If this pin is high, the bits on AD_7 - AD_0 are address bus. If this pin is low, the bits on AD_7 - AD_0 are data bits
- **RD(READ):** When microprocessor reads data or codes from a memory location or

an input device , it is called READ operation.RD is a signal sent by the microprocessor to the memory /input device to control READ operation. Low signal in this pin shows the read operation either from I/O devices or from the memory unit. This signal indicates that the selected IO or memory device is to be read and is ready for accepting data available on the data bus.

- **WR(WRITE):** When microprocessor sends data to a memory location or an output device ,it is called WRITE operation. WR is a signal sent by the microprocessor to the memory /output device to control WRITE operation. Low signal in this pin represents the write operation at the memory or I/O devices. This signal indicates that the data on the data bus is to be written into a selected memory or IO location.

4./Status Signal

- **IO/M:**This signal is used to differentiate between IO and Memory operations, i.e. when it is high indicates IO operation and when it is low then it indicates memory operation.
- **S0 & S1 :**These signals are used to identify the type of current operation. It indicate the type of machine cycle in progress.

S0	S1	Operations
0	0	HALT
0	1	WRITE
1	0	READ
1	1	FETCH

- **Ready:**This signal indicates that the device is ready to send or receive data. If READY is low, then the CPU has to wait for READY to go high.

1. Power Supply

- There are 2 power Supply signals:
- **VCC :**VCC indicates +5v power supply.
- **VSS :**VSS indicates ground signal.

2. Interrupt Signals

- **TRAP:**TRAP is usually used for power failure and emergency shutoff.
- **RST 7.5** It is a maskable interrupt.
It has the second highest priority.

- **RST 6.5** It is a maskable interrupt. It has the third highest priority.
- **RST 5.5** It is a maskable interrupt. It has the fourth highest priority.
- **INTR** It is a general purpose interrupt. It is a maskable interrupt. It has the lowest priority

3. Externally Initiated Signals

- **INTA** It is an interrupt acknowledgment signal.
- **RESET IN** This signal is used to reset the microprocessor by setting the program counter to zero.
- **RESET OUT** This signal is used to reset all the connected devices when the microprocessor is reset.
- **HOLD** This signal indicates that another master is requesting the use of the address and data buses.
- **HLDA** It indicates that the CPU has received the HOLD request and it will relinquish the bus in the next clock

4. Serial I/O Signals SOD (Serial Output Data line)

- **SID (Serial Input Data line)**-with this pin ,data is serially fedto the processor directly through the input devices.
- The data on this line is loaded into accumulator whenever a RIM instruction is executed.
- **SOD(Serial Output Data)** – Once the data is processed in the microprocessor then this pin represents bit by bit results at the output devices.
- The output SOD is set/reset as specified by the SIM instruction.
-

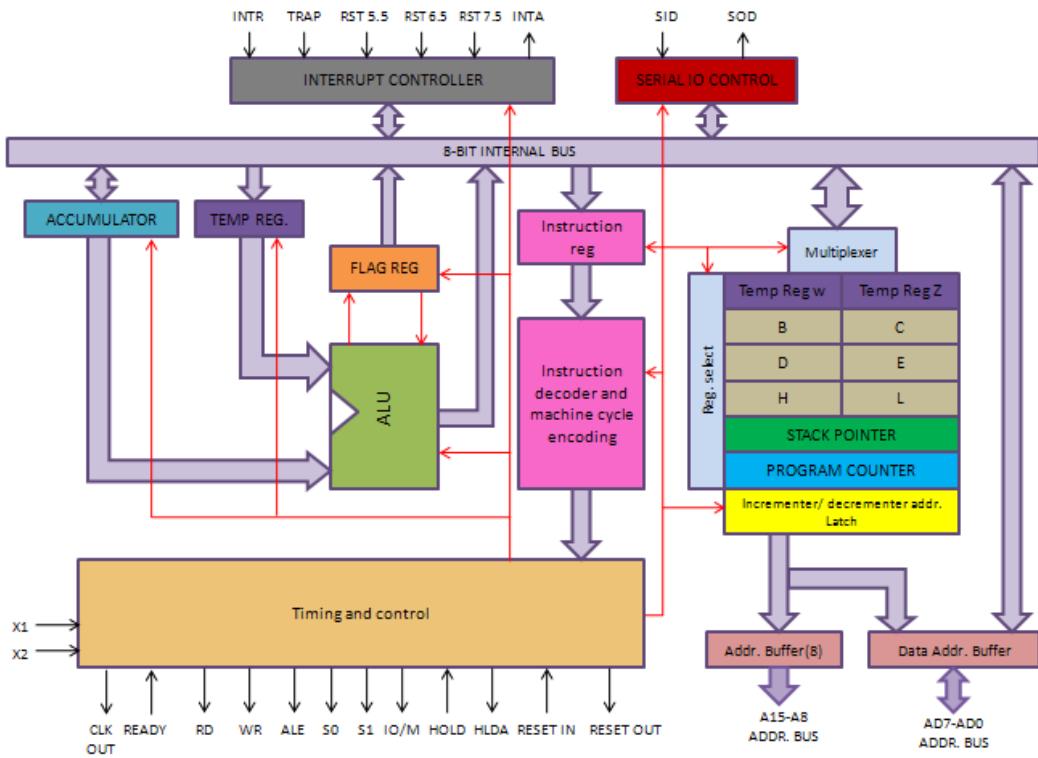
5. Clock signals

- **X1, X2** A crystal (RC, LC N/W) is connected at these two pins and is used to set frequency of the internal clock generator.
- This frequency is internally divided by 2.
- **CLK OUT** This signal is used as the system clock forexternal devices connected with the microprocessor.

14. Draw the block diagram and explain the components of Intel 8085 .

Architecture of 8085 Microprocessor

Below figure shows the architecture of 8085 microprocessor



The following are the different blocks in the 8085 processor.

ALU:

It is 8-bit ALU. It can perform arithmetic and logical operations on 8-bit data. If an operation needs to be performed on 16-bit data, it needs to be broken into two 8-bit parts and each 8-bit operation should be performed on each 8-bit data. It takes operand inputs from accumulator and a temporary register. Result of the operation is stored in accumulator. Depending on the result of operation, flags in flag register values will be changed.

Accumulator

It is a 8-bit register which is used to perform arithmetical and logical operation. It stores the output of any operation. It also works as registers for i/o accesses.

Temporary Register

It is a 8-bit register which is used to hold the data on which the accumulator is computing operation. It is also called as operand register because it provides operands to ALU.

Register Array:

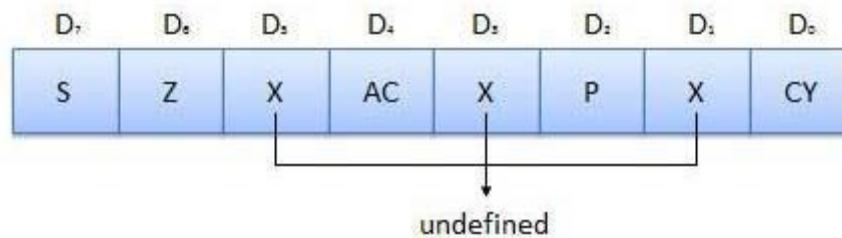
- 8085 has six general purpose registers B, C, D, E, H, L. They can be used as pairs

to hold 16-bit data as BC, DE, HL.

- Flag register contains five flags, namely S, Z, CY, AC, P flags.
- 8085 has two 16-bit register PC and SP.
- PC always consists of address of next instruction to be executed.
- SP always points to top of stack. i.e. address of top memory location of stack. Stack is a data structure. It is used to store return addresses whenever call to subprograms or an interrupt occurs.
- Two temporary registers W, Z are also present.
- Increment and decrement address latch is for incrementing the PC content for every fetch cycle.

Flag register:

As already explained contents of flag register will be changed according to the result of ALU operation. Below figure shows the flag register format of 8085.



Sign flag (S): when the result of ALU operation is negative sign flag is set. If the result is positive, then sign flag is reset

- Zero flag (Z): when the result of ALU operation is zero, Zero flag is set. If the result is non-zero then flag is reset.
- Auxiliary carry (AC): If an ALU operation results in carry from lower nibble to upper nibble (or) bit D3 to bit D4, Auxiliary flag is set. Else it is reset. This flag is used in BCD arithmetic.
- Parity flag (P): If the result contains even number of ones, the flag is set else it is reset. So the parity flag is odd parity bit.
- Carry flag (CY): If the arithmetic operation results in carry, CY flag is set, else it is reset.

Instruction Register and Decoding:

- Instruction register holds instruction that is fetched from memory.
- Instruction decoder decodes the opcode (which is part of fetched instruction present in instruction register).
- Instruction register is not accessible to the programmer.

Interrupt Controller:

- 8085 has 5 external interrupts. TRAP, INTR, RST 5.5, RST 6.5, and RST 7.5. Whenever processor gets interrupt it finishes current instruction execution and issues INTA (interrupt acknowledge) signal to the peripheral which raised the interrupt and goes to execute interrupt service routine. Interrupt controller controls the interrupts.

Serial I/O control:

Serial data can be sent out using SOD pin and serial data can be read from SID pin. It controls serial IO related operations.

Program Counter

- It is a 16 bit register used as memory pointer.
- It stores the memory address of the next instruction to be executed. So we can say that this register is used to sequencing the program.
- Generally the memory have 16 bit addresses so that it has 16 bit memory. The program counter is set to 0000H.

Stack Pointer

- It is also a 16 bit register used as memory pointer.
- It points to the memory location called stack.
- Generally stack is a reserved portion of memory where information can be stores or taken back together.

Timing and Control Unit

- It provides timing and control signal to the microprocessor to perform the various operation.

- It has three control signal.
- It controls all external and internal circuits.

There are three control signal:

1. ALE-Airthmetic Latch Enable, It provides control signal to synchronize the components of microprocessor.
2. RD- This is active low used for reading operation.
3. WR-This is active low used for writing operation.

Module 2

	Part A
1.	<p>What are the types of addressing mode in 8085 microprocessor?</p> <p>To perform any operation, we have to give the corresponding instructions to the microprocessor. In each instruction, programmer has to specify 3 things</p> <ul style="list-style-type: none">: 1. Operation to be performed.2. Address of source of data.3. Address of destination of result<ul style="list-style-type: none">• The method by which the address of source of data or the address of destination of result is given in the instruction is called Addressing Modes.• The term addressing mode refers to the way in which the operand of the instruction is specified. <p>Intel 8085 uses the following addressing modes:</p> <ul style="list-style-type: none">1. Direct Addressing Mode2. Register Addressing Mode3. Register Indirect Addressing Mode4. Immediate Addressing Mode5. Implicit Addressing Mode <p><u>1. . Direct Addressing Mode</u></p> <p>In this mode, the address of the operand is given in the instruction itself.</p> <div style="background-color: #8B4513; color: white; padding: 10px; text-align: center;">LDA 2500 H Load the contents of memory location 2500 H in accumulator.</div> <ul style="list-style-type: none">• LDA is the operation.• 2500 H is the address of source.• Accumulator is the destination. <p><u>2. Register Addressing Mode</u></p> <p>In this mode, the operand is in general purpose register.</p> <div style="background-color: #8B4513; color: white; padding: 10px; text-align: center;">MOV A, B Move the contents of register B to A.</div> <ul style="list-style-type: none">• MOV is the operation.• B is the source of data.• A is the destination.

3. Register Indirect Addressing Mode

In this mode, the address of operand is specified by a register pair.

MOV A, M**Move data from memory location
specified by H-L pair to accumulator.**

- MOV is the operation.
- M is the memory location specified by H-L register pair
- . A is the destination.

4.. Immediate Addressing Mode

In this mode, the operand is specified within the instruction itself

MVI A, 05 H**Move 05 H in accumulator.**

- MVI is the operation.
- 05 H is the immediate data (source).
- A is the destination.

5. Implicit Addressing Mode

If address of source of data as well as address of destination of result is fixed, then there is no need to give any operand along with the

CMA**Complement accumulator.**

instruction.

- CMA is the operation.
- A is the source.
- A is the destination

2. What are the flags available in 8085 processor ?

Flag register in 8085 microprocessor

The Flag register is a Special Purpose Register. Depending upon the value of result after any arithmetic and logical operation the flag bits become set (1) or reset (0). In 8085 microprocessor, flag register consists of 8 bits and only 5 of them are useful.

The 5 flags are:

D7	D6	D5	D4	D3	D2	D1	D0
S	Z	x	AC	x	P	x	CY

The positions marked by “x” are to be considered as don't care bits in the flags register.

1.Sign Flag (S) –

- After any operation if the MSB (B(7)) of the result is 1, it indicates the number is negative and the sign flag becomes set, i.e. 1.
- If the MSB is 0, it indicates the number is positive and the sign flag becomes reset i.e. 0.
from 00H to 7F, sign flag is 0
from 80H to FF, sign flag is 1

1- MSB is 1 (negative)

0- MSB is 0 (positive)

Example:

MVI A 30 (load 30H in register A)

MVI B 40 (load 40H in register B)

SUB B (A = A – B)

These set of instructions will set the sign flag to 1 as 30 – 40 is a negative number.

