Chapter 2

Instruction Set and Programming of 8085

Probable marks : 43

Scope of the syllabus :-

- Addressing modes in 8085
- Programming model of 8085
- Instruction Set and
- Programming of 8085
- Study of instruction set :
- Data transfer, Arithmetic, Logic, Branching, Stack, I/O and machine control instructions.
- Assembly language programming based on instructions.

Q. 1 What do you mean by addressing modes of a microprocessor ? Enlist the addressing modes in 8085.

Ans. :

(1) Addressing mode of a microprocessor is the various formats of specifying one operands (directly, indirectly etc). The operand can be data (8 or 16 bit), address, register or it can be implicit.

(2) Every microprocessor has its own set of instructions. Each of these instructions uses one of the addressing modes.

(3) The microprocessor 8085 has five addressing modes, which are given below :

i) Direct addressing

ii) Register addressing

iii) Immediate addressing

iv) Register indirect addressing

v) Implied addressing

Q. 2 Explain any two addressing modes in 8085. OR
Explain all addressing modes in 8085. OR
Explain direct and implicit addressing modes of 8085 microprocessor. OR
(March 02, 18)

What are different addressing modes used in 8085 microprocessor ? Explain any two of them with a suitable example. **(Mar.02, 04, 05, 09, 2011 Oct. 2009, 2010; July 17)**

Ans. : (I) Direct addressing mode : **(July 2017, March 2018)**

(1) In direct addressing, the address appears after opcode of instruction in program memory.

(2) The address of operand is specified within the instruction.

(3) The instructions using direct addressing mode are three byte instructions. Byte 1 is opcode of instruction, Byte 2 is lower order address and Byte 3 is high order address.

(4) For e.g. LDA 9FFFH

i.e. This instruction loads accumulator with content of memory location 9FFF H.

(II) Register addressing mode : **(March 2004)**

(1) In register addressing mode, the source operanas are general purpose registers whose name is specified within the instruction.

(2) These instructions are single byte instructions.

(3) All actions occur within the CPU.

(4) For e.g. MOV A, B.

i.e. This instruction transfers the content of register B to accumulator without modifying the content of B.

(III) Immediate addressing mode : **(March 2018, July 2019)**

(1) In immediate addressing the data appears immediately after opcode of instruction in program memory.

(2) In these instructions the actual data is specified within the instruction.

(3) These operations are specified with either 2 or 3 byte instructions.

(4) For e.g. ADI 05H

i.e. this instruction adds immediate data 05 H to the content of accumulator. The result is stored in accumulator.

(IV) Register indirect addressing mode : **(Oct. 2005, 2009; July 2017)**

(1) In register indirect addressing the content of register pair points to the address of the operand.

(2) A register pair (H-L pair) is specified for addressing 16-bit address of memory location.

(3) These are generally 1-byte instruction.

(4) For e.g. ADD M

i.e. this instruction will add the content of memory location whose address is stored in H-L pair to the content of accumulator.

(V) Implicit addressing mode : **(Oct. 2005)**

(1) In this type of instructions, generally operand is not specified within the instruction and it is predetermined.

(2) Generally the operand is accumulator.

(3) Most of the logical group instructions belong to this addressing mode.

(4) These are single byte instructions.

(5) All actions occur within the CPU.

(6) For e.g. CMA

i.e. this instruction will complement the content of accumulator. Here, the actual operand is not specified in the instruction, but is predetermined (accumulator). The result is stored in accumulator.

Q. 3 Explain register indirect and immediate addressing modes in case of 8085 microprocessor with the help of suitable examples.

(Oct. 03,07,08; June 16; July 17, March 2020)

Ans. : I) Register indirect addressing mode :

1) Register indirect instructions reference memory using the contents of a register pair to point the address of the operand.

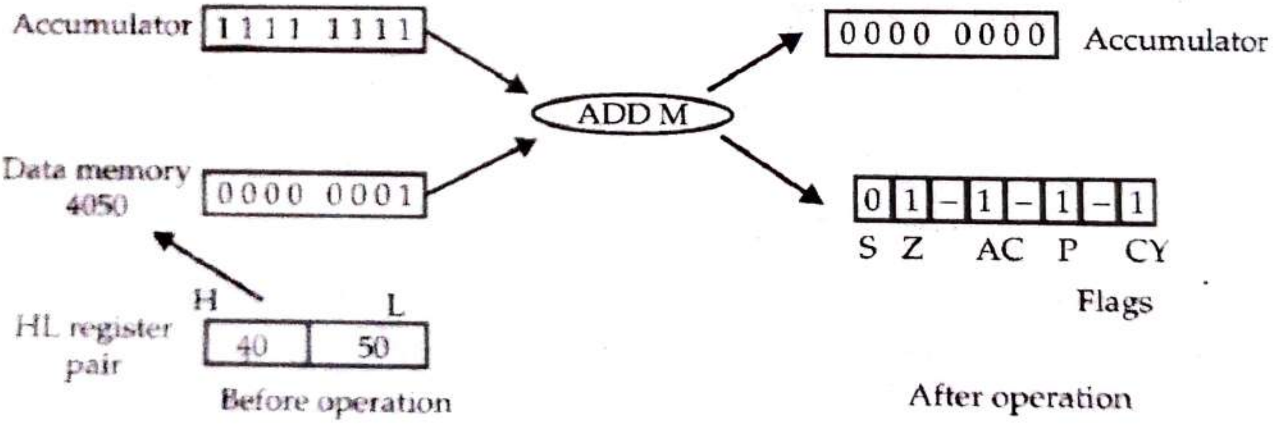
2) A register pair (H-L pair) is specified for addressing 16-bit address of memory location.

3) These are generally 1 byte instructions.

4) For example :

ADD M (add memory)

This instruction adds the contents of the accumulator to the contents of memory pointed to by the address in the HL register pair.



In the given example, the HL register pair points to memory location 4050 H. The data in this location $(0000\ 0001)_2$ is then added to the contents of the accumulator $(1111\ 1111)_2$.

