



4.1 8086 Instruction Set

The instructions of 8086 are classified into SIX groups. They are:

1. Data Transfer Instructions
2. Arithmetic Instructions
3. Bit Manipulation Instructions
4. String Instructions
5. Program Execution Transfer Instructions
6. Process Control Instructions

4.2 Data Transfer Instructions:

The DATA TRANSFER INSTRUCTIONS are those, which transfers the DATA from any one source to any one destination. The data's may be of any type. They are again classified into four groups. They are:

General-Purpose Byte or Word Transfer Instructions	Special Address Transfer Instruction
MOV XCHG	LEA LDS LES

4.3 MOV