2. Zero Flag (Z) –

- After any arithmetical or logical operation if the result is 0 (00)H, the zero flag becomes set i.e. 1, otherwise it becomes reset i.e. 0.
00H zero flag is 1.
from 01H to FFH zero flag is 0

1- zero result

0- non-zero result

Example:

MVI A 10 (load 10H in register A)

SUB A (A = A - A)

These set of instructions will set the zero flag to 1 as 10H - 10H is 00H

3. Auxiliary Carry Flag (AC) –

- This flag is used in BCD number system(0-9).
- If after any arithmetic or logical operation D(3) generates any carry and passes on to B(4) this flag becomes set i.e. 1, otherwise it becomes reset i.e. 0.
- This is the only flag register which is not accessible by the programmer

1-carry out from bit 3 on addition or borrow into bit 3 on subtraction

0-otherwise

Example:

MOV A 2B (load 2BH in register A)

MOV B 39 (load 39H in register B)

ADD B (A = A + B)

These set of instructions will set the auxiliary carry flag to 1, as on adding 2B and 39, addition of lower order nibbles B and 9 will generate a carry.

4. Parity Flag (P) –

- If after any arithmetic or logical operation the result has even parity, an even number of 1 bits, the parity register becomes set i.e. 1, otherwise it becomes reset i.e. 0.

1-accumulator has even number of 1 bits

0-accumulator has odd parity

Example:

MVI A 05 (load 05H in register A)

This instruction will set the parity flag to 1 as the BCD code of 05H is 00000101, which contains even number of ones i.e. 2.

5. Carry Flag (CY) –

- Carry is generated when performing n bit operations and the result is more than n

	<p>bits, then this flag becomes set i.e. 1, otherwise it becomes reset i.e. 0.</p> <ul style="list-style-type: none"> • During subtraction (A-B), if A>B it becomes reset and if (A<B) it becomes set. Carry flag is also called borrow flag. <p>1-carry out from MSB bit on addition or borrow into MSB bit on subtraction 0-no carry out or borrow into MSB bit</p> <p>Example:</p> <p>MVI A 30 (load 30H in register A) MVI B 40 (load 40H in register B) SUB B (A = A – B)</p> <p>These set of instructions will set the carry flag to 1 as 30 – 40 generates a carry/borrow.</p>
3.	<p>Explain instruction format of 8085?</p> <p>An instruction is a command to the microprocessor to perform a given task on a specified data.</p> <ul style="list-style-type: none"> • Each instruction has two parts: one is task to be performed, called the <u>operation code</u> (opcode), and the second is the data to be operated on, called the <u>operand</u>. The operand (or data) can be specified in various ways. It may include 8-bit (or 16-bit) data, an internal register, a memory location, or 8-bit (or 16-bit) address. In some instructions, the operand is implicit. <p>Instruction word size</p> <p>The 8085 instruction set is classified into the following three groups according to word size:</p> <ul style="list-style-type: none"> ü One-word or 1-byte instructions ü Two-word or 2-byte instructions ü Three-word or 3-byte instructions <p>In the 8085, "byte" and "word" are synonymous because it is an 8-bit microprocessor. However, instructions are commonly referred to in terms of bytes rather than words.</p> <p><u>1 One-Byte Instructions</u></p> <p>A 1-byte instruction includes the opcode and operand in the same byte. Operand(s) are internal register and are coded into the instruction</p>

FORMAT :

Opcode

1 byte

Table 2.1 Example for 1 byte Instruction

Task	Op code	Operand	Binary Code	Hex Code
Copy the contents of the accumulator in the register C.	MOV	C,A	0100 1111	4FH
Add the contents of register B to the contents of the accumulator.	ADD	B	1000 0000	80H
Invert (compliment) each bit in the accumulator.	CMA		0010 1111	2FH

These instructions are 1-byte instructions performing three different tasks. In the first instruction, both operand registers are specified. In the second instruction, the operand B is specified and the accumulator is assumed. Similarly, in the third instruction, the accumulator is assumed to be the implicit operand. These instructions are stored in 8-bit binary format in memory; each requires one memory location.

2 Two-Byte Instructions

In a two-byte instruction, the first byte specifies the operation code and the second byte specifies the operand. Source operand is a data byte immediately following the opcode. For example:

FORMAT :

Opcode	Operand
--------	---------

2 bytes

Table 2.2 Example for 2 byte Instruction

Table 2.2 Example for 2 byte Instruction

Task	Opcode	Operand	Binary Code	Hex Code	
Load an 8-bit data byte in the accumulator.	MVI	A, Data	0011 1110 DATA	3E Data	First Byte Second Byte

The instruction would require two memory locations to store in memory.

3 Three-Byte Instructions

In a three-byte instruction, the first byte specifies the opcode, and the following two bytes specify the 16-bit address. Note that the second byte is the low-order address and the third byte is the high-order address. opcode + data byte + data byte

FORMAT :

Opcode	Operand	Operand
--------	---------	---------

3 bytes

Table 3.3 Example for 3 byte Instruction

Task	Opcode	Operand	Binary code	Hex Code	
Transfer the program sequence to the memory location 2085H.	JMP	2085H	1100 0011 1000 0101 0010 0000	C3 85 20	First byte Second Byte Third Byte

This instruction would require three memory locations to store in memory.

Three byte instructions - opcode + data byte + data byte

4. Specify the size of data, address, memory word and memory capacity of 8085 microprocessor.
Size of data = 8-bit

	<p>Size of address = 16-bit</p> <p>Memory Word = 8-bit</p> <p>Memory capacity = Upto 64Kb</p>
5.	<p>Discuss the steps involved in fetch a byte in 8085. Explain instruction format of 8085?</p> <p>Steps involved in Fetch byte</p> <ul style="list-style-type: none"> • The pc places the 16-bit memory address on the address bus • The control unit sends the control signal RD to enable the memory chip • The byte from the memory location is placed on the data bus • The byte is placed in the instruction decoder of the microprocessor and the task is carried out according to the instruction. <p>An instruction is a command to the microprocessor to perform a given task on a specified data.</p> <ul style="list-style-type: none"> • Each instruction has two parts: one is task to be performed, called the <u>operation code</u> (opcode), and the second is the data to be operated on, called the <u>operand</u>. The operand (or data) can be specified in various ways. It may include 8-bit (or 16-bit) data, an internal register, a memory location, or 8-bit (or 16-bit) address. In some instructions, the operand is implicit. <p>Instruction word size</p> <p>The 8085 instruction set is classified into the following three groups according to word size:</p> <ul style="list-style-type: none"> ü One-word or 1-byte instructions ü Two-word or 2-byte instructions ü Three-word or 3-byte instructions <p>In the 8085, "byte" and "word" are synonymous because it is an 8-bit microprocessor. However, instructions are commonly referred to in terms of bytes rather than words.</p> <p><u>1 One-Byte Instructions</u></p> <p>A 1-byte instruction includes the opcode and operand in the same byte. Operand(s) are internal register and are coded into the instruction</p>

FORMAT :

Opcode

1 byte

Table 2.1 Example for 1 byte Instruction

Task	Op code	Operand	Binary Code	Hex Code
Copy the contents of the accumulator in the register C.	MOV	C,A	0100 1111	4FH
Add the contents of register B to the contents of the accumulator.	ADD	B	1000 0000	80H
Invert (compliment) each bit in the accumulator.	CMA		0010 1111	2FH

These instructions are 1-byte instructions performing three different tasks. In the first instruction, both operand registers are specified. In the second instruction, the operand B is specified and the accumulator is assumed. Similarly, in the third instruction, the accumulator is assumed to be the implicit operand. These instructions are stored in 8-bit binary format in memory; each requires one memory location.

2 Two-Byte Instructions

In a two-byte instruction, the first byte specifies the operation code and the second byte specifies the operand. Source operand is a data byte immediately following the opcode. For example:

FORMAT :

Opcode	Operand
--------	---------

2 bytes

Table 2.2 Example for 2 byte Instruction

Table 2.2 Example for 2 byte Instruction

Task	Opcode	Operand	Binary Code	Hex Code	
Load an 8-bit data byte in the accumulator.	MVI	A, Data	0011 1110 DATA	3E Data	First Byte Second Byte

The instruction would require two memory locations to store in memory.

3 Three-Byte Instructions

In a three-byte instruction, the first byte specifies the opcode, and the following two bytes specify the 16-bit address. Note that the second byte is the low-order address and the third byte is the high-order address. opcode + data byte + data byte

FORMAT :

Opcode	Operand	Operand
--------	---------	---------

3 bytes

Table 3.3 Example for 3 byte Instruction

Task	Opcode	Operand	Binary code	Hex Code	
Transfer the program sequence to the memory location 2085H.	JMP	2085H	1100 0011 1000 0101 0010 0000	C3 85 20	First byte Second Byte Third Byte

This instruction would require three memory locations to store in memory.

Three byte instructions - opcode + data byte + data byte

Part B

6. Explain the following instructions with suitable example of each: (i) LXI. (ii) STA. (iii) SHLD. (iv) LDAX. (v) CMP.

I) LXI

Data Transfer Instructions

Opcode	Operand	Description
LXI	Reg. pair, 16-bit data	Load register pair immediate

- This instruction loads 16-bit data in the register pair.
- **Example:** LXI H, 2034 H

II) STA

In 8085 Instruction set, **STA** is a mnemonic that stands for STore Accumulator contents in memory. In this instruction, Accumulator 8-bit content will be stored to a memory location whose 16-bit address is indicated in the instruction as a16

STA: - the content of accumulator are copied into the memory location.

Eg: - STA 3000H (the content of accumulator is stored into the memory location 3000h)

III) SHLD

SHLD(store H and L register direct): - The contents of register L are stored into the memory location specified by the 16-bit address in the operand and the contents of H register are stored into the next memory location by incrementing the operand. The contents of registers HL are not altered. This is a 3-byte instruction, the second byte specifies the low-order address and the third byte specifies the high-order address.

Eg: - SHLD 3000H

IV) LDAX

Data Transfer Instructions

Opcode	Operand	Description
LDAX	B/D Register Pair	Load accumulator indirect

- The contents of the designated register pair point to a memory location.
 - This instruction copies the contents of that memory location into the accumulator.
 - The contents of either the register pair or the memory location are not altered.

• Example: LDAX B



v) CMP

In 8085 Instruction set, CMP is a mnemonic that stands for “CoMPareAccumulator” and here R stands for any of the following registers, or memory location M pointed by HL pair.

R = A, B, C, D, E, H, L, or M

This instruction is used to compare contents of the Accumulator with given register R. The result of compare operation will be stored in the Temp register. Temp is not a GPR (General Purpose Register) but an internal register that is not accessible to the programmer.

CMP: - (compare register or memory with accumulator) The contents of the operand register or memory are M compared with the contents of the accumulator. Both contents are preserved . The result of the comparison is shown by setting the flags of the PSW as follows:
if $(A) < \text{reg}/\text{mem}$: carry flag is set.

if (A) = reg/mem:

zero flag is set.

if (A) > reg/mem: carry and zero flag

() 8, , , 8

Eg. - CMP B

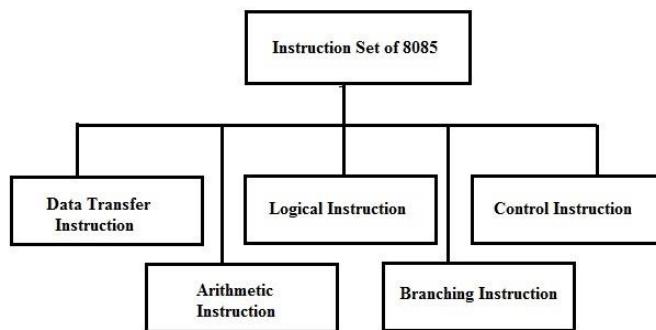
CMP M

7. Discuss about each group of instruction set of Intel 8085 microprocessor with examples.

A binary command that is used to perform a function in the microprocessor over provided data is known as instruction. A set of instructions is known as an instruction set that decides the

microprocessor function. Every instruction includes two parts like Opcode & the Operand where Opcode is used to specify the function to be executed & operand gives the data to be functioned on.

Classification of Instruction Set of 8085



Data Transfer Instructions

- These instructions move data between registers, or between memory and registers.
- These instructions copy data from source to destination.
- While copying, the contents of source are not modified.

1

Data Transfer Instructions

Opcode	Operand	Description
MOV	Rd, Rs M, Rs Rd, M	Copy from source to destination.

- This instruction copies the contents of the source register into the destination register.
- The contents of the source register are not altered.
- If one of the operands is a memory location, its location is specified by the contents of the HL registers.
- **Example:** MOV B, C or MOV B, M

2

Data Transfer Instructions

Opcode	Operand	Description
MVI	Rd, Data M, Data	Move immediate 8-bit

- The 8-bit data is stored in the destination register or memory.
- If the operand is a memory location, its location is specified by the contents of the H-L registers.
- **Example:** MVI B, 57H or MVI M, 57H

Clip slide

BEFORE EXECUTION

A	F
B	C
D	E
H	L

AFTER EXECUTION

A	F
B	60
D	E
H	L

MVI B,60H

BEFORE EXECUTION

204FH
HL=2050H
2051H

AFTER EXECUTION

204FH
HL=2050H
2051H

MVI M,40H

3

Data Transfer Instructions

Opcode	Operand	Description
LDA	16-bit address	Load Accumulator

- The contents of a memory location, specified by a 16-bit address in the operand, are copied to the accumulator.
- The contents of the source are not altered.
- Example:** LDA 2034H

BEFORE EXECUTION

A	
2000H	30

AFTER EXECUTION

A	30
2000H	30

LDA 2000H

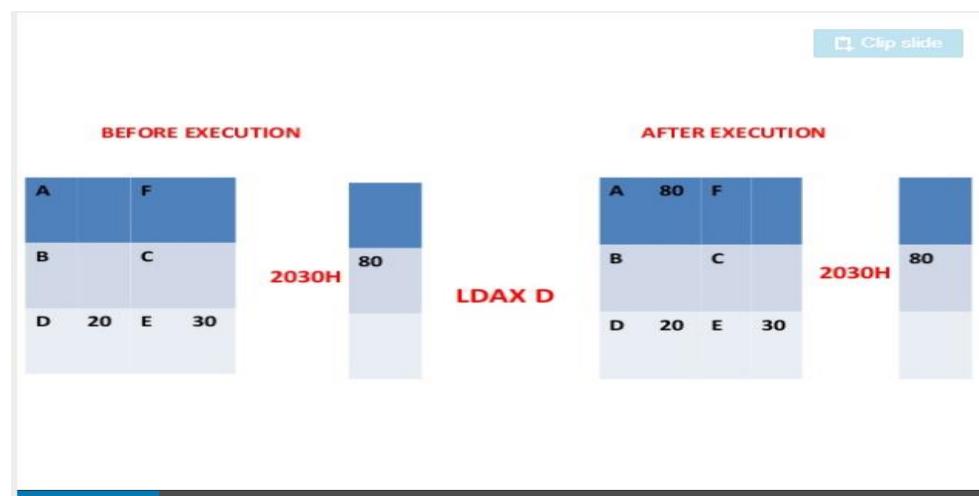
In 8085 Instruction set, **LDAX** is a mnemonic that stands for LoaD Accumulator from memory pointed by eXtended register pair denoted as “rp” in the instruction. This instruction uses register indirect addressing for specifying the data. It occupies only 1-Byte in the memory.