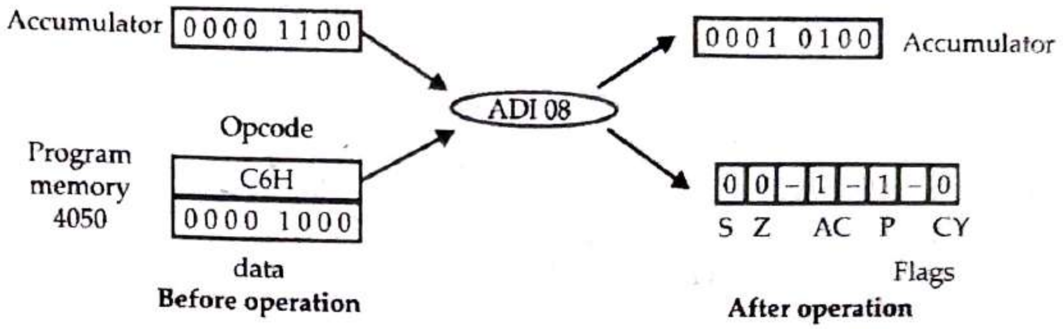
The sum is stored in accumulator and appropriate flags are also set and reset based on the result.

II) Immediate addressing mode : **(March 2009)**

1) In immediate addressing the data appears immediately after opcode of instruction in program memory.

2) These are generally 2 or 3 byte instructions.

3) **For example** **ADI 08 H**



In the example, MPU fetches the opcode (C6) from program memory and after decoding it. MPU finds the immediate data $(0000\ 1000)_2$ in the next consecutive program memory location. This data is added to the contents of the accumulator and result is placed in the accumulator.

Q. 4 What do you understand by register indirect and implicit addressing modes ? Explain with suitable example. List the names of any four instructions which make accumulator content clear. **(Oct. 2005)**

Ans. : Register indirect addressing : Please refer to Q. 3 (I).

Implicit addressing : Please refer to Q. 3 (II).

Following four instructions make accumulator content clear :

(1) ANI 00H (2) XRA A (3) SUB A (4) MVI A, 00H

Q. 5 What are the different ways of clearing Accumulator (A = 00H) in single instruction ? **(March 2011, 2017)**

Ans : (i) MVI A, 00H (ii) SUB A
(iii) XRA A (iv) ANI 00H

Q. 6 What are the groups in which instructions in 8085 are classified? **(Oct.1998)**

Ans. : The instructions in 8085 can be classified into following five groups, depending upon their function :

1) Data transfer group
2) Arithmetic group
3) Logical group
4) Branching group
5) Machine control group

1) Data Transfer Group :

This group of instruction copies data from a location called source to another location called a destination without modifying the content of source. These instructions move data between registers or between memory locations and registers. For eg. MOV, MVI etc.

2) Arithmetic Group :

The instruction of this group performs arithmetic operations such as addition, subtraction, increment or decrement etc. on data in registers or memory. For eg. ADD, SUB, INR, DCR etc.

3) **Logical Group :**

The logical group instructions perform logical operations such as AND, OR, XOR, complement etc. generally with the accumulator.

4) **Branching Group :**

The branching group instructions allow programmer to change the sequence of execution of program either conditionally or unconditionally.

For e.g. JMP, JC, JZ etc.

5) **Machine Control Group :**

These instructions control machine operations such as Halt, Interrupt

For e.g. NOP, HLT.

Q. 7 How instructions of 8085 are grouped according to its length ?

Ans. : Instructions of microprocessor 8085 are grouped into three groups according to its length as follows :-

(i) One byte instructions (One word)

(ii) Two byte instructions (Two word)

(iii) Three byte instructions (Three word)

(i) One byte instructions :

These instructions are having opcode and operand both in one byte. These require only one byte to store in memory.

OPCODE & OPERAND	1 Byte

e.g. MOV A B, ADD B etc.

(ii) Two byte instructions :

These instructions require two bytes to store in memory. First byte gives opcode and second byte gives operand, which is generally 8-bit immediate data.

OPCODE	First Byte
Data/Address	Second Byte

e.g. SUI 35 H, ADI FF H etc.

(iii) Three byte instructions :

These instructions require three bytes to store in memory. First byte gives opcode and second and third byte gives 16-bit address of memory location or 16-bit immediate data. Note that second byte gives lower order address/data byte and third byte gives higher order address/ data byte.

OPCODE	First Byte
Low order 8-bit of address/data	Second byte
High order 8-bit of address/data	Third byte

e.g. LDA 8AFF H,

LXI H, FFE3 H

Q. 8 Explain the following instructions of 8085 microprocessor with suitable example.

(a) SPHL (b) PCHL **(Mar. 02, 2010, 03, Oct. 08. 10)**

Ans. : (a) SPHL : [Copy H and L register to SP]

Format	:	$[SP_L] \leftarrow [L]$ $[SP_H] \leftarrow [H]$
Addressing	:	Register addressing
Group	:	Machine control group [stack operation]
Bytes	:	1 byte
Flag	:	None

Comment : This instruction copies the content of register L into lower order byte of stack pointer and the content of register H into higher order byte stack pointer. The content of registers H and L are not affected. This instruction is used for initializing the stack pointer.

Example : Let, [H] = 25 H and [L] = 59 H
Instruction : SPHL
After execution : [SP] = 2559 H

(b) **PCHL :** **(June 2016)**

[LOAD PROGRAM COUNTER WITH HL REGISTER PAIR CONTENT]

Format	:	$[PC_H] \leftarrow [H]$ $[PC_L] \leftarrow [L]$
Addressing	:	Register addressing
Group	:	Branching group
Bytes	:	1 byte
Flag	:	None

Comment : This instruction moves the content of register H to higher order byte of program counter and the content of register L to lower order byte of program counter. This instruction is equivalent to one byte unconditional jump instruction, with jump address stored in H-L pair.

