(ii) ADD [0301], AX :- This inst. adds the content of AX to the contents of mem. locations 0301 and 0302.

II Register Indirect Addressing :- The operand's offset is placed in any one of the registers BX, BP, SI or DI as specified in the inst.

(i) MOV AX, [BX] :- This inst moves the contents of memory locations addressed by the reg BX to the reg AX. for eg :- BX contains 0301, the contents of 0301 is 53H and the content of the next mem-location is 95H.

The 9553H will move to AX.

III based addressing :- operand's offset is the sum of an 8-bit / 16-bit displacement and the contents of the base register BX or BP.
↳ used as base reg. for SS.
↓
used as base reg. for DS.

offset := [BX + 8bit/16bit disp]

eg :- MOV AL, [BX+05]

↓ suppose BX contains 0301

0301+05 = 0306.

↑ the contents of 0306 will move to AL.

Addressing Modes of 8086:- [An inst performs specific operation on the specified data (operand). The way by which an operand is specified for an inst is called addressing mode.

8086 has 8 add. modes:

(1) For inst which operates on reg. or imm. data:

→ Reg. addressing:-
MOV AX, CX, ADD AL, BL

→ Immediate add:-
MOV AL, 35H
MOV BX, 0301H

(2) Specify the location of an operand which is placed in a memory. The mem. address of an operand consists of two components:

Starting add + offset

(resides in corresp. seg. reg)

(is determined by adding displacement, base and index.)

any combination of these add elements

→ The combs. depends on the add. mode of the inst.

* Displacement:- 8 bit / 16 bit (immm. value given in the inst.)
* Base (content of the base reg., BX or BP)
* Index " " " index reg, SI or DI).

combs. of three add. elements give six memory addressing modes:-

(1) Direct addressing:- the operand's offset is given in the inst as an 8 bit or 16-bit displacement element.

i) ADD AL, [0301] :- This inst adds the contents of offset address 0301 to AL. The operand is placed at the given offset (0301) within the data seg (DS).

IV Indexed addressing: operand's offset is:

offset = [SI or DI + 8 bit/16 bit disp]

MOV AX, [SI+05]

MOV AX, [SI+1528H]

V. Based Indexed addr: operand's offset

offset = [BX or BP] + [SI or DI]

BP → SS.

used as base reg for DS.

eg: ADD AX, [BX+SI] MOV CX, [BX+SI]

VI Based Indexed with disp:

offset = [BX or SP] + [SI or DI] + 8 bit/16 bit disp

MOV AX, [BX+SI+05]

MOV AX, [BX+SI+1235H]