4

Data Transfer Instructions

Opcode	Operand	Description
LDAX	B/D Register Pair	Load accumulator indirect

- The contents of the designated register pair point to a memory location.
- This instruction copies the contents of that memory location into the accumulator.
- The contents of either the register pair or the memory location are not altered.
- Example:** LDAX B



5

Data Transfer Instructions

Opcode	Operand	Description
LXI	Reg. pair, 16-bit data	Load register pair immediate

- This instruction loads 16-bit data in the register pair.
- **Example:** LXI H, 2034 H

6

Data Transfer Instructions

Opcode	Operand	Description
LHLD	16-bit address	Load H-L registers direct

- This instruction copies the contents of memory location pointed out by 16-bit address into register L.
- It copies the contents of next memory location into register H.
- **Example:** LHLD 2040 H

1

Arithmetic Instructions

Opcode	Operand	Description
ADD	R M	Add register or memory to accumulator

- The contents of register or memory are added to the contents of accumulator.
- The result is stored in accumulator.
- If the operand is memory location, its address is specified by H-L pair.
- All flags are modified to reflect the result of the addition.
- **Example:** ADD B or ADD M

BEFORE EXECUTION				AFTER EXECUTION			
A	20					A	50
B			C	30		B	C
D					D	E	
H					H	L	

ADD C
A=A+R

BEFORE EXECUTION				AFTER EXECUTION			
A	20					A	30
B			C			B	C
D					D	E	
H	20	L	50	10	2050	10	2050

ADD M
A=A+M

Addition

- Any 8-bit number, or the contents of register, or the contents of memory location can be added to the contents of accumulator.
- The result (sum) is stored in the accumulator.
- No two other 8-bit registers can be added directly.
- **Example:** The contents of register B cannot be added directly to the contents of register C.

Subtraction

- Any 8-bit number, or the contents of register, or the contents of memory location can be subtracted from the contents of accumulator.
- The result is stored in the accumulator.
- Subtraction is performed in 2's complement form.
- If the result is negative, it is stored in 2's complement form.
- No two other 8-bit registers can be subtracted directly.

Increment / Decrement

- The 8-bit contents of a register or a memory location can be incremented or decremented by 1.
- The 16-bit contents of a register pair can be incremented or decremented by 1.
- Increment or decrement can be performed on any register or a memory location.

2

Arithmetic Instructions

Opcode	Operand	Description
ADC	R M	Add register or memory to accumulator with carry

- The contents of register or memory and Carry Flag (CY) are added to the contents of accumulator.
- The result is stored in accumulator.
- If the operand is memory location, its address is specified by H-L pair.
- All flags are modified to reflect the result of the addition.
- **Example:** ADC B or ADC M

BEFORE EXECUTION		AFTER EXECUTION	
CY	1	CY	0
A	50	A	71
B		B	C 20
D	E	D	E
H	L	H	L
ADC C A=A+R+CY			

BEFORE EXECUTION		AFTER EXECUTION	
CY	1	CY	0
A	20	A	51
2050H	30	2050H	30
H	20 L 50	H	20 L 50
ADC M A=A+M+CY			

3

Arithmetic Instructions

Opcode	Operand	Description
SUB	R M	Subtract register or memory from accumulator

- The contents of the register or memory location are subtracted from the contents of the accumulator.
- The result is stored in accumulator.
- If the operand is memory location, its address is specified by H-L pair.
- All flags are modified to reflect the result of subtraction.
- Example:** SUB B or SUB M

BEFORE EXECUTION

A	50	
B	30	C
D	E	
H	L	

AFTER EXECUTION

A	20	
B	30	C
D	E	
H	L	

SUB B
A=A-R

BEFORE EXECUTION

A	50
	1020H
H	L
10	20

AFTER EXECUTION

A	40
	1020H
H	L
10	20

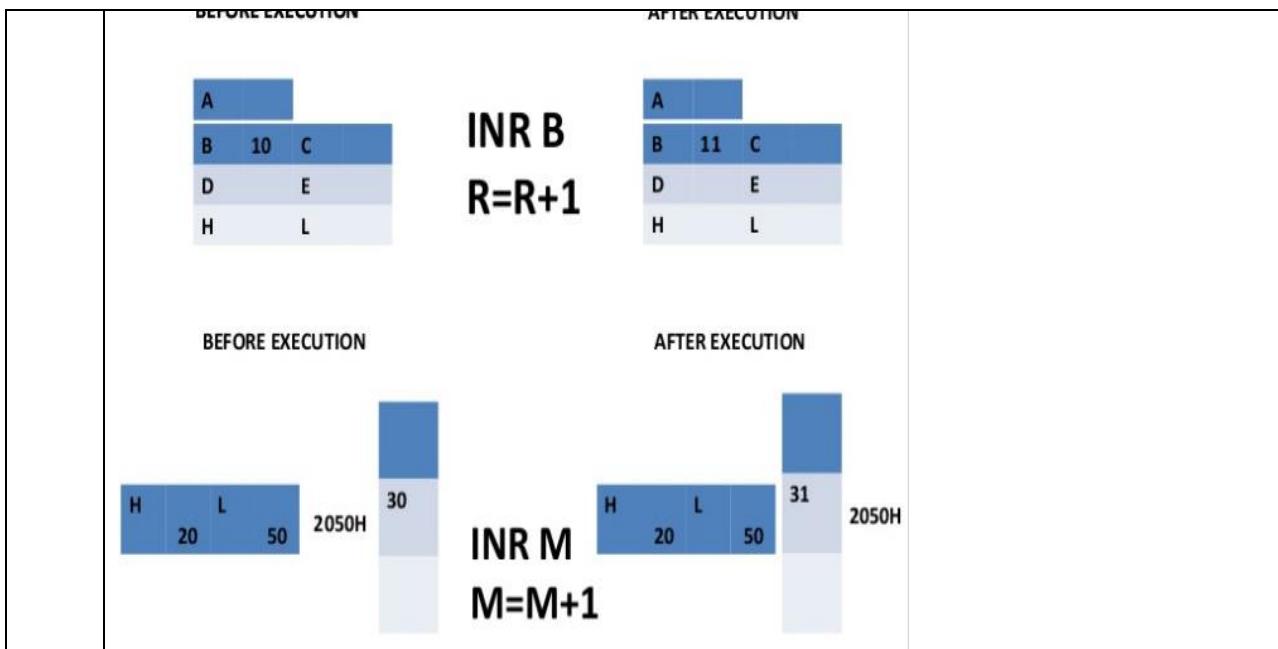
SUB M
A=A-M

4

Arithmetic Instructions

Opcode	Operand	Description
INR	R M	Increment register or memory by 1

- The contents of register or memory location are incremented by 1.
- The result is stored in the same place.
- If the operand is a memory location, its address is specified by the contents of H-L pair.
- Example:** INR B or INR M



5

Arithmetic Instructions

Opcode	Operand	Description
DCR	R M	Decrement register or memory by 1

- The contents of register or memory location are decremented by 1.
- The result is stored in the same place.
- If the operand is a memory location, its address is specified by the contents of H-L pair.
- Example:** DCR B or DCR M

Logical Instructions

- These instructions perform logical operations on data stored in registers, memory and status flags.
- The logical operations are:
 - AND
 - OR
 - XOR
 - Rotate
 - Compare
 - Complement

AND, OR, XOR

- Any 8-bit data, or the contents of register, or memory location can logically have
 - AND operation
 - OR operation
 - XOR operation
- with the contents of accumulator.
- The result is stored in accumulator.

Rotate

- Each bit in the accumulator can be shifted either left or right to the next position.

1

Compare

- Any 8-bit data, or the contents of register, or memory location can be compared for:

- Equality
- Greater Than
- Less Than

with the contents of accumulator.

- The result is reflected in status flags.

Complement

- The contents of accumulator can be complemented.
- Each 0 is replaced by 1 and each 1 is replaced by 0.

Logical Instructions

Opcode	Operand	Description
CMP	R M	Compare register or memory with accumulator

- The contents of the operand (register or memory) are compared with the contents of the accumulator.
- Both contents are preserved .
- The result of the comparison is shown by setting the flags of the PSW as follows:

Logical Instructions

Opcode	Operand	Description
CMP	R M	Compare register or memory with accumulator

- if $(A) < (\text{reg}/\text{mem})$: carry flag is set
- if $(A) = (\text{reg}/\text{mem})$: zero flag is set
- if $(A) > (\text{reg}/\text{mem})$: carry and zero flags are reset.
- **Example:** CMP B or CMP M

2

Logical Instructions

Opcode	Operand	Description
CPI	8-bit data	Compare immediate with accumulator

- The 8-bit data is compared with the contents of accumulator.
- The values being compared remain unchanged.
- The result of the comparison is shown by setting the flags of the PSW as follows:

3

Logical Instructions

Opcode	Operand	Description
CPI	8-bit data	Compare immediate with accumulator

- if $(A) < \text{data}$: carry flag is set
- if $(A) = \text{data}$: zero flag is set
- if $(A) > \text{data}$: carry and zero flags are reset
- **Example:** CPI 89H

Logical Instructions

Opcode	Operand	Description
ORA	R	Logical OR register or memory with accumulator
	M	

- The contents of the accumulator are logically ORed with the contents of the register or memory.
- The result is placed in the accumulator.
- If the operand is a memory location, its address is specified by the contents of H-L pair.
- S, Z, P are modified to reflect the result.
- CY and AC are reset.
- Example:** ORA B or ORA M.

Opcode	Operand	Description
RLC	None	Rotate accumulator left

- Each binary bit of the accumulator is rotated left by one position.
- Bit D₇ is placed in the position of D₀ as well as in the Carry flag.
- CY is modified according to bit D₇.
- S, Z, P, AC are not affected.
- Example:** RLC.

	<p>The diagram illustrates the RRC (Rotate Right Circular) instruction. It shows a 9-bit register with bits B7 to B0. Bit B7 is labeled 'CY'. Four blue arrows point from right to left, indicating the rotation of bits B6 to B0. A green arrow points from the rightmost position back to the CY bit. Below the register, the bits are shown in their new positions: B7, B6, B5, B4, B3, B2, B1, B0, B7.</p>						
6	<table border="1"> <thead> <tr> <th>Opcode</th><th>Operand</th><th>Description</th></tr> </thead> <tbody> <tr> <td>RRC</td><td>None</td><td>Rotate accumulator right</td></tr> </tbody> </table>	Opcode	Operand	Description	RRC	None	Rotate accumulator right
Opcode	Operand	Description					
RRC	None	Rotate accumulator right					
<ul style="list-style-type: none"> • Each binary bit of the accumulator is rotated right by one position. • Bit Do is placed in the position of D7 as well as in the Carry flag. • CY is modified according to bit Do. • S, Z, P, AC are not affected. • Example: RRC. 							
<h2>Branching Instructions</h2>							
<ul style="list-style-type: none"> • The branching instruction alter the normal sequential flow. • These instructions alter either unconditionally or conditionally. 							

1

Branching Instructions

Opcode	Operand	Description
JMP	16-bit address	Jump unconditionally

- The program sequence is transferred to the memory location specified by the 16-bit address given in the operand.
- Example:** JMP 2034 H.

Conditional Jumps

Instruction Code	Description	Condition For Jump
JC	Jump on carry	CY=1
JNC	Jump on not carry	CY=0
JP	Jump on positive	S=0
JM	Jump on minus	S=1
JPE	Jump on parity even	P=1
JPO	Jump on parity odd	P=0
JZ	Jump on zero	Z=1
JNZ	Jump on not zero	Z=0

2

Branching Instructions

Opcode	Operand	Description
Jx	16-bit address	Jump conditionally

- The program sequence is transferred to the memory location specified by the 16-bit address given in the operand based on the specified flag of the PSW.
- Example:** JZ 2034 H.

Jump Conditionally

Opcode	Description	Status Flags
JC	Jump if Carry	CY = 1
JNC	Jump if No Carry	CY = 0
JP	Jump if Positive	S = 0
JM	Jump if Minus	S = 1
JZ	Jump if Zero	Z = 1
JNZ	Jump if No Zero	Z = 0
JPE	Jump if Parity Even	P = 1
JPO	Jump if Parity Odd	P = 0

3

Branching Instructions

Opcode	Operand	Description
CALL	16-bit address	Call unconditionally

- The program sequence is transferred to the memory location specified by the 16-bit address given in the operand.
- Before the transfer, the address of the next instruction after CALL (the contents of the program counter) is pushed onto the stack.
- Example:** CALL 2034 H.