Example : Let, [H] = 25 H and [L] = 39 H
Instruction : PCHL
After execution : [PC] = 2539 H

After execution of PCHL instruction, the control will be transferred to memory location 2539 H.

Q. 9 Explain PCHL Instruction of micro-processor 8085 and justify the statement that it is equivalent to 3 byte unconditional jump instruction. **(Oct. 2012)**

Ans. : For only Explanation of PHCL Please refer Q. 8(b), Pg. No. 2-6.

In PCHL program counter is loaded with memory address stored in HL pair. In JMP address program counter is loaded with memory address specified in instruction. Working of both instruction is same i.e. load PC with memory address hence PCHL is equivalent to unconditional jump

Q. 10 What are branching instructions ? Explain the jumping instructions with jump conditions. **(March 2006, Oct. 2009; July 18)**

Ans. :

(1) The instructions which allow user to change the control of flow of program execution are called the branching instructions.

(2) There are two types of branch instructions - unconditional and conditional. In unconditional transfer, no condition is tested. In conditional transfer the status of one of the flag is tested to determine whether the branch is to be executed or not.

Two types of jump instructions are as follows :

(3) Unconditional jump :

Instruction	:	JMP addr
Format	:	[PC] ← addr
Addressing	:	Immediate addressing
Group	:	Branching group
Bytes	:	3 bytes
Flag	:	None

Comment : The control is transferred unconditionally to the memory location, whose address is specified in the instruction.

(4) Conditional jump :

In conditional jump instructions , the jump is taken only if the condition is true. The conditional jump instructions and conditions are as given below.

i)	JNZ addr	:	Jump on not zero (Z = 0)
ii)	JZ addr	:	Jump on zero (Z = 1)
iii)	JNC addr	:	Jump on not carry (Cy = 0)
iv)	JC addr	:	Jump on carry (Cy = 1)
v)	JPO addr	:	Jump on odd parity (P = 0)
vi)	JPE addr	:	Jump on even parity (P = 1)
vii)	JP addr	:	Jump on plus (S = 0)
viii)	JM addr	:	Jump on minus (S = 1)

If the condition is satisfied, then only the address of memory location specified in the instruction is loaded in program counter.

Q. 11 How a subroutine can be called conditionally and unconditionally ? OR

Explain CALL instruction of 8085 microprocessor. OR

In case of 8085 microprocessor, explain unconditional and conditional call instructions with suitable example. **(Mar.09,Oct 2010; March 2018; July 18)**

Ans. :

(1) A subroutine is a short set of program instructions written separately from the main program to perform a function that occurs repeatedly in the main program.

(2) A subroutine can be called by two ways :

(a) Unconditional call :

Instruction : CALL addr

Format : $[[SP] - 1] \leftarrow [PC_H]$

$[[SP] - 2] \leftarrow [PC_L]$

$[SP] \leftarrow [SP] - 2$

$[PC] \leftarrow addr$

Addressing : Immediate addressing

Group : Branching group

Bytes : 3 bytes

Comment : CALL instruction is used to call a subroutine unconditionally. Before the control is transferred to the subroutine, the address of next instruction to be executed of the main program is stored in the stack. The content of SP are decremented by 2. Then, the program jumps to the subroutine whose starting address is specified in the instruction.

(b) Conditional Call :

In conditional call, the subroutine is called only if the condition is satisfied. In conditional call, following procedure is followed, if the condition is true.

Format : $[[SP] - 1] \leftarrow [PC_H]$

$[[SP] - 2] \leftarrow [PC_L]$

$[SP] \leftarrow [SP] - 2$

$[PC] \leftarrow addr$

The conditional call instructions and conditions are as listed below :

i)	CC addr	:	Call if carry (Cy = 1)
ii)	CNC addr	:	Call if no carry (Cy = 0)
iii)	CZ addr	:	Call if zero (Z = 1)
iv)	CNZ addr	:	Call if no zero (Z = 0)
v)	CP addr	:	Call if plus (S = 0)
vi)	CM addr	:	Call if minus (S = 1)
vii)	CPO addr	:	Call if odd parity (P = 0)
viii)	CPE addr	:	Call if even parity (P = 1)

Q. 12 Explain return procedure in RET instruction.

Ans. :

(1) The RET instructions are used to return from subroutine to main program.

(2) The return instruction is written at the end of the subroutine indicating end of subroutine.

(3) Instruction : RET

Format	:	$[PC_L] \leftarrow [[SP]], [PC_H] \leftarrow [\,[SP] + 1]$ $[SP] \leftarrow [SP] + 2$
Addressing	:	Register indirect
Group	:	Branching group
Bytes	:	1 byte

Comment : The content of memory location, whose address is specified in stack pointer are moved to the lower order byte of program counter. The content of the memory location whose address is one more than the content of SP moved to the higher order byte of program counter. The contents of stack pointer are incremented by 2.

(4) On completing the execution of subroutine the RET instruction is executed which loads back the stack contents i.e. addresses of the instruction following CALL instruction, in program counter. Thus the control returns into main program.

Q. 13 Explain the use of stack and stack pointer register in Intel 8085. **[Oct. 05]**

Ans. :

1) The stack is a part of Read/Write memory that is used for temporary storage of binary information during the execution of a program.

2) The binary information is basically the immediate results and the return address in case of subroutine programs.

3) The stack is shared by the programmer and the microprocessor.

4) The programmer can store and retrieve the contents of a register pair by using PUSH and POP instructions.

5) Similarly, the microprocessor automatically stores the contents of the program counter when subroutine is called.

6) The stack is implemented with the help of special memory pointer register called the stack pointer.

7) During PUSH and POP operation stack pointer register gives the address of memory where the information is to be stored or to be read.

8) The memory location currently provided by stack pointer is called as top of stack.

Q. 14 Accumulator contains data 2A H. What will be content of accumulator after execution of each instruction independently?

