Instruction :-

- An instruction is a command given in binary pattern to the microprocessor to perform a specified operation on given data.
- The entire group of instructions is called as Instruction set of microprocessor.
- Each instruction consist of two parts : opcode & operand.

→ Opcode

The first part of instruction which specifies the operation to be performed by the computer is called opcode.

→ Operand

The second part of the instruction is the data to be operated on, is called operand.

- The various techniques to specify data or operand for instruction is given below

a) 8-bit or 16-bit data may be directly given in the instruction itself.

eg MVI A, 02 H — 02H is the 8-bit data.

LXI H, 1500 H — 1500H is the 16-bit data.

H indicates hexadecimal data.

b) The address of Memory location or I/O port address may be given in instruction itself.

eg STA 5000H 5000H is the 16-bit memory address.

c) In some instruction only one register is specified. The content of the specified register is one of the operand

eg ADD B The content of B register is one operand and the content of A register is other operand

d) Some instruction specify one or two register. The content of registers are the required data.

eg MOV A, B Two registers are specified i.e A & B.

INR B One register is specified i.e B.

e) In some ~~register~~ instruction data is not specified. The most instruction of this type operate on the content of accumulator.

eg CMA Data is in A register.

Instruction format

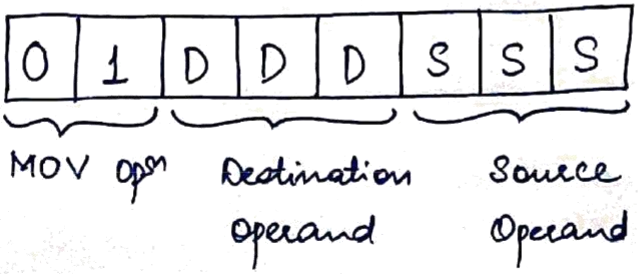
- A digital computer / microprocessor understood instruction in binary codes known as opcodes.
- Due to different ways of specifying data or operand for instructions, the machine code of all instructions are not of same length.
- The size of machine code is one, two or three byte known as Instruction Word Size.
- Depending upon the size of machine codes, the 8085 instructions are classified into three types

(1) One Byte Instruction

A 1 byte instruction include the opcode and operand in the 8-bit only ie one byte.

format | Opcode |

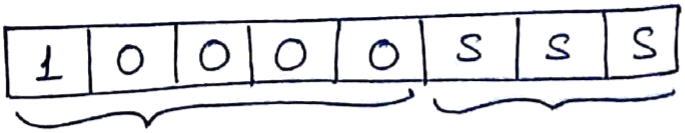
eg (i) MOV Instruction



MOV A, B = 01 111 000

= 78 H

(ii) ADD Instruction



ADD B = 10000 000 = B0 H

(2) Two - Byte Instruction

In two byte instruction the first byte is opcode and second byte is operand.

format

Opcode	Operand

eg (i) MVI B, 05 H Move 05 to register B.

06 , 05

The first byte 06 is the opcode for MVI B and the second byte is operand.

(ii) IN 02 Read the data at port

DB 02

The first byte DB is the opcode for IN and the second byte is operand i.e 8- bit address of port from which the data is transfered to A register.

(3) Three - Byte Instruction

In three byte instruction the first byte is opcode and second and third bytes are operands.

Format:

Opcode	Operand 1	Operand 2

eg (i) LXI H, 2500 H Load HL pair with 2500 H.

21, 00, 25 Code form

The first byte 21 H is the opcode for instruction LXI H. The second byte 00 is the lower byte of data, which is loaded in register L. The third byte 25 is the higher byte of data loaded in register H.

(ii) STA 2600 H Store the content of A register to memory location 2600 H.

32, 00, 60 Code form.

The first byte 32 is the opcode for instruction STA. The second byte 00 is lower byte of address of memory location 2600 H & The third byte 26 is the higher byte of the address of memory location 2600 H.

Opcode Format:

- Each instruction has a unique code.
- The opcode contains information regarding operation, register used, memory to be used.
- The opcode for each instruction is fixed.

Register	Code
B	000
C	001
D	010
E	011
H	100
L	101
M	110
A	111

Register Pair	Code
BC	00
DE	01
HL	10
Sf or SB	11

eg (1) MOV r_1, r_2 Transfer content of r_2 register to r_1 register.

Opcode

0	1	D	D	D	S	S	S

MOV A, B 01 111 000 = 78H

MOV A, C 01 111 001 = 79 H

MOV A, M 01 111 110 = 7E H

(ii) LXI rp, data load HL pair with 16-bit data.

Opcode

0	0	D	D	0	0	0	1

LXI B, Data 00 00 0001 = 01 H

LXI H, Data 00 10 0001 = 21 H

LXI D, Data 00 01 0001 = 11 H

(iii) ADD r Add register to accumulator.

Opcode

1	0	0	0	0	S	S	S

ADD B 10000 000 = 80 H

ADD C 10000 001 = 81 H

ADD E 10000 011 = 83 H