4

Branching Instructions

Opcode	Operand	Description
RET	None	Return unconditionally

- The program sequence is transferred from the subroutine to the calling program.
- The two bytes from the top of the stack are copied into the program counter, and program execution begins at the new address.
- Example:** RET.

1

Control Instructions

Opcode	Operand	Description
NOP	None	No operation

- No operation is performed.
- The instruction is fetched and decoded but no operation is executed.
- **Example:** NOP



2

Control Instructions

Opcode	Operand	Description
HLT	None	Halt

- The CPU finishes executing the current instruction and halts any further execution.
- An interrupt or reset is necessary to exit from the halt state.
- **Example:** HLT

3

Control Instructions

Opcode	Operand	Description
DI	None	Disable interrupt

- The interrupt enable flip-flop is reset and all the interrupts except the TRAP are disabled.
- No flags are affected.
- **Example:** DI

4

Control Instructions

Opcode	Operand	Description
EI	None	Enable interrupt

- The interrupt enable flip-flop is set and all interrupts are enabled.
- No flags are affected.
- This instruction is necessary to re-enable the interrupts (except TRAP).
- **Example:** EI

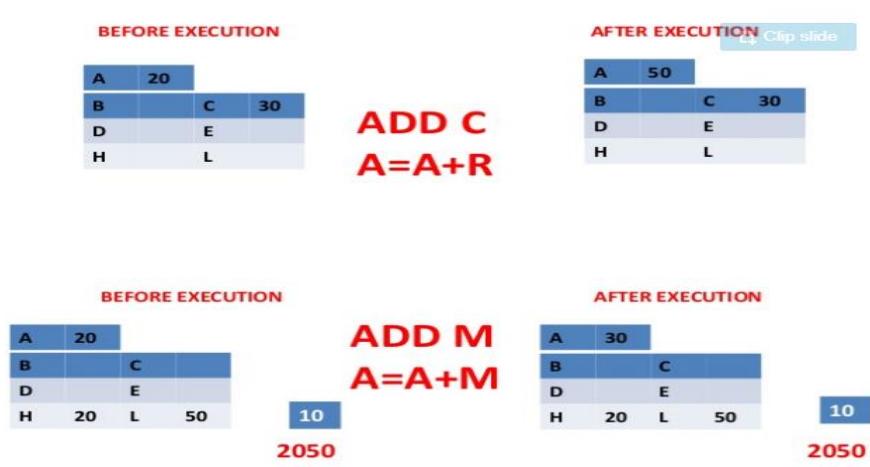
8.

Explain any five arithmetic group instructions of Intel 8085.

Arithmetic Instructions

Opcode	Operand	Description
ADD	R M	Add register or memory to accumulator

- The contents of register or memory are added to the contents of accumulator.
- The result is stored in accumulator.
- If the operand is memory location, its address is specified by H-L pair.
- All flags are modified to reflect the result of the addition.
- Example:** ADD B or ADD M



Addition

- Any 8-bit number, or the contents of register, or the contents of memory location can be added to the contents of accumulator.
- The result (sum) is stored in the accumulator.
- No two other 8-bit registers can be added directly.
- Example:** The contents of register B cannot be added directly to the contents of register C.

Subtraction

- Any 8-bit number, or the contents of register, or the contents of memory location can be subtracted from the contents of accumulator.
- The result is stored in the accumulator.
- Subtraction is performed in 2's complement form.
- If the result is negative, it is stored in 2's complement form.
- No two other 8-bit registers can be subtracted directly.

Increment / Decrement

- The 8-bit contents of a register or a memory location can be incremented or decremented by 1.
- The 16-bit contents of a register pair can be incremented or decremented by 1.
- Increment or decrement can be performed on any register or a memory location.

2

Arithmetic Instructions

Opcode	Operand	Description
ADC	R M	Add register or memory to accumulator with carry

- The contents of register or memory and Carry Flag (CY) are added to the contents of accumulator.
- The result is stored in accumulator.
- If the operand is memory location, its address is specified by H-L pair.
- All flags are modified to reflect the result of the addition.
- **Example:** ADC B or ADC M

BEFORE EXECUTION		AFTER EXECUTION	
CY	1	CY	0
A	50	A	71
B		B	C 20
D	E	D	E
H	L	H	L
ADC C A=A+R+CY			

BEFORE EXECUTION		AFTER EXECUTION	
CY	1	CY	0
A	20	A	51
2050H	30	2050H	30
H	20 L 50	H	20 L 50
ADC M A=A+M+CY			

3

Arithmetic Instructions

Opcode	Operand	Description
SUB	R M	Subtract register or memory from accumulator

- The contents of the register or memory location are subtracted from the contents of the accumulator.
- The result is stored in accumulator.
- If the operand is memory location, its address is specified by H-L pair.
- All flags are modified to reflect the result of subtraction.
- Example:** SUB B or SUB M

BEFORE EXECUTION

A	50	
B	30	C
D	E	
H	L	

AFTER EXECUTION

A	20	
B	30	C
D	E	
H	L	

SUB B
A=A-R

BEFORE EXECUTION

A	50
	1020H
H	L
10	20

AFTER EXECUTION

A	40
	1020H
H	L
10	20

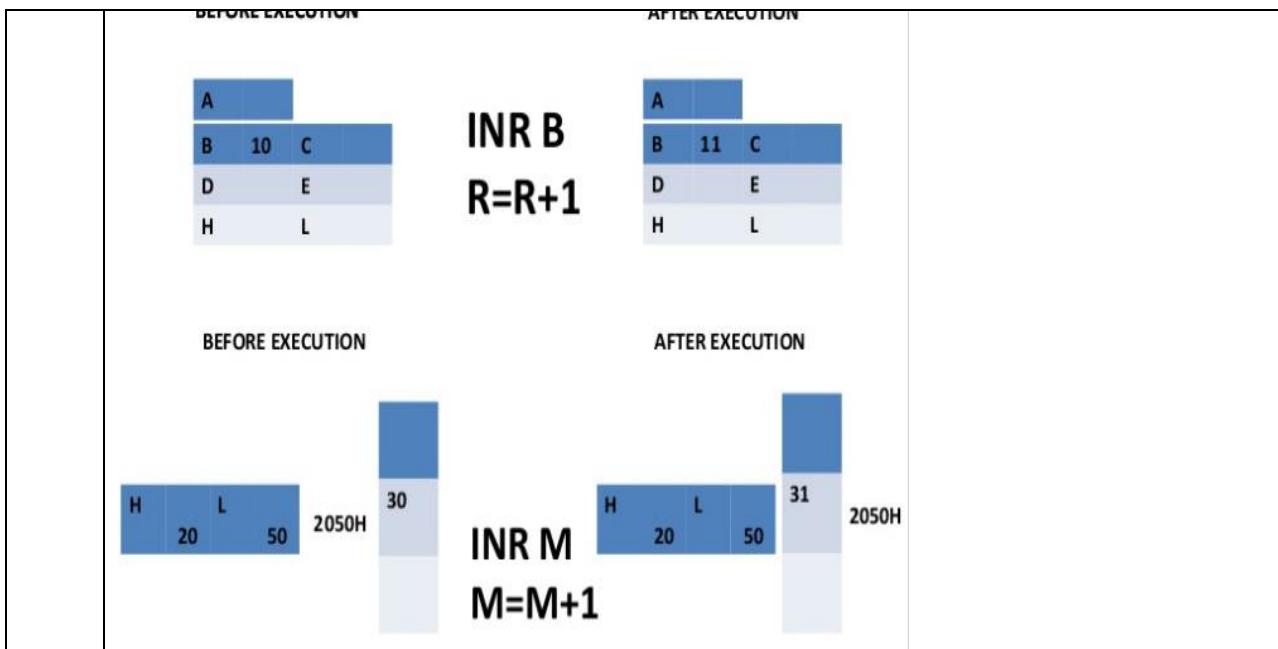
SUB M
A=A-M

4

Arithmetic Instructions

Opcode	Operand	Description
INR	R M	Increment register or memory by 1

- The contents of register or memory location are incremented by 1.
- The result is stored in the same place.
- If the operand is a memory location, its address is specified by the contents of H-L pair.
- Example:** INR B or INR M



5

Arithmetic Instructions

Opcode	Operand	Description
DCR	R M	Decrement register or memory by 1

- The contents of register or memory location are decremented by 1.
- The result is stored in the same place.
- If the operand is a memory location, its address is specified by the contents of H-L pair.
- Example:** DCR B or DCR M

Part C

9.	Explain the operations carried out when 8085 executes the instructions: (i) MOVA, M ii) XCHG (ii) DAD B. (iv) DAA. i) MOVA, M
----	--

	<ul style="list-style-type: none"> ○ MOV A, M - Move the content of the memory location, whose address is in H-L pair (i.e. 2500 H) to the accumulator. <p>ii) XCHG</p> <p>In 8085 Instruction set, there is one mnemonic XCHG, which stands for eXCHanGe. This is an instruction to exchange contents of HL register pair with DE register pair. This instruction uses implied addressing mode. As it is 1-Byte instruction, so it occupies only 1-Byte in the memory. After execution of this instruction, the content between H and D registers and L and E registers will get swapped respectively.</p> <p>XCHG: - Exchange H and L with D and E. The contents of register H are exchanged with the contents of register D, and the contents of register L are exchanged with the contents of register E.</p> <p>Eg: - XCHG</p> <p>(iii) DAD B.</p> <p>DAD is a mnemonic, which stands for Double ADD</p> <p>The given command is DAD B, where B represents the BC register pair.</p> <p>The content of the HL register pair will get added to the BC register pair and the result thus produced will be stored on the HL register pair.</p> <p>(iv) DAA.</p> <p>DAA means decimal adjust accumulator and it is used in BCD addition.</p> <p>DAA instruction changes the binary values of contents of the accumulator to BCD.</p> <p>DAA: - Decimal adjust accumulator.</p> <p>The contents of the accumulator are changed from a binary value to two 4-bit binary coded decimal (BCD) digits. This is the only instruction that uses the auxiliary flag to perform the binary to BCD conversion, and the conversion procedure is described below. S, Z, AC, P, CY flags are altered to reflect the results of the operation.</p> <p>If the value of the low-order 4-bits in the accumulator is greater than 9 or if AC flag is set, the instruction adds 6 to the low-order four bits. If the value of the high-order 4-bits in the accumulator is greater than 9 or if the Carry flag is set, the instruction adds 6 to the high-order four bits.</p>
10.	<p>Explain in detail, the instruction set of 8085 microprocessor</p> <p>Refer Qno No:7</p>

Module 3

Part A	
1.	<p>State the functions of BIOS.</p> <ul style="list-style-type: none"> • The BIOS (Basic Input/Output System) is firmware stored in a chip on your computer's motherboard. It is the first program that runs when you turn on your computer. • The BIOS performs the POST, which initializes and tests your computer's hardware. Then it locates and runs your boot loader, or loads your operating system directly. • The BIOS will most likely be stored in a 32-pin chip, which can typically be identified by a silver or gold sticker that shows the name of the BIOS company. • The BIOS comprises several separate routines, serving different functions. The first part runs as soon as the machine is powered on. It inspects the computer to determine what hardware is fitted and then conducts some simple tests to check that everything is functioning normally.
2.	<p>Mention the types of RAM Packages.</p> <pre> graph TD CM[Computer Memory] --> IMP[Internal/Main/Primary] CM --> ES[External/Secondary] IMP --> ROM[ROM (Non-Volatile)] IMP --> RAMV[RAM (Volatile)] RAMV --> SRAM[SRAM] RAMV --> DRAM[DRAM] DRAM --> SDRAM[SDRAM] DRAM --> RDRAM[RDRAM] DRAM --> DDRSDRAM[DDR SDRAM] DDRSDRAM --> DDR1[DDR1] DDRSDRAM --> DDR2[DDR2] DDRSDRAM --> DDR3[DDR3] DDRSDRAM --> DDR4[DDR4] ES --> HDD[HDD] ES --> SSD[SSD] ES --> CD[Compact Disc] ES --> UFD[USB Flash Drive] </pre> <p>Two main types of RAM are:</p> <ul style="list-style-type: none"> • Static RAM • Dynamic RAM <p>Static RAM</p>

	<p>Static RAM is the full form of SRAM. In this type of RAM, data is stored using the state of a six transistor memory cell. Static RAM is mostly used as a cache memory for the processor (CPU).</p> <h2>Dynamic RAM</h2> <p>DRAM stands for Dynamic Random Access Memory. It is a type of RAM which allows you to stores each bit of data in a separate capacitor within a specific integrated circuit. Dynamic RAM is a standard computer memory of the many modern desktop computers.</p> <p>This type of RAM is a volatile memory that needs to be refreshed with voltage regularly. Else it loses the information stored on it.</p>
3.	<p>Describe the use of address bus, data bus, and control bus.</p> <ul style="list-style-type: none"> The system bus is a pathway composed of cables and connectors used to carry data between a computer microprocessor and the main memory. The bus provides a communication path for the data and control signals moving between the major components of the computer system. The system bus works by combining the functions of the three main buses: namely, 1. Data bus, 2. Address bus 3. control buses. Each of the three buses has its separate characteristics and responsibilities. <p>The system bus combines the functions of the three main buses, which are as follows:</p> <ul style="list-style-type: none"> The control bus carries the control, timing and coordination signals to manage the various functions across the system. The address bus is used to specify memory locations for the data being transferred. The data bus, which is a bidirectional path, carries the actual data between the processor, the memory and the peripherals.
4.	<p>Write any four features of PCI bus.</p> <p>The PCI stands for Peripheral Component Interconnect. It is high performance Bus that is used to integrate chips, processor, memory subsystems and expansion boards.</p> <ul style="list-style-type: none"> Singling Environment : Support both 3.3 and 5 volt signaling environments. Reliability: It offers the ability to replace modules without disturbing a system's