(i) CMA (ii) ANI 05 H (iii) STC

Ans. : (i) CMA : Complement the accumulator

Before execution	:	[A] = 2A H = 0 0 1 0 1 0 1 0
Instruction	:	CMA
After execution	:	[A] = 1 1 0 1 0 1 0 1 = D5 H
i.e.		[A] = D5 H

(ii) **ANI 05 H** : AND immediate data 05 H with [A]

Before execution : [A] = 2A H = 0 0 1 0 1 0 1 0

Instruction : ANI 05 H

2AH = 0 0 1 0 1 0 1 0

AND 05 H = $\underline{0\,0\,0\,0\,0\,1\,0\,1}$

0 0 0 0 0 0 0 0 = 00 H

i.e. [A] = 00 H

(iii) **STC :** Set carry

Before execution : [A] = 2AH

Instruction : STC

This instruction will only set carry flag to 1 and contents of accumulator will remain as it is.

After execution : [A] = 2AH

i.e. [A] = 2AH and [Cy] = 1

Q. 15 **Accumulator contains data E3H. What will be the content of accumulator after stepwise execution of each of following instructions ?** **(Oct. 2008)**

(i) ANI 58 H **(ii) RRC** **(iii) CMA**

Ans. : (i) ANI 58 H : Logically ANDed 58 H with [A]

Before execution : [A] = E3H

Instruction : ANI 58 H

E3 H = 1 1 1 0 0 0 1 1

AND 58 H = $\underline{0\,1\,0\,1\,1\,0\,0\,0}$

0 1 0 0 0 0 0 0 = 40 H

After execution [A] = 40 H

(ii) **RRC** : Rotate accumulator right by one bit.

Before execution : [A] = 40 H = 010000000

Instruction : RRC

After execution [A] = 0 0 1 0 0 0 0 0 = 20 H

and [Cy] = 0 H

∴ [A] = 20 H

(iii) **CMA :** Complement the contents of accumulator

Before execution : [A] = 20 H = 0 0 1 0 0 0 0 0

Instruction : CMA

After execution : [A] = 1 1 0 1 1 1 1 1 = DFH□ A] = DFH

Q. 16 **The accumulator in 8085 contains the data B8H and register B contains data 40 H. What will be the content of accumulator after execution of each of the following instructions independently ?** **(March 2002)**

(a) RLC **(b) ORI 29 H** **(c) ANA B**

Ans. :

(a) **RLC :** Rotate accumulator left through carry.

Before execution : [A] = B8H = 10111000

Instruction : RLC

After execution : [A] = 0 1 1 1 0 0 0 1 = 71 H

∴ [A] = 71 H and [Cy] = 1 H

(b) **ORI 29 H :** Logically ORed 29 H with [A]

Before execution : [A] = B8H = 10111000

Instruction : ORI 29 H

B8H = 1 0 1 1 1 0 0 0

OR 29 H = $\underline{0\,0\,1\,0\,1\,0\,0\,1}$

1 0 1 1 1 0 0 1 = B9H

After execution : [A] = B9H

(c) **ANA B :** Logically AND [Reg. B] with [A]

Before execution : [A] = B8H = 1 0 1 1 1 0 0 0

[Reg. B] = 40 H = 0 1 0 0 0 0 0 0

Instruction : ANA B

1 0 1 1 1 0 0 0

AND $\underline{0\,1\,0\,0\,0\,0\,0\,0}$

0 0 0 0 0 0 0 0 = 00 H

After execution :

[A] = 00 H

Q. 17 **The accumulator contains the data A4H. What will be its contents after execution of following instructions independently.** **(Oct. 02)**

i) XRI 08H ii) CMA iii) SUB A

Ans. :

Accumulator = A4 H

= 1010 0100

i) **XRI 08 H :**

Logically Ex-ORed 08H with contents of accumulator.

Before execution : [A] = A4 H

Instruction : XRI 08 H

A4 H = 1 0 1 0 0 1 0 0

XOR 08 H = $\underline{0\,0\,0\,0\,1\,0\,0\,0}$

1 0 1 0 1 1 0 0

= ACH

After execution : [A] = ACH

ii) CMA :

Complement the accumulator

Before execution : [A] = A4 H

= 1 0 1 0 0 1 0 0

Instruction : CMA

After execution :

[A] = 0 1 0 1 1 0 1 1

= 5B H

i.e. [A] = 5B H

iii) SUB A :

Subtract accumulator from itself.

Before execution : [A] = A4 H

Instruction : SUB A

[A] : A4 H = 1 0 1 0 0 1 0 0

2's complement of A4 = 0 1 0 1 1 1 0 0

+ [A] : A4 = $\underline{1\,0\,1\,0\,0\,1\,0\,0}$

$\boxed{1}$ 0 0 0 0 0 0 0 0

complement carry ↓

$\boxed{0}$ 0 0 0 0 0 0 0 0

Result [A] = 00 H

Q. 18 The accumulator of 8085 contains data 43H. What will be its contents after the execution of following instructions independently ?

(i) CMA (ii) ANI 09H (iii) INR A (Oct. 03)

Ans. :

i) **CMA** : Before executions accumulator content is :

[A] = 43 H

= 0 1 0 0 0 0 1 1

CMA instruction complements the contents of accumulator and result is stored in the accumulator itself.

[A] = 1 0 1 1 1 1 0 0

= BCH

∴ After execution : [A] = BCH

ii) ANI 09H : This instruction logically ANDed 09H with the contents of accumulator.

$$\begin{array}{rcl} \text{Before execution : [A]} & = & 43\text{ H} \\ & = & 01000011 \\ \text{[A]} = 43\text{H} & = & 01000011 \\ \text{AND } 09\text{H} & = & \underline{00001001} \\ & & 00000001 = 01\text{ H} \end{array}$$

∴ After execution : [A] = 01H

iii) INR A : This instruction increments the contents of accumulator by one and result stored in the accumulator itself.