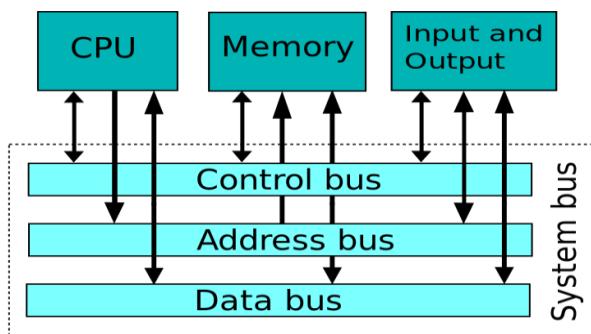
	<p>operation called as hot plug and hot swap.</p> <ul style="list-style-type: none"> ○ Speed: It can transfer up to 132 MB per second. ○ Configurability : The ability to configure a system automatically means automatically identify the interfacing systems and assigns new addresses. ○ Synchronous bus architecture : PCI is a synchronous bus where data transfer takes place according to a system clock. ○ 32 and 64 bit addressing : The PCI bus also supports 64 bit addressing with the same 32 bit connector. ○ Large bandwidth : It can handle both 32 bit as well as 64 bit data hence the maximum bandwidth will be 132 MB per second.
5.	<p>What is meant by co-processor?</p> <ul style="list-style-type: none"> • A coprocessor is a computer processor used to supplement the functions of the primary processor(the CPU). • Operations performed by the coprocessor may be floating point arithmetic, graphics, signal processing, string processing, cryptography or I/O interfacing with peripheral devices.
6.	<p>Write about super 1/O chip.</p> <p>Super i/o chip is a single chip which controls Slower i/o devices which are not controlled by South bridge.</p> <p>A super I/O chip combines interfaces for a variety of low-bandwidth devices. Super I/O is a class of I/O controller integrated circuits that began to be used on personal computer motherboards in the late 1980s, originally as add-in cards, later embedded on the motherboards. A super I/O chip combines interfaces for a variety of low-bandwidth devices. Now it is mostly merged with EC.</p> <p>Diagram of a motherboard, which supports many on-board peripheral functions as well as several expansion slots. ITE Super I/O chip (IT8712F) SMSC™ (now Microchip) Super I/O chip (FDC37M813) on IBM motherboard</p> <p>The functions below are usually provided by the super I/O if they are on the motherboard:</p> <ul style="list-style-type: none"> • A floppy-disk controller • An IEEE 1284-compatible parallel port (commonly used for printers) • One or more 16C550-compatible serial port UARTs • Keyboard controller for PS/2 keyboard and/or mouse <p>Most Super I/O chips include some additional low-speed devices, such as:</p> <ul style="list-style-type: none"> • Temperature, voltage, and fan speed interface • Thermal Zone • Chassis intrusion detection • Mainboard power management • LED management • PWM fan speed control • An IrDA Port controller

	<ul style="list-style-type: none"> • A game port (not provided by recent super I/O chips anymore because Windows XP is the last Windows OS to support a game port unless the vendor has a custom driver in the future OS) • A watchdog timer • A consumer IR receiver • A MIDI port • Some GPIO pins • Legacy Plug and Play or ACPI support for the included devices.
7.	<p>How can we select a motherboard?</p> <p>Motherboard Selection Criteria (Knowing What to Look For)</p> <p>It helps to think like an engineer when you make your selection. Consider every aspect and detail of the motherboards in question. For instance, you should consider both present usage as well as any future uses and upgrades. Technical support at a professional (as opposed to a user) level is extremely important.</p> <p>The following list includes some of the most important criteria to consider when selecting a motherboard:</p> <ul style="list-style-type: none"> • <u>Motherboard chipset:</u> The motherboard chipset is the backbone of a system and is perhaps the single most important part you'll consider. Compare the features of the available chipsets to ensure that the board will do what you want. For example, some chipsets include support for faster memory, PCIe 2.x cards, SATA 3Gbps drives, and optional RAID capabilities • <u>Processor socket:</u> The processor socket on a motherboard dictates the specific processor makes and models you will be able to install. • <u>Memory:</u> The type and amount of memory compatible with a system depends on the motherboard you choose. Most motherboards today support either DDR2 or DDR3 memory, in single, dual, or triple-channel operation. • <u>Form factor:</u> The form factor indicates the size and shape of the board, and must be compatible with the chassis or case and power supply. • <u>Bus slots:</u> Current systems offer one to five or more PCI and PCI Express slots (depending on the form factor). • <u>Other built-in interfaces</u> Ideally, a motherboard should contain as many built-in standard controllers and interfaces as possible. Most boards feature integrated USB, sound, and LAN (look for those offering gigabit Ethernet), whereas others also have integrated video, FireWire, eSATA, dual LAN adapters, and more. <p><u>Documentation:</u> Good technical documentation is important.</p>

	<ul style="list-style-type: none"> • <u>Technical support:</u> Good online technical support goes beyond documentation
8.	<p>Write about USB.</p> <ul style="list-style-type: none"> • A Universal Serial Bus (USB) is a common interface that enables communication between devices and a host controller such as a personal computer (PC). • It connects peripheral devices such as digital cameras, mice, keyboards, printers, scanners, media devices, external hard drives and flash drives
9.	<p>What is POST and Bootstrap loader ? Explain .</p> <p>Alternatively called bootstrapping, bootloader, or boot program, a bootstrap loader is a program that resides in the computer's EPROM, ROM, or another non-volatile memory. It is automatically executed by the processor when turning on the computer. The bootstrap loader reads the hard drives boot sector to continue to load the computer's operating system. The term bootstrap comes from the old phrase "Pull yourself up by your bootstraps."</p> <p>When the computer is turned on or restarted, the bootstrap loader first performs the power-on self-test, also known as POST. If the POST is successful and no issues are found, the bootstrap loader loads the operating system for the computer into memory. The computer can then access, load, and run the operating system.</p>
10.	<p>Differentiate between memory bus and processor bus.</p> <p>A <u>processor bus</u>, also known as the front-side bus, refers to a specific electrical connection within a computer that connects a computer's processor to a chip known as the north bridge. For a computer to properly function, the processor, also known as the central processing unit (CPU), must send out orders and submit pieces of information to the computer's memory. The processor bus brings messages back and forth between the processor and the north bridge, which in turn sends messages to the computer's memory and other parts of the computer. This allows the computer to perform multiple tasks at once at fast speeds.</p> <p>The <u>memory bus</u> connects the memory system and the northbridge area of the chipset. This section of the chipset also connects directly to the central processing unit and the graphics system. While this means the northbridge is the center of many important computer functions, it is actually the computer's memory that determines the bus's speed. In essence, the speed of the computer's memory creates the speed of this bus, which determines the speed of the rest of the system.</p>
11.	<p>Define processor socket .</p> <p>Processor socket is also known as CPU socket.</p> <ul style="list-style-type: none"> • A CPU socket is a single connector between a microprocessor and motherboard. • A CPU socket is a distinct mount used only for the CPU on the motherboard to ensure

	<p>correct circuit chip insertion.</p> <ul style="list-style-type: none"> • A CPU socket also has a lock to prevent CPU movement, and its design helps secure heat sink placement above the CPU.
	Part B
12.	<p>Discuss the primary functions of the motherboard and list out their various form factors.</p> <p>Some of the prime functions of a computer motherboard are as follows:</p> <ul style="list-style-type: none"> - The motherboard acts as the central backbone of a computer on which other modular parts are installed such as the CPU, RAM and hard disks. - The motherboard also acts as the platform on which various expansion slots are available to install other devices / interfaces. - The motherboard is also responsible to distribute power to the various components of the computer. - They are also used in the coordination of the various devices in the computer and maintain an interface among them. <p>Some of the Sizes in which the motherboards are available are : BTX, ATX, mini-ATX, micro-ATX, mini-ITX etc.</p> <p>The shape and layout of the motherboard is called its form factor. The form factor acts as a standard for defining the motherboard. It determines the power supply, the type of case and the place where individual components will fit. The most popular motherboard form factor in domestic and industrial computers is ATX as it reduces the size of the motherboard greatly.</p> <p>The most common form factor is ATX, which evolved to mini-ATX, nano-ATX, pico-ATX, and further. A different type of form factor, and smaller than the ATX is the ITX, which is significantly smaller than the micro-ATX. The ITX form factor can also be found as nano-ITX, pico-ITX, mobile-ITX, and more.</p>
13.	<p>Write a note on system buses.</p> <p>The system bus is a pathway composed of cables and connectors used to carry data between a computer microprocessor and the main memory.</p> <p>The bus provides a communication path for the data and control signals moving between the major components of the computer system.</p> <p>The system bus works by combining the functions of the three main buses: namely,</p> <ol style="list-style-type: none"> 1. Data bus, 2. Address bus 3. control buses. Each of the three buses has its separate characteristics and responsibilities. <p>The system bus combines the functions of the three main buses, which are as follows:</p>

- The control bus carries the control, timing and coordination signals to manage the various functions across the system.
- The address bus is used to specify memory locations for the data being transferred.
- The data bus, which is a bidirectional path, carries the actual data between the processor, the memory and the peripherals.



A. The Internal Processor Bus: data, address, and control bus

- A bus is a pathway for digital signals to rapidly move data.
- There are three internal buses associated with processors: the data bus, address bus, and control bus. Together, these three make up the “system bus.”
- The system bus is an internal bus, intended to connect the processor with internal hardware devices, and is also called the “local” bus, Front Side Bus, or is sometimes loosely referred to as the “memory bus.”

B. Memory Bus

- The memory bus is a type of computer bus, usually in the form of a set of wires or conductors which connects electrical components and allow transfers of data and addresses from the main memory to the central processing unit (CPU) or a memory controller.
- A memory bus is made up of two parts: the data bus and the address bus.
- The data bus is responsible for the transfer of information between the memory and the chipset.
- The address bus communicates with the system on where specific information can be located or stored when data either enters or leaves the memory

C. Input/output bus

- The input/output bus or IO bus is the pathway used for input and output devices to communicate with the computer processor.
- I/O buses connect the CPU to all other components, except RAM.
- Data are moved on the buses from one component to another, and data from other components to the CPU and RAM.
- The I/O buses differ from the system bus in speed. Their speed will always be lower than the system bus speed.
- On modern PCs, you will usually find four buses:

The ISA bus, which is an old low speed bus, soon to be excluded from the PC design.

- An Industry Standard Architecture bus (ISA bus) is a computer bus that allows additional expansion cards to be connected to a computer's motherboard.
- It is a standard bus architecture for IBM compatibles

The PCI bus, which is a new high speed bus.

- Conventional PCI, often shortened to PCI, is a local computer bus for attaching hardware devices in a computer.
- PCI is the initialism for Peripheral Component Interconnect and is part of the PCI LocalBus standard.

The USB bus (Universal Serial Bus), which is a new low speed bus.

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

The AGP bus which solely is used for the graphics card.

- The Accelerated Graphics Port (AGP) was designed as a high-speed point-to-point channel for attaching a video card to a computer system, primarily to assist in the acceleration of 3D computer graphics.
- It was originally designed as a successor to PCI-type connections for video cards.

14.	<p>Define co-processor and explain about math co-processor.</p> <ul style="list-style-type: none"> • A coprocessor is a computer processor used to supplement the functions of the primary processor(the CPU). • Operations performed by the coprocessor may be floating point arithmetic, graphics, signal processing, string processing, cryptography or I/O interfacing with peripheral devices. <p>A math coprocessor is a computer chip that handles the floating point operations and mathematical computations in a computer. In early PCs, this chip was separate and often optional, and it was primarily used in computers where Computer Aided Design (CAD) was the primary focus. In today's computers, it is generally built into the CPU, allowing the central processor to offload the mathematical computations to this chip. This helps the CPU maintain more processes at one time.</p>
15.	<p>Define motherboard and write the importance of motherboard in the system.</p> <p>Motherboard is a main (PCB) printed circuit board. Where all the components or devices are connected directly and indirectly. The motherboard is a computer's central communications backbone connectivity point, through which all components and external peripherals connect.</p> <p>A motherboard controls the flow of external data through the ports such as the USB and the IDE and guides the data to the hard drives. The main function of the motherboard is to hold the microprocessor chip e.g. Intel Pentium 4, Intel Celeron and AMD Athlon. Also, almost all the components that help in the function of the computer are either parts of the motherboard or are plugged into it.</p>
	Part C
16.	<p>Discuss the various components of a motherboard with the help of a diagram.</p> <p>The computer motherboard connects all the parts(components) of a computer together. Mainboard, baseboard, mobo (abbreviation), system board, MB (abbreviation), logic board are the synonyms of computer's motherboard . The motherboard is the most important component in the PC. All the component such as RAM stick, hard disk drive, optical drives, processor, processor fan and external card are plugin into motherboard.</p> <p>Computer motherboard is single platform to connect all of the parts (components) of a computer together,Hence it considered as the backbone of a computer.</p> <p><u>I I Components of motherboard</u></p> <p>A. Expansion slots</p>

- B. Processor socket
- C. Coprocessor
- D. Memory modules
- E. BIOS and CMOS
- F. Chipset
- G. Super I/O chip
- H. ROM BIOS

A. Expansion Slot

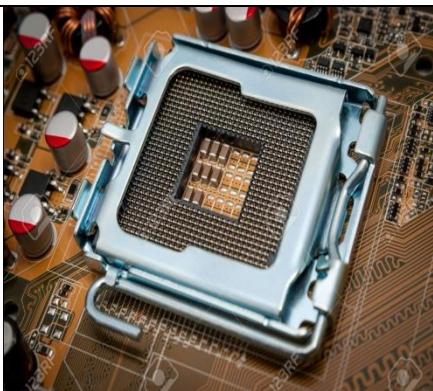
- An expansion slot is a socket on the motherboard that is used to insert an expansion card (or circuit board), which provides additional features to a computer such as video, sound, advanced graphics, Ethernet or memory.