Before execution :

$$\begin{array}{rcl} \text{[A]} = 43\text{ H} & = & 01000011 \\ \text{[A]} = 43\text{ H} & = & 01000011 \\ +\ 0\,1\text{H} & = & \underline{00000001} \\ & & 01000100 = 44\text{ H} \end{array}$$

∴ After execution : [A] = 44 H

Q. 19 The accumulator of 8085 microprocessor contains the data 45 H and register E contains the data 7BH. What will be the content of accumulator after execution of each of following instructions independently ? **(Mar. 04, 2011)**

Ans. : (i) XRA E : This instruction logically EX-ORed the contents of register E i.e. 7BH with the contents of accumulator i.e. 45H.

Before execution :

$$\begin{array}{lrcl} & \text{[A]} = 45\text{H} & = & 01000101 \\ & \text{[E]} = 7\text{BH} & = & 01111011 \\ \text{Then,} & 45\text{ H} & = & 01000101 \\ \text{XOR,} & 7\text{BH} & = & \underline{01111011} \\ & & & 00111110 = 3\text{E H} \end{array}$$

So after execution of instruction,

$$\text{[A]} = 3\text{E H}$$

(ii) ADI C5H :

This instruction adds the data C5H to the content of accumulator i.e. 45H.

$$\begin{array}{rcl} \text{[A]} = 45\text{H} & = & 01000101 \\ +\text{C5H} & = & \underline{11000101} \\ & & 00001010 = 0\text{A H} \\ \text{After execution,[A]} & = & 00001010 \\ & = & 0\text{A H} \end{array}$$

(iii) ORI 5BH :

This instruction logically ORed the data 5BH with the content of accumulator i.e. 45H.

$$\begin{array}{rcl} [A] = 45H & = & 0\,1\,0\,0\,0\,1\,0\,1 \\ OR\ 5BH & = & \underline{0\,1\,0\,1\,1\,0\,1\,1} \\ & & 0\,1\,0\,1\,1\,1\,1\,1 = 5F\,H \end{array}$$

After execution, [A] = 0 1 0 1 1 1 1 1 = 5F H

Q. 20 Accumulator of 8085 contains data 56 H. What will be the contents after the execution of following instruction independently. (March 2005)

(i) CMA (ii) ANI ACH (iii) INR A

Ans. : [A] = 56H = 0 1 0 1 0 1 1 0

(i) CMA :

This instruction complements the contents of accumulator and result is placed in accumulator itself.

$$[A] = 56\,H = 0\,1\,0\,1\,0\,1\,1\,0$$

After execution : CMA

$$\begin{array}{rcl} [A] = 1\,0\,1\,0\,1\,0\,0\,1 & = & A\,9\,H \\ \therefore\ [A] & = & A9H \end{array}$$

(ii) ANI ACH :

This instruction is logically ANDed ACH with the contents of accumulator.

Before execution :

$$\begin{array}{rcl} [A] = 56\,H & = & 0\,1\,0\,1\,0\,1\,1\,0 \\ Data = ACH & = & 1\,0\,1\,0\,1\,1\,0\,0 \\ [A] = 56\,H & = & 0\,1\,0\,1\,0\,1\,1\,0 \\ AND\ ACH & = & \underline{1\,0\,1\,0\,1\,1\,0\,0} \\ & & 0\,0\,0\,0\,0\,1\,0\,0 = 04\,H \end{array}$$

After execution : [A] = 04 H

(iii) INR A : This instruction increments the contents of accumulator by one and result stored in the accumulator itself.

Before execution :

$$\begin{array}{rcl} [A] & = & 56\,H = 0\,1\,0\,1\,0\,1\,1\,0 \\ [A] = 56\,H & = & 0\,1\,0\,1\,0\,1\,1\,0 \\ +\,01H & = & \underline{0\,0\,0\,0\,0\,0\,0\,1} \\ & & 0\,1\,0\,1\,0\,1\,1\,1 = 57\,H \end{array}$$

$\therefore$ After execution : [A] = 57 H

Q. 21 If ACC contains data BCH, register C contains ADH. What will be the content of accumulator after execution of each of the following instructions independently?
(i) SUB C (ii) CMA (iii) XRA C **(Mar. 06)**

Ans. : Accumulator = BCH = 1 0 1 1 1 1 0 0

(i) SUB C : Subtract contents of register C from accumulator.

Before execution :

[A] = BCH = 1 0 1 1 1 1 0 0

[C] = ADH = 1 0 1 0 1 1 0 1

Instruction : SUB C

$$
\begin{array}{rcl}
[A] = \text{BCH} & = & 10111100 \\
[C] = \text{ADH} & = & 10101101 \\
\text{2's complement of } [C] & = & 01010011 \\
[A] & = & 10111100 \\
\text{+ 2's complement of } [C] & = & \underline{01010011} \\
\boxed{1} & & 00001111
\end{array}
$$

complement carry ↓

$$
\begin{array}{rcl}
\boxed{0} & & 000001111 \\
\text{Result } [A] & = & 00001111 = \text{0F H}
\end{array}
$$

(ii) CMA : Complement the accumulator.

Before execution :

[A] = BCH = 1 0 1 1 1 1 0 0

Instruction : CMA

After execution :

[A] = 0 1 0 0 0 0 1 1 = 43 H

Result = [A] = 43H

(iii) XRA C : Logically Ex-ORed contents of C with contents of accumulator..

Before execution : [A] = BCH

Instruction XRA C

$$
\begin{array}{rcl}
[A] = \text{BCH} & = & 10111100 \\
\text{XOR } [C] = \text{ADH} & = & \underline{10101101} \\
 & & 00010001 \\
 & = & \text{11 H}
\end{array}
$$

After execution : [A] = 11 H

Q. 22 The accumulator of 8085 processor contains data B8H and register B contains 44 H. What will be the content of accumulator after execution of each of the following instruction independently ? **(March 2013)**

(i) ORI F0H (ii) ANA B (iii) XRI 0FH

Ans. :

Accumulator A = B 8 H =1011 1000

(i) ORI F0 H — logically OR ed F0 H with [A]

B8 H = 10 11 1000

ORI F0 H = 11 11 0000

11 11 1000 = F 8 H

After execution [A] = F 8 H

(ii) ANA B – Logically AND (Reg B) with [A]

B8 H = 10 11 1000

AND B – 44 H = 01 00 0100

00 00 0000 = 00 H

After execution [A] = 00 H

(iii) XRI 0F H – Logically Ex-ored 0F H with [A]

B8 H = 10 11 1000

XRI 0F H = 00 00 1111

10 11 0111 = B 7 H

After execution [A] = B7 H

Q. 23 If Accumulator Contains the Data 23H and B Register Contains 35H. What will be the contents of Accumulator. After execution of each of the following instruction independently : **(Oct. 2013)**

(i) XRA (ii) ANI FOH (iii) CPI OAH

Ans. :

A = 23 H = 0010 0011

B = 35 H = 0011 0101

(i) XRA B – Ex.ored reg. B with contents of A.

$$\begin{array}{rl} 23\text{ H} = & 0010\ 0011 \\ \text{XOR with } 35\text{ H} = & \underline{0011\ 0101} \\ = & \underbrace{0001}_{1}\ \underbrace{0110}_{6} \end{array}$$

A = 16 H

(ii) ANIFOH – Logically AND FOH with contents of A.

$$\begin{array}{rcl} \text{FO} & = & 1111\ 0000 \\ \text{AND } 23 & = & \underline{0010\ 0011} \\ & & \underline{0010}\ \underline{0000} \\ & & \ \ 2 \qquad 0 \end{array}$$

A = 20 H

(iii) CPIOAH – Compare OAH with A reg.

A = 23 H ← Before execution

While compairing Accumulator remains

Unchanged hence A = 23 H

Q. 24 Accumulator contents are B8H and Register B contents are C9H. What are the contents of Accumulator and Flag register after execution of instructions ANA B, SUB B independently. **(March 2014)**

Ans. :

(A) = B8H = 10111000

(B) = C9H = 11001001

1) ANAB : Logically AND with Accumulator

$$\begin{array}{r} 1011\ 1000 \\ \underline{1100\ 1001} \\ 1000\ 1000 \end{array}$$

A = 88H

Flags → S = 1, Z = 0, AC = 0, P = 1, CY = 0

(2) SUB B : Subtract B Reg. from Accumulator

$$\begin{array}{rcl} \text{B} & = & 1100\ 1001 \\ \text{1's complement of B} & = & 0011\ 0110 \\ +\ 1 & + & 1 \\ \text{2's complement of B} & & 0011\ 0111 \end{array}$$

Add A with 2's complement of B

$$\begin{array}{r} 0011\ 0111 \\ \underline{+\ 1011\ 1000} \\ \boxed{0}\ 1110\ 1111 \end{array}$$

Complement carry ↓

$$\boxed{1}\ 1110\ 1111$$

Result → A = EFH

Flags → S = 1, Z = 0, AC = 0, P = 0, CY = 1

Q. 25 For the following instructions, write the addressing mode, instruction group and the length of the instruction (in terms of bytes). **(Oct. 05)**

(i) LHLD ABCDH (ii) LDAX B

(iii) LXI H, BABAH (iv) SPHL

Ans. : (i) LHLD ABCDH : Please refer Appendix I (9).

(ii) LDAX B : Please refer Appendix I (11).

(iii) LXI H, BABAH : Please refer Appendix I (6).

(iv) SPHL : Please refer Appendix V (6).

Q. 26 The following instructions are intended to clear ten (10) memory locations starting from the memory address 0009H. Explain why a large memory block will be erased or cleared and the program will stay in an infinite loop. **(Oct. 05)**

```
LXI H, 0009H
Loop MVI M, 00H
DCX H
JNZ Loop
HLT
```

Ans. :

(1) In given loop, large memory block will be erased or cleared and the program will stay in an infinite loop.

(2) In the given loop, the sequence is repeated by the instruction JNZ (Jump on No zero) until the count becomes zero. However, the instruction DCX does not set the zero flag. Therefore, the instruction JNZ would be unable to recognize when the count has reached zero and the program would remain in a continuous loop.

Q. 27 Explain following instructions of 8085 microprocessor. **(March 2002)**

(i) ORI data (ii) STAX rp (iii) LHLD addr

Ans. : Refer Appendix

Q. 28 What are different addressing modes ? Which type of addressing mode is used for following instructions ? **(Oct. 1998)**

(i) XCHG (ii) XRI (iii) SUB M (iv) CMC

Ans. : Refer Q.1 and appendix.

Q. 29 Identify the addressing modes of the following instructions and justify your answer. (i) LDA 2000 H (ii) LDAX B (iii) STC (iv) ADC D **(March 1998)**

Ans. : Refer appendix – to find addressing mode and refer Q. 2 to justify answer.

Q. 30 Describe following instructions of 8085 microprocessor. **(Oct. 2002)**

(i) XCHG (ii) RAR (iii) ADC R

Ans. : Please refer appendix.

Q. 31 Explain the following instructions of 8085 microprocessor with suitable example. **(Oct. 2002)**

(i) RLC (ii) DAA

Ans. : Refer appendix.

Q. 32 Explain the following instructions. **(March 2003)**

(i) PCHL (ii) PUSH PSW (iii) OUT (iv) NOP

Ans. : Refer appendix.

Q. 33 Explain the addressing modes of following instructions : **(Oct. 2006)**

(i) LDA (ii) STAX (iii) CMA

Ans. : Refer appendix.

i) Group I (7) ii) Group I (12) iii) Group III (17)

Q. 34 Accumulator contains 45H [(A) = 45], Register E contains data 3BH [(E) =3B] Write the contents of Accumulator after execution of following instructions independently : i) SUB E ii) XRA E iii) RRC iv) MOV E,A **(Oct. 2009, 2010)**

Ans : a) [A] = 45 H = 0100 0101

[E] = 3BH = 0011 1011

i) SUB E : Subtract contents of register E from accumulator.