Example.

- Sound card or Multimedia
- LAN card.
- SCSI controller card.
- Internal Modem card.
- TV tuner card.
- Additional hard disc controller card

B. Central Processing Unit (CPU) Socket (CPU Socket)

- A CPU socket is a single connector between a microprocessor and motherboard.
- A CPU socket is a distinct mount used only for the CPU on the motherboard to ensure correct circuit chip insertion.
- A CPU socket also has a lock to prevent CPU movement, and its design helps secure heat sink placement above the CPU.

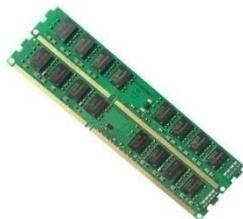


C. Coprocessor

- A coprocessor is a computer processor used to supplement the functions of the primary processor(the CPU).
- Operations performed by the coprocessor may be floating point arithmetic, graphics, signal processing, string processing, cryptography or I/O interfacing with peripheral devices.

D. memory module

- A memory module is a circuit board that contains DRAM integrated circuits that are installed into the memory slot on a computer motherboard.
- Below is an image of a 512 MB DIMM computer memory module and the most common type of memory used today.



E.BIOS AND CMOS

- The BIOS (Basic Input/Output System) is firmware stored in a chip on your computer's motherboard. It is the first program that runs when you turn on your computer.
- The BIOS performs the POST, which initializes and tests your computer's hardware. Then it locates and runs your boot loader, or loads your operating system directly.

CMOS

- When you make changes to your BIOS configuration, the settings are not stored on the BIOS chip itself. Instead, they are stored on a special memory chip, which is referred to as "the CMOS."
- CMOS stands for "Complementary Metal-Oxide-Semiconductor."
- Like most RAM chips, the chip that stores your BIOS settings is manufactured using the CMOS process. It holds a small amount of data, usually 256 bytes.
- The information on the CMOS chip includes what types of disk drives are installed on your computer, the current date and time of your system clock, and your computer's boot sequence.

CMOS battery



- The CMOS battery is a lithium-ion battery about the size of a coin.
- It can hold a charge for up to ten years before needing to be replaced.
- If your CMOS battery dies, your BIOS settings will reset to their defaults when your computer is turned off.

F.CHIPSET

- A chipset is a group of interdependent motherboardchips or integrated circuits that control the flow of data and instructions between the central processing unit (CPU) or microprocessor and external devices.
- Achipset controls external buses, memory cache and some peripherals.

	<p>Type of chipset</p> <p>The term chipset often refers to a specific pair of chips on the motherboard:</p> <ul style="list-style-type: none"> • The northbridge and the southbridge. • The northbridge links the CPU to very high-speed devices, especially RAM and graphics controllers, and the • southbridge connects to lower-speed peripheral buses (such as PCI or ISA). In many modern chipsets, the southbridge contains some on-chip integrated peripherals, such as Ethernet, USB, and audio devices. <p>G.super I/O chip</p> <ul style="list-style-type: none"> • Super i/o chip is a single chip which controls Slower i/o devices which are not controlled by South bridge. <p>H.ROM BIOS</p> <ul style="list-style-type: none"> • All motherboards include a small block of Read Only Memory (ROM) which is separate from the main system memory used for loading and running software. • The BIOS will most likely be stored in a 32-pin chip, which can typically be identified by a silver or gold sticker that shows the name of the BIOS company. • The BIOS comprises several separate routines, serving different functions. The first part runs as soon as the machine is powered on. It inspects the computer to determine what hardware is fitted and then conducts some simple tests to check that everything is functioning normally
--	--

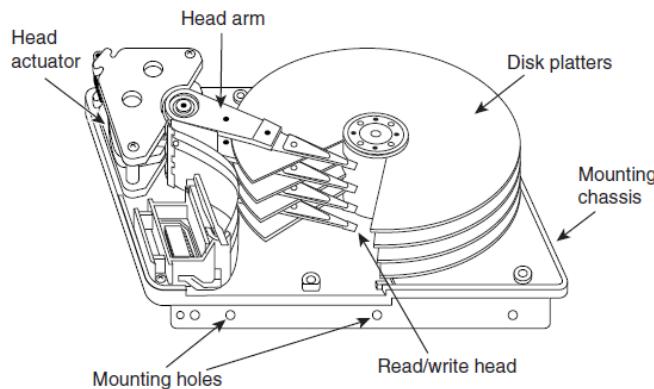
Module 4

	Part A
1.	<p>Write any two features each of FAT32 and NTFS File System.</p> <p><u>Differences between the NTFS and FAT32 memory card file systems</u></p> <p>Most commercially available memory cards use the FAT32 file system by default. NTFS and FAT32 file systems differ in the following ways:</p> <ul style="list-style-type: none">• The FAT32 file system cannot store individual files larger than 4 GB, while the NTFS file system can.• Compared to FAT32, the NTFS file system has higher disk utilization and can manage disk space more effectively.• The read and write speeds of NTFS are also faster than that of FAT32.• The FAT32 file system is suitable for smaller capacity flash memory devices (such as memory cards, USB drives, and other storage devices), while the NTFS file system is suitable for large capacity disks. <p>FAT32 has the following features that overcome the issues in previous versions of File systems:</p> <ul style="list-style-type: none">○ It can support drive sizes up-to 2 terabytes.○ It is space-efficient because it uses smaller clusters, which provides efficient use of available disk space compared to FAT16 devices.○ It does not contain any limit for the number of root folder entries on the drive. The root folder is a simple cluster chain that can be placed anywhere within the drive.○ The dynamic resizing of FAT32 partitions is possible.○ It contains a robust nature because, in this, the folder can be relocated, and it can use the backup copy of the FAT instead of the default copy. <p>NTFS supports:</p> <ul style="list-style-type: none">• Very large files• Different file permissions and encryption• Automatically restores consistency by using log file and checkpoint information• File compression when running out of disk space

	<ul style="list-style-type: none"> Establishing disk quotas, limiting space users can use
2.	<p>What is USB ?</p> <ul style="list-style-type: none"> A Universal Serial Bus (USB) is a common interface that enables communication between devices and a host controller such as a personal computer (PC). It connects peripheral devices such as digital cameras, mice, keyboards, printers, scanners, media devices, external hard drives and flash drives Short for universal serial bus, is a plug and play interface that allows a computer to communicate with peripheral and other devices. USB-connected devices cover a broad range; anything from keyboards and mice to music players and flash drives. For more information on these devices, see our USB devices section.
3.	<p>Write briefly about high level formatting .</p> <ul style="list-style-type: none"> High-level formatting is not really a physical formatting of the drive, but rather the creation of a table of contents for the disk. In low-level formatting, which is the real physical formatting process, tracks and sectors are written on the disk. <p>High-level formatting is the process of writing. Writing on a file system, cluster size, partition label, and so on for a newly created partition or volume. It is done to erase the hard-disk and again installing the operating system on the disk-drive.</p> <pre> graph TD A[High-level Formatting] --> B[Clear data on hard-disk] B --> C[Generate boot information] C --> D[Initialize FAT] D --> E[Label logical bad sectors] </pre> <p>Figure – Steps of High-level Formatting</p> <p>Firstly High-level formatting clears the data on hard-disk, then it will generate boot information, then it will initialise FAT after this it will go for label logical bad sectors when partition has existed.</p> <p>Formatting done by the user is the high-level formatting.</p> <p>Generally, It does not harm the hard-disk. It can be done easily with the Administrator, Windows snap-in Disk Management tool, diskpart, etc.</p> <p>We can use such a format to fix some problems like errors in the file system, corrupted hard-drive</p>

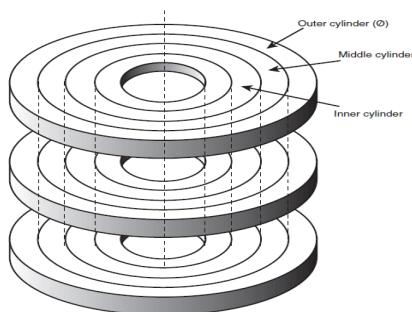
	and develop bad sectors.
4.	<p>What is FAT?</p> <p>The FAT File System was initially used on PC operating systems such as MS-DOS and early versions of Microsoft Windows. Still, it is widely used on USB memory sticks or memory cards for file storage. It is simple, robust and offers good performance especially in embedded applications.</p> <p>FAT is an acronym for File Allocation Table. This table provides the index of the files in the system and is statically allocated at the time of formatting the drive. It contains an entry for each cluster (a data storage area). The FAT's entries contain either the number of the next cluster in the file, or a marker indicating EOF (end of file), unused disk space, or other areas of the drive that are specially reserved. The drive's root directory contains the number of the first cluster of each file in the directory; the operating system traverses the FAT table, looking up the cluster number of each successive part of the file, until the end of the file is reached. Likewise, sub-directories are implemented as special files containing the directory entries of their respective files.</p>
5.	<p>What is NTFS?</p> <p>NTFS file system, known as New Technology File System (NTFS), is a proprietary journaling file system developed by Microsoft. Starting with Windows NT 3.1, it is the default file system of the Windows NT family. It superseded File Allocation Table (FAT) as the preferred filesystem on Windows and is supported in Linux and BSD as well.</p> <p>NTFS file system consists of 5 components, including O.S Boot Record, MFT 1, MFT Metadata, MFT2, and Data Area.</p> <p>See how each part works:</p> <ul style="list-style-type: none"> • Partition Boot Sector: Also known as PBS, it holds the boot information. • Master File Table: Short for MFT, is a directory that stores metafile data (or metadata), including file name, creation date, access permissions, size, etc. • Metafiles: It helps to define and organize the file system, backup critical file system data, buffer file system changes, manage free space allocation, and helps structure metadata more efficiently. • Data Area: The unit that stores files and data in a partition or external drive.
6.	<p>Define HDD.</p> <p>A hard disk drive, hard disk, hard drive, or fixed disk, is an electro-mechanical data storage device that stores and retrieves digital data using magnetic storage with one or more rigid rapidly rotating platters coated with magnetic material.</p> <p>A computer hard drive (or a hard disk or HDD) is one kind of technology that stores the operating system, applications, and data files such as documents, pictures and music that your computer uses.</p> <ul style="list-style-type: none"> • A hard disk drive is a sealed unit that a PC uses for nonvolatile data storage.

- Nonvolatile, or semi-permanent, storage means that the storage device retains the data even when no power is supplied to the computer.
- A hard disk drive contains rigid, disk-shaped platters, usually constructed of aluminum or glass (see Figure 10.1).
- In most hard disk drives, you cannot remove the platters, which is why they are sometimes called fixed disk drives.



Hard disk heads and platters.

7. What is meant by a cylinder in HDD ?



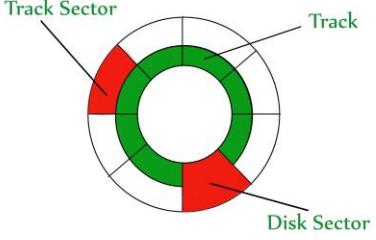
Cylinder is a cylindrical intersection through the stack of platters in a disk, centered around the disk's spindle. Combined, cylinder and head intersect to a circular line, or more precisely: a circular strip of physical data blocks called track.

The cylinder is the aggregate of the same track number on every platter used for recording.

The aggregate of all tracks that reside in the same location on every disk surface. On multiple-platter disks, the cylinder is the sum total of every track with the same track number on every surface.

Cylinder The set of all the TRACKS on a HARD DISK drive with multiple PLATTERS that may be read at the same time. All the tracks are the same distance from the central spindle, so they can be imagined as tracing a cylinder in space. The HEADS on all the platters move together in a parallel motion – a sequence of data stored within the same cylinder can be read at optimum speed without requiring any movement.

Part B

8.	<p>Write briefly about tracks, sectors and cylinders in hard disk.</p> <p>A disk is divided into tracks, cylinders, and sectors. A track is that portion of a disk which passes under a single stationary head during a disk rotation, a ring 1 bit wide. A cylinder is comprised of the set of tracks described by all the heads (on separate platters) at a single seek position. Each cylinder is equidistant from the center of the disk. A track is divided into segments of sectors, which is the basic unit of storage.</p>  <p>The disk is divided into tracks. Each track is further divided into sectors. Each disk consists of platters, rings on each side of each platter called tracks, and sections within each track called sectors. A sector is the smallest physical storage unit on a disk, almost always 512 or 4096 bytes in size.</p> <p>On hard disks, the data is stored on the disk in thin concentric bands called tracks. There can be more than a thousand tracks on a 3½ inch hard disk.</p> <p>Tracks are a logical rather than physical structure, and are established when the disk is low-level formatted. Track numbers start at 0, and track 0 is the outermost track of the disk. The highest numbered track is next to the spindle.</p> <p>A cylinder consists of a set of tracks that start at the same head position on the disk.</p>
9.	<p>Explain different read/write heads.</p> <ul style="list-style-type: none"> ❖ A hard disk drive usually has one read/write head for each platter surface (meaning that each platter has two sets of read/write heads—one for the top side and one for the bottom side). ❖ These heads are connected, or ganged, on a single movement mechanism. ❖ The heads, therefore, move across the platters in unison. ❖ Each head is on an actuator arm that is spring-loaded to force the head into contact with a platter.