[A] = 45 H = 0100 0101

[E] = 3BH = 0011 1011

2's complement of [E]

= 1's complement of [E] + 1

= 1 1 0 0 0 1 0 0 + 1

= 1 1 0 0 0 1 0 1

∴ [A] = 0 1 0 0 0 1 0 1

2's complement of [E] = $\underline{1\ 1\ 0\ 0\ 0\ 1\ 0\ 1}$

= $\boxed{1}$ 0 0 0 0 1 0 1 0

complement carry $\boxed{0}$ 0 0 0 0 1 0 1 0

∴ Result [A] = 0 0 0 0 1 0 1 0

= 0 A H

ii) **XRA E :** Logically Ex–ORed contents of E with contents of accumulator.

[A] = 45 H = 0 1 0 0 0 1 0 1

XOR [E] = 3BH = $\underline{0\ 011\ 1011}$

0111 1110 = 7E H

Result : [A] = 7EH

iii) RRC : Rotate accumulator right by one bit

Before execution :

[A] = 45 H = 0 1 0 0 0 1 0 1

After execution :

[A] = 1 0 1 0 0 0 1 0

= A2 H

∴ [A] = A2H

iv) Mov E, A : Copy accumulator contents to register E.

Before execution :

[A] = 45 H = 0 1 0 0 0 1 0 1

[E] = 3BH = 0 0 1 1 1 0 1 1

After execution :

[A] = 45 H = 0 1 0 0 0 1 0 1

[E] = 45 H = 0 1 0 0 0 1 0 1

∴ **[A] = 45H**

Q. 35 The accumulator contains 05H and register B contains 08H.What will be the effect of 'SUB B' instruction on flags ? Explain it with diagram. **(March 2010)**

Ans : ACC = 05 H = 0000 0101

B = 08 H = 0000 1000

After execution of SUB B

Accumulator contains 2's compliment of magnitude of result

i.e. ACC = 1111 1101 = FDH

Mentioning the status of carry flag

Q. 36 Differentiate DAD and ADD Instruction of 8085 Micro-Processor **(Oct. 2013)**

Ans. :

	DAD	ADD
1.	In this instruction contents of register pair rp are added to the contents of HL pair & result is placed in register H and L.	In this instruction register r or content of memory location whose address is stored in H-L pair is added with content of accumulator and result is placed in accumulator.
2.	Only carry flags in affected.	All flags are affected.
3.	Register pairs BC, DE are used.	Only register A, B, C, D, H, E, L are used.
4.	Used for 16 bit addition	Used for 8 bit addition

Q. 37 Differentiate between PUSH and POP. **(Oct. 2013)**

Ans. :

	PUSH	POP
1.	The contents of the higher order register of register pair rp are moved to memory location whose address is one less than the content of stack pointer.	The contents of the memory location whose address is specified by the stack pointer are moved to low order register of register pair rp.
2.	The contents of the low order register of register pair rp are moved to the memory location whose address is two less than the content of stack pointer.	The contents of the memory location, whose address is one more than the content of stack pointer are moved to high order register of register pair rp.
3.	In this instruction, stack pointer is decremented by two.	In this instruction, stack pointer is incremented by two.
4.	Let [SP] = D015, [B] = 25 H and [C] = 55 H After execution of PUSH B [D014] = 25 H [D013] = 55H and [SP] = D013 H	Let [SP] = 2001 H [2001] = 10 H, [2002] = 20 H After execution of POP H [H] = 20H, [L] = 10H [SP] = 2003 H

Q. 38 The accumulator of 8085 contains data B7h. What will be its contents after execution of the following instructions independently ? **[March 2012]**

(i) ORI 58 H (ii) CMA (iii) ANI E3 H

Ans. :

Accumulator = B7H

= 1 0 1 1 0 1 1 1

(i) ORI 58 H

Logically ORed 58H with [A]

Before Execution [A] = B7H =1 01 1 0 1 1 1

B7H = 1 0 1 1 0 1 1 1

OR 58H = 0 1 0 1 1 0 0 0

1 1 1 1 1 1 1 1 = FFH

After execution [A] = FFH

(ii) CMA - Complement the contents of accumulator

Before execution [A] = B7H = 1 0 1 1 0 1 1 1

Instruction CMA

After execution [A] = 01001000 = 48 H

(iii) ANIE3H - Logically ANDed E3H with [A]

Before execution [A] = B7H =1 0 1 1 0 1 1 1

Instruction - ANIE3H

B7H = 1 0 1 1 0 1 1 1

OR E3H = 1 1 1 0 0 0 1 1

1 0 1 0 0 0 1 1 = A3H

after execution [A] = A3H

Q. 39 The accumulator contain 3CH, what will be the effect on its content if following instructions are executed independently ? **[Oct. 201**

(i) ANI 05H (ii) RRC (iii) MOV B, A

Ans. : A = 3C H = 00111100

(i) ANI 05 H - Logically And 05 H with Accumulator

05 H = 00000101

A = 00111100 = 3C H

A = 00001000 = 04 H

(ii) RRC - Rotate Accumulator Right

A = 3C H = 00111100

After Execution of RRC A will contain 00011110 = 1E H

(iii) MOV B, A - Move contents of A reg in B register

Accumulator contents remain unchanged i.e. B & A will be 3C H

Q. 40 The accumulator in 8085 Micro-processor contains the data 78H and register D contains data 33H. What will be the content of accumulator after execution of each of the following instructions independently. **(Oct. 2015)**

(i) SUB D (ii) AND D (iii) CMA

Ans. :