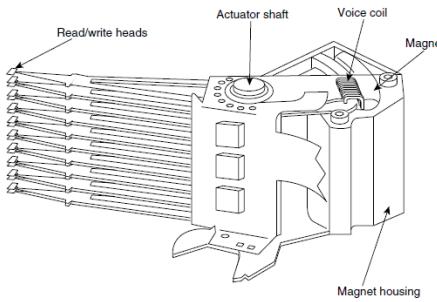


Figure 10.7 Read/write heads and rotary voice coil actuator assembly.

A read/write head is a specific physical part of a hard disk that is responsible for reading data from, and writing data to, the disk. Read/write heads are typically made up of a thin horizontal magnetic blade attached to an actuator arm. By changing the electrical polarity of bits on a magnetic disk, the read/write arm effectively records data to a disk drive.

A component of a disk drive that records and retrieves data from magnetic disks. Read/write heads are also used sometimes to record and retrieve data from magnetic tapes. In the case of disk drives the assembly consists of a head, sometimes known as a slider, and a mounting arm, known as a flexure.

10. Describe the HD features.

To make the best decision in purchasing a hard disk for your system or to understand what distinguishes

one brand of hard disk from another, you must consider many features. This section examines

some of the issues you should consider when you evaluate drives:

- _ Reliability
- _ Performance
- _ Cost

Reliability

When you shop for a drive, you might notice a statistic called the Mean Time Between Failures

(MTBF) described in the drive specifications. MTBF figures usually range from 300,000 to 1,000,000

hours or more. I usually ignore these figures because they are derived theoretically.

S.M.A.R.T.

S.M.A.R.T. (Self-Monitoring, Analysis, and Reporting Technology) is an industry standard providing

failure prediction for disk drives. When S.M.A.R.T. is enabled for a given drive, the drive monitors predetermined attributes that are susceptible to or indicative of drive degradation. Based on changes in the monitored attributes, a failure prediction can be made. If a failure is deemed likely to occur, S.M.A.R.T. makes a status report available so the system BIOS or driver software can notify the user of the impending problems, perhaps enabling the user to back up the data on the drive before any real problems occur.

Performance

When you select a hard disk drive, one of the important features you should consider is the performance (speed) of the drive. Hard drives can have a wide range of performance capabilities. As is true of many things, one of the best indicators of a drive's relative performance is its price. An old saying from the automobile-racing industry is appropriate here: "Speed costs money. How fast do you want to go?"

You can measure the speed of a disk drive in two ways:

- _ Average seek time
- _ Transfer rate

Average seek time, normally measured in milliseconds (ms), is the average amount of time it takes to

move the heads from one cylinder to another a random distance away. The standard method used by many drive manufacturers to measure the average seek time involves measuring the time it takes the heads to move across one-third of the total cylinders. Average seek

time depends only on the drive itself; the type of interface or controller has little effect on this specification. The rating is a gauge of the capabilities of the head actuator.

Average Access Time

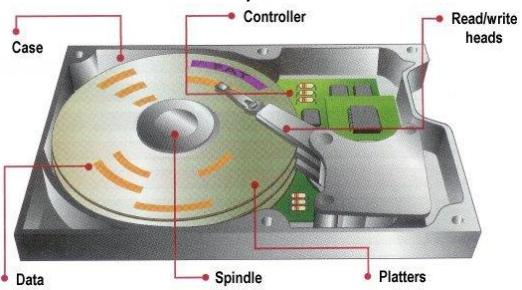
A slightly different measurement, called average access time, involves another element called latency. Latency is the average time (in milliseconds) it takes for a sector to be available after the heads have reached a track. On average, this figure is half the time it takes for the disk to rotate once. A drive that spins twice as fast would have half the latency. A measurement of a drive's average access time is the sum of its average seek time and latency. This number provides the average amount of time required before the drive can access a randomly requested sector.

	<p>Latency</p> <p>Latency is a factor in disk read and write performance. Decreasing the latency increases the speed of access to data or files and is accomplished only by spinning the drive platters more quickly</p> <p>Transfer Rate</p> <p>The media transfer rate is far more important than the interface transfer rate because the media transfer rate is the true rate at which data can be read from the disk, which is how fast data can be read from the drive platters (media). It is the maximum rate that any sustained transfer can hope to achieve.</p> <p>Cost</p> <p>The cost of hard disk storage is continually falling. You can now purchase a 30GB ATA drive for around \$150, which is about half a cent per megabyte. A drive I bought in 1983 had a maximum capacity of 10MB and cost \$1,800. At current pricing (0.5 cents per megabyte), that drive is worth about 5 cents! Of course, the cost of drives continues to fall, and eventually, even half a cent per megabyte will seem expensive. Because of the low costs of disk storage today, not many 3-1/2 inch drives with capacities of less than 10GB are even being manufactured.</p> <p>Capacity</p> <p>Four figures are commonly used in advertising drive capacity:</p> <ul style="list-style-type: none"> _ Unformatted capacity, in millions of bytes _ Formatted capacity, in millions of bytes _ Unformatted capacity, in megabytes _ Formatted capacity, in megabytes
11.	<p>Differentiate between ISA and EISA expansion buses .</p> <p>Extended Industry Standard Architecture (EISA) is a 32 bit modification to ISA bus. As computers became larger and had wider data buses, a new bus was required that would transfer 32-bit data. Clocking speed limited up to 8MHz. The most common application for EISA bus is a disk controller or as a video graphics adapter. These applications benefit from wider data bus width since the data transfer rate for these devices are high.</p>

	<p>Extended Industry Standard Architecture (EISA) is a bus architecture that extends the Industry Standard Architecture (ISA) from 16 bits to 32 bits. EISA was introduced in 1988 by the Gang of Nine - a group of PC manufacturers. EISA is also known as Extended ISA.</p> <p>INDUSTRY STANDARD ARCHITECTURE (ISA): Industry Standard Architecture (ISA) is a computer bus standard for IBM PC compatible computers introduced with the IBM Personal Computer to support its Intel 8088 microprocessor's 8-bit external data bus and extended to 16 bits for the IBM Personal Computer/AT's Intel 80286 processor. The ISA bus was further extended for use with 32-bit processors as Extended Industry Standard Architecture (EISA). For general desktop computer use it has been supplanted by later buses such as IBM Micro Channel, VESA Local Bus, Peripheral Component Interconnect and other successors.</p> <p>EXTENDED INDUSTRY STANDARD ARCHITECTURE (EISA): The Extended Industry Standard Architecture in practice almost always shortened to EISA is a bus standard for IBM PC compatible computers. It was announced in September 1988 by a consortium of PC clone vendors (the "Gang of Nine") as a counter to IBM's use of its proprietary Micro Channel architecture (MCA) in its PS/2 series.</p>
12.	<p>What is CMOS ? Write the features of CMOS.</p> <ul style="list-style-type: none"> • When you make changes to your BIOS configuration, the settings are not stored on the BIOS chip itself. Instead, they are stored on a special memory chip, which is referred to as "the CMOS." • CMOS stands for "Complementary Metal-Oxide-Semiconductor." • Like most RAM chips, the chip that stores your BIOS settings is manufactured using the CMOS process. It holds a small amount of data, usually 256 bytes. • The information on the CMOS chip includes what types of disk drives are installed on your computer, the current date and time of your system clock, and your computer's boot sequence. <p>CMOS battery</p> 

- The CMOS battery is a lithium-ion battery about the size of a coin.
- It can hold a charge for up to ten years before needing to be replaced.
- If your CMOS battery dies, your BIOS settings will reset to their defaults when your computer is turned off.

Part C

13.	<p>Explain the Hard disk operations and Hard disk drive installation procedure in detail.</p> <ul style="list-style-type: none"> • The hard drive contains a spinning platter with a thin magnetic coating • A "head" moves over the platter, writing 0's and 1's as tiny areas of magnetic North or South on the platter • To read the data back, the head goes to the same spot, notices the North and South spots flying by, and so deduces the stored 0's and 1's • A Modern hard drive can store well over a trillion 0/1 bits per platter, so the individual North/South spots are quite small • "Flash" storage is made with chips (no moving parts) and is gradually replacing spinning hard drives like this. Flash chips are what's inside camera SDHC memory cards and USB storage keys. <p>The disc platters are mounted on a single spindle that spins at a typical 10,000rpm. On EIDE and SCSI drives the disk controller is part of the drive itself. It controls the drive's servo-motors and translates the fluctuating voltages from the head into digital data for the CPU.</p> <p>Data is recorded onto the magnetic surface of a platter in exactly the same way as it is on floppies or digital tapes. Essentially, the surface is treated as an array of dot positions, with each domain' of magnetic polarisation being set to a binary 1 or 0. The position of each array element is not identifiable in an absolute sense, and so a scheme of guidance marks helps the read/write head find positions on the disk. The need for these guidance markings explains why disks must be formatted before they can be used.</p>  <p>When it comes to accessing data already stored, the disk spins round very fast so that any part of its circumference can be quickly identified. The drive translates a read request from the computer into reality. There was a time when the cylinder/head/sector location that the computer worked out really was the data's location, but today's drives are more complicated than the BIOS can</p>

handle, and they translate BIOS requests by using their own mapping.

The basic steps required to install a hard drive are:

1. Configure the drive as a master or slave device (PATA only).
2. Mount the drive in the chassis.
3. Connect the data cable to the drive and to the PATA or SATA interface.
4. Connect a power cable to the drive. Before you remove the case panels to install the hard drive:
5. Restart the system and run BIOS Setup. Note the current configuration which ATA and SATA ports are in use and the descriptions of the devices that are connected to them. Alternatively, use a diagnostic program such as Everest Home Edition to determine the current configuration of your drives and interfaces.
6. If you are also installing a PATA or SATA interface card or RAID adapter, configure that card per the maker's instructions and attach the cables to it. If that card will replace some or all of the embedded PATA or SATA interfaces, use CMOS Setup to disable those interfaces.

14. Explain the hard disk operations

- The basic physical construction of a hard disk drive consists of spinning disks with heads that move over the disks and store data in tracks and sectors.
- The heads read and write data in concentric rings called tracks, which are divided into segments called sectors, which normally store 512 bytes each

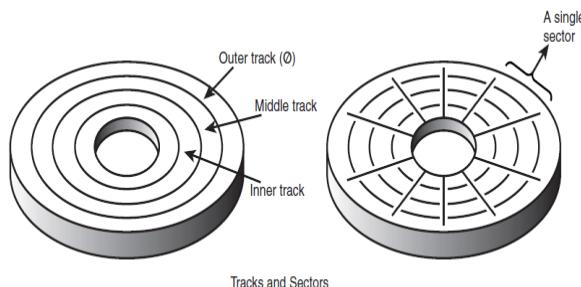
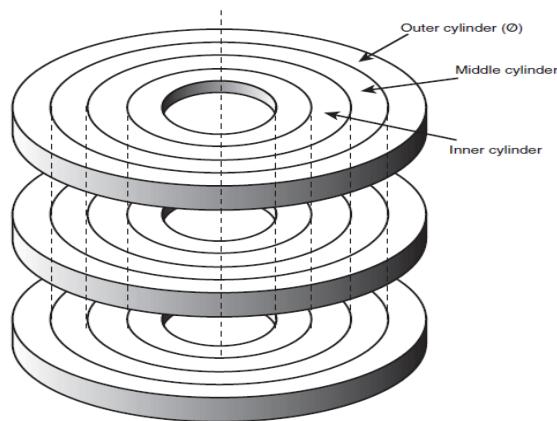


Figure 10.2 The tracks and sectors on a disk.

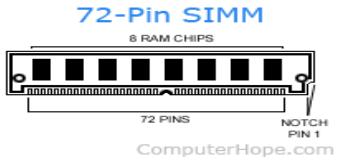
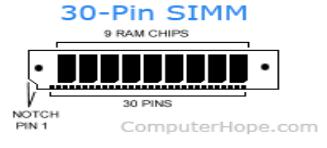
- Hard disk drives usually have multiple disks, called platters, that are stacked on top of each other and spin in unison, each with two sides on which the drive stores data.
- The identically aligned tracks on each side of every platter together make up a cylinder.
- A hard disk drive normally has one head per platter side, with all the heads mounted on a common carrier device or rack.
- The heads move radially across the disk in unison; they cannot move independently because they are mounted on the same carrier or rack, called an actuator.
- The heads in most hard disk drives do not (and should not!) touch the platters during normal operation.
- While the drive is running, a very thin cushion of air keeps each head suspended a short distance above or below the platter. If the air cushion is disturbed by a particle of dust or a shock, the head can come into contact with the platter while it is spinning at full speed.
- Most drives have special lubricants on the platters and hardened surfaces that can withstand the daily “takeoffs and landings” as well as more severe abuse.



15 Write note on (a) HD features (b) HD installation procedure
Refer qno:13

Module 5

	Part A
1.	<p>List the features of SIMMs.</p> <p>A SIMM (single in-line memory module) is a module containing one or several random access memory (RAM) chips on a small circuit board with pins that connect to the computer motherboard.</p> <p>DIMM and SIMM are two major types of random-access memory standards for personal computers. DIMM is an acronym for “dual in-line memory module,” while SIMM stands for “single in-line memory module.”</p> <p>The edge connector of this expansion board is plugged into a special SIMM sockets on the motherboard. This design allows the memory to be added and removed from the computer without the risk of destroying it.</p> <p>SIMMs are available in SRAM, DRAM, VRAM etc.</p> <p>Different type of SIMM are</p> <ol style="list-style-type: none">1. 30-pin SIMM2. 72 pin SIMM <p>30-pin SIMM: It is the first SIMM introduced with the PC system.</p> <p>The 30-pin SIMM measures about 3.5 inches wide and an inch tall.</p> <p>30-pin SIMMs use one byte-wide data buses</p> <p>72- pin SIMM: a new SIMM with more pins to accommodate wider data buses was required . This lead IBM to introduce 72-pin SIMMs</p> <p>Its 72-pin could pack four byte-wide banks on a single module</p> <p>The notch in the center of the SIMM edge connector prevents one from accidentally sliding a 30-pin SIMM into a 72-pin socket</p> <p>To achieve higher capacities, 72-pin SIMMs are often double-sided.</p>

	 
2.	<p>Write the advantages of RIMM over other types of memory modules</p> <p>In a computer, a RIMM is a memory module developed by Kingston Technology Corp. that takes up less space inside the computer than the older DIMM module and has different pin characteristics.</p> <p>A RIMM has a 184-pin connector</p> <p>RIMMs also have different signal pins on each side. Three different physical types of RIMMs are available: a 16/18-bit version with 184 pins, a 32/36-bit version with 232 pins, and a 64/72-bit version with 326 pins. Each of these plugs into the same sized connector, but the notches in the connectors and RIMMs are different to prevent a mismatch. A given board will accept only one type. By far the most common type is the 16/18-bit version. The 32-bit version was introduced in late 2002, and the 64-bit version was introduced in 2004.</p> <p>The standard 16/18-bit RIMM has 184 pins, one notch on either side, and two notches centrally located in the contact area. The 16-bit versions are used for non-ECC applications, whereas the 18-bit versions incorporate the additional bits necessary for ECC.</p>
3.	<p>What are the advantages of VFAT over FAT.</p> <p>A VFAT is primarily an upgrade to the file allocation table (FAT) system, and is installed as a driver on a host computer. After installation, VFAT runs in a 32-bit protected mode VCACHE cache. Unlike FAT, which restricts file names to having no more than eight characters, VFAT expanded that range to accommodate up to 255 characters. VFAT is also supported by other operating systems and is installed as a driver extension for all of them.</p>
4.	<p>Differentiate UMA and HMA</p> <p><u>Upper Memory Area(UMA):</u></p> <ul style="list-style-type: none"> •The upper 384 KB of the first megabyte of system memory just above

	<p>the Conventional Memory is called as upper memory area.</p> <ul style="list-style-type: none"> • Function of UMA is ROM shadowing and loading drivers. <p><u>High Memory Area(HMA):</u></p> <ul style="list-style-type: none"> • The first 64 KB of extended memory are termed as HMA. • It can accessed also when the processor is in real mode. • It is usually used for DOS. • It occupies addresses 100000h to 10FFEFh. • The HMA starts at the 1 megabyte limit.i.e the location 1024KB and goes up to 1088KB of memory • BZ it is not contiguous with the address range of lower memory,it cannot be used as extra memory by ordinary DOS applications. <p>Short for Upper Memory Area, UMA is the area of RAM between 640 KB and 1,024 KB (1 MB) in legacy computers, that is made available to user applications as RAM. In DOS based systems, memory is split into five areas: conventional memory; upper memory; high memory; extended memory; and expanded memory. The UMA consists of upper memory blocks, and users can access this area through a special memory manager, such as EMM386.exe.</p> <p>Short for high memory area, HMA is the first 64 KB (65,520 bytes) of extended memory above the first 1 MB of memory found on IBM compatible computers. The HMA is used to swap programs from conventional memory into high memory making the computer believe more conventional memory is available on the computer.</p> <p>In MS-DOS, the HMA can be accessed using Microsoft's HIMEM.SYS file and specifying the DOS=HIGH command in the config.sys. For additional information about these files, see our Autoexec.bat and Config.sys guide.</p> <p>Early computers had issues with HMA and the keyboard controller that caused A20 errors.</p>
	Part B
5.	<p>Write a note on expanded memory.</p> <ul style="list-style-type: none"> • <u>Expanded Memory</u> Specification (EMS), or expanded memory, is a technique for utilizing more than 1MB of main memory in DOS -based computers.

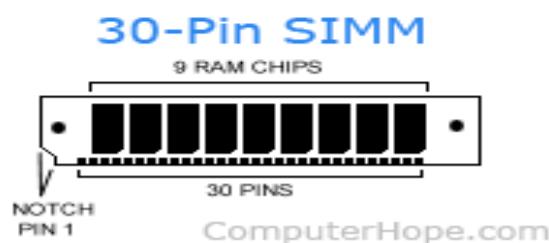
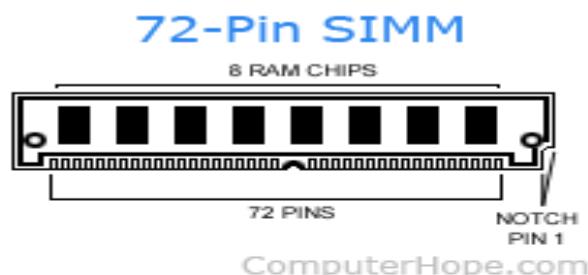
	<ul style="list-style-type: none"> The limit of 1MB is built into the DOS operating system. The upper 384K is reserved for special purposes, leaving just 640K of conventional memory for programs. The EMS is not part of the main memory it is a separate memory installed into the system which can be accessed in a fixed sized pages using a method called “bank switching”. <p>Expanded memory (EM) is an overarching or umbrella term for several technology variants that do not necessarily work with each other or are directly related to each other. However, these technologies were meant to solve the same problem, the 640 KB limit on usable memory for programs in the DOS operating system. The most widely used expanded memory variant was the Expanded Memory Specification (EMS) or the LIM EMS.</p>
6.	<p>Write note on extended memory .</p> <p>The memory above 1 MB area is called extended memory as shown in fig.</p> <ul style="list-style-type: none"> It is used for programs and data when using an operating system running in protected mode. This memory is available from addresses 10FF0H to the last memory location. <p>Extended memory specification(XMS)</p> <ul style="list-style-type: none"> This specification works with all the processors that can address extended memory This specification also allows the real mode dos programs to use the extended memory, and to use a special area in the extended memory called High memory area(HMA) <p>VIRTUAL CONTROL PROGRAM INTERFACE(vcpi)</p> <ul style="list-style-type: none"> This specification was mainly developed to make the DOS programs in the virtual 86 mode to work without any conflicts. <p><u>Upper Memory Area(UMA):</u></p> <ul style="list-style-type: none"> The upper 384 KB of the first megabyte of system memory just above the Conventional Memory is called as upper memory area. Function of UMA is ROM shadowing and loading drivers. <p><u>High Memory Area(HMA):</u></p> <ul style="list-style-type: none"> The first 64 KB of extended memory are termed as HMA. It can accessed also when the processor is in real mode. It is usually used for DOS.

	<ul style="list-style-type: none"> • It occupies addresses 100000h to 10FFEFh. • The HMA starts at the 1 megabyte limit.i.e the location 1024KB and goes up to 1088KB of memory • BZ it is not contiguous with the address range of lower memory,it cannot be used as extra memory by ordinary DOS applications.
7.	<p>Describe FAT32.</p> <p>FAT32 is a disk format or filing system used to organise the files stored on a disk drive. The disk drive is marked up into addressable chunks called sectors and a “File Allocation Table” or FAT is created at the start of the drive so that each piece of information in the file can be found by the host computer.</p> <p>FAT32 is a version of the File Allocation Table (FAT) file system that was introduced by Microsoft in 1996 with its Windows 95 OEM Service Releases 2 (OSR2) operating system. It is an extension of Microsoft’s FAT16 file system.</p> <p>The purpose of FAT32 was to overcome the limitations of FAT16 and add support for larger media. The major enhancements introduced by FAT32 included support for much larger volumes, better performance and more flexibility and robustness.</p>
8.	<p>What is meant by logical memory and physical memory ?</p> <p>Logical memory :</p> <ul style="list-style-type: none"> ❖ It is a virtual address generate in CPU ❖ Set of all logical addresses generated in CPU in reference to a program is referred as Logical space ❖ User can view a logical address of a program ❖ User uses the logical address to access the physical address <p>Physical memory :</p> <ul style="list-style-type: none"> ❖ It is a location in a memory unit ❖ User can never view physical address of a program ❖ User cannot directly access physical address
	Part C
9.	<p>Discuss about each one (a) Conventional memory (b) UMA (c) HMA</p> <p>Conventional memory</p>

	<p>Random Access Memory (RAM) was utilized quite differently on IBM® PC machines and clones in days of old when DOS, a command line operating system used prior to Windows®, ruled. The first 0-640 Kilobytes (KB) of RAM were designated as conventional memory; the area where DOS conventionally loaded upon booting. The 640KB – 1 Megabyte (MB) block consisting of 384KB was the Upper Memory Area (UMA); the first 64KB over the 1MB boundary became the High Memory Area (HMA or HIMEM), and everything over that was designated as Extended Memory. Conventional memory was used as the read/write area for the operating system and for programs, making for a tight fit. DOS also loaded routines, system drivers and system parameters into this space. To help free up room, some residents of conventional memory were moved up the RAM ladder to the UMA and HMA.</p> <p>Refer Qno:6</p>
10.	<p>Explain the following memory modules. SIMM, DIMM, RIMM.</p> <p><u>SIMM</u></p> <ul style="list-style-type: none"> • A SIMM (single in-line memory module) is a module containing one or several random access memory (RAM) chips on a small circuit board with pins that connect to the computer motherboard. • DIMM and SIMM are two major types of random-access memory standards for personal computers. DIMM is an acronym for “dual in-line memory module,” while SIMM stands for “single in-line memory module.” • The edge connector of this expansion board is plugged into a special SIMMsockets on the motherboard. This design allows the memory to be added and removed from theComputer without the risk of destroying it. • SIMMs are available in SRAM, DRAM, VRAM etc. <p>Different type of SIMM are</p> <ol style="list-style-type: none"> 3. 30-pin SIMM 4. 72 pin SIMM <p>30-pin SIMM:IT is the first SIMM introduced with the PC system.</p> <ul style="list-style-type: none"> • The 30-pin SIMMmeasures about 3.5 inches wide and an inch tall. • 30-pin SIMMs use one byte-wide data buses

72-pin SIMM: a new SIMM with more pins to accommodate wider data buses was required .This lead IBM to introduce 72-pin SIMMs

- Its 72-pin could pack four byte-widebanks on a single module
- The notch in the center of the SIMM edge connector prevents one from accidentally sliding a 30-pin SIMM into a 72-pin socket
- To achieve higher capacities, 72-pin SIMMs are often double-sided.



DIMM

- DIMM (dual in-line memory module) is a type of computer memory that is natively 64 bits, enabling fast data transfer.
- DIMM is a module that contains one or several random access memory ([RAM](#)) chips on a small circuit board with pins that connect it to the computer [motherboard](#).
- The DIMM stores each data bit in a separate memory cell.
- DIMMs use a 64-bit data path, since processors used in personal computers have a 64-bit data width.
- DIMMs are typically used in desktop PCs, laptops, printers and other devices.
- Another evolution in DIMMs is the use of cooling fins or structures attached directly to the DIMM.

- The increase in chip density in typical 8 GB or 16 GB DIMMS, and the increase in clock speed, led to an increase in heat production.

Types of DIMM: The most common standard DIMMs, with a typical length of 5.5 inches and height of 1.18 inches, are:

- Unbuffered DIMMs (UDIMMs)** -- Used mainly on desktop and laptop computers. Although they run faster and cost less, UDIMMs aren't as stable as registered memory.
- Fully-buffered DIMMs (FB-DIMMs)** – Usually used as main memory in systems that require large capacities, such as servers and workstations
- Registered DIMMs (RDIMMs)** -- Also known as buffered memory, RDIMMs are often used in servers and other applications that require robustness and stability.

RIMM

- In a computer, a **RIMM** is a **memory module** developed by Kingston Technology Corp. that takes up less space inside the computer than the older **DIMM module** and has different pin characteristics.
- A **RIMM** has a 184-pin connector

S.NO	SIMM	DIMM
1.	In SIMM, Pins present in either facet are connected.	DIMM pins are freelance.
2.	SIMM supports 32 bit channel for data transferring.	DIMM supports 64 bit channel for data transferring.
3.	SIMM consumes 5 volts of power.	DIMM consumes 3.3 volts of power.
4.	SIMM provides the storage 4 MB to 64 MB.	DIMM provides the storage 32 MB to 1 GB.

	<p>5. The classic or most common pin configuration of the SIMM module is 72 pins.</p> <p>6. SIMMs are the older technology.</p>	<p>The foremost common pin configuration of the DIMM module is 168 pins.</p> <p>DIMMs are the replacement of the SIMMs.</p>	
	<p>7. SIMMs are installed in pairs at a time.</p> <p>8. SIMMs are used by 486 CPU as well as early Pentium computers.</p>	<p>DIMMs are installed one at a time.</p> <p>DIMMs are used by modern Pentium computers.</p>	
	<p>9. The length and width of SIMM are respectively 4.25 inches and 1 inch.</p> <p>10. There are single notches in SIMMs.</p>	<p>The length and width of DIMM are respectively 1.67 to 5.25 inches and 1 to 1.75 inches.</p> <p>There are two notches in DIMMs.</p>	