Accumulator = A = 78 H = **0111 1111**

D = 33H = 0011 0011

(ii) SUB D (Subtract D from Accumulator)

is complement of D = 1100 1100

+ 1 + 1

2's Complement of D = 1100 1101

A =

= 0111 11 11

111 11 11

= $\boxed{1}$ 0100 11 00

Complement carry $\boxed{0}$ 0100 1100

Result = A = 4 CH

(ii) AND D : Logically and with accumulator

A = 78 H = 0111 1111

AND = 0011 0011

0011 0011

A = 33 H

(iii) CMA : Complement Accumulator

A = 78 H = 0111 1111

After CMA, A = 1000 0000

A = 80 H

Q. 41 The accumulator in 8085 microprocessor contains data 71H register E contains data 39H. What will be the contents of accumulator in Hexadecimal after execution of the following instructions independently ? **(March 2017)**

(i) ADD E (ii) ORA E (iii) RRC

Ans. : (i) AAH, (ii) 79H, (iii) B8H

Q. 42 Accumulator contains data A4H and Register E contains data 69H write the contents of Accumulator in hex digits after execution of each of the following instructions independently : (July 2017)

Ans. :

(i) ANA E = 20 H (ii) CMP E = A4H (iii) ORA = EDH

Q. 43 The registers A and C of 8085 contains the data E2H and 47H. What will be the contents of Accumulator in Hex digits after execution of each of the following instructions independently ? (March 2018)

(i) SUB C (ii) XRA C (iii) ADD C

Ans. :

(i) SUB C = 9BH

(ii) XRA C = A5H

(iii) ADD C = 29 H Cy = 1

Q. 44 Accumulator contain data 45H and register B contain data 82H. What will be the result in Accumulator after execution of each instruction independently. (July 2018)

(i) XRA B (ii) ADI 54H (iii) NI 57H

Ans. :

(i) XRA B

Accumulator	=	45H	
reg B	=	82H	
Acc = 45H	=	0100	0101
B = 82H	=	1000	0010
		1100	0111
		B	7
XRA B	=	B7H	

(ii) ADI 54H

Acc = 45H	=	0100	0101
54H = 54H	=	0101	0100
		1001	1001
		9	9
ADI 54H	=	99H	

(iii) ANI 57H

Acc = 45H	=	0100	0101
57H = 57H	=	0101	0111
		0100	0101
		4	5
ANI 57H	=	45H	

Q. 45 Write the appropriate instructions for the following task : **(July 2018)**

(i) Load accumulator from B register.

(ii) Complement the accumulator

(iii) Add 01H with the accumulator

(iv) Store the content of accumulator at the memory location addressed by the BC register pair.]

(v) Clear the accumulator.

Ans. :

(i)	Load accumulator from B register	–	LDAX B / MOV A, B
(ii)	Complement the accumulator	–	CMA
(iii)	Add 01H with the accumulator	–	ADI 01H
(iv)	Store the content of accumulator at the memory location addressed by the BC register pair.]	–	STAX B
(v)	Clear the accumulator.	–	XRA A / SUB A / MVI A, 00

Q. 46 Give any two instructions of following addressing modes : **(March 2019)**

(i) Immediate **(ii) Register Indirect** **(iii) Register**

Ans. :

(i)	Immediate	–	ADI 05H / MVI B, 04H
(ii)	Register Indirect	–	ADD M / MOV M, D
(iii)	Register	–	MOV A, B / ADD C

Q. 47 The accumulator contains AA H and register C contains 55 H. What will be the contents of accumulator if following instructions are executed independently ?

(i) CMP C **(ii) ANA C**

(iii) ORA C **(iv) SUB C** **(March 2019)**

Ans. :

(i) **CMP C** It's CMP means subtraction of A-C. But after subtraction Accumulator content remains unchanged. Means after CMPC.

A = AA H

(ii) **ANA C**

A	=	AA H	=	1010	1010
C	=	55 H	=	0101	0101
logical AND operation			=	0000	0000

∴ ANA C = 00 H Accumulator Content

(iii) ORA C

$$\begin{array}{lcllcc}
 & & A = AA\,H & = & 1010 & 1010 \\
 & & C = 55\,H & = & 0101 & 0101 \\
\hline
\text{logical OR operation} & & & = & 1111 & 1111 \\
\therefore\ \text{ORA C} = FF\,H & & & & \multicolumn{2}{l}{\text{Accumulator Content}}
\end{array}$$

(iv) SUB C

$$\begin{array}{lcccl}
A = AA\,H = & & 1010 & 1010 & \\
C = 55\,H = & & 0101 & 0101 & \\
\hline
 & & 1010 & 1010 & \text{1's complement} \\
 & + & & 1 & \text{Add 1} \\
\hline
 & & 1010 & 1011 & \text{2's complement} \\
 & & 1010 & 1010 & \text{1's no.} \\
\hline
\text{Substraction} & & 0\ 0101 & 0101 & \therefore\ \text{SUB C} = 55H
\end{array}$$

Q. 48 Select the correct alternative and rewrite the following.

1 .... instruction belongs to data transfer group of instruction set of 8085.

(i) LHLD (ii) CMA (iii) JMP (iv) POP

Ans. : (i) LHLD

2. .... flag is affected by the instruction RRC of 8085.

(i) zero (ii) parity (iii) carry (iv) all

Ans. : (iii) carry

3. Which of the following instruction does not affect any flag ....

(i) ADD (ii) RAR (iii) STC (iv) PCHL

Ans. : (iv) PCHL

4. Instruction STAX belongs to .... addressing mode.

(i) Direct (ii) Register (iii) Register indirect (iv) Immediate

Ans. : (iii) Register indirect

5. In 8085 .... instruction affects flag register.

(i) MOV B, A (ii) CMA (iii) MVI A, data (iv) CPI data

Ans. : (iv) CPI data

6. Instruction PCHL belongs to .... group

(i) Arithmetic operation

(ii) Logical operation

(iii) Data transfer

(iv) Branching operation

Ans. : (iv) Branching operation

7. LXI H, addr is .... byte instruction. **(March 2002**

(i) 1 (ii) 2 (iii) 3 (iv) 4

Ans. : (iii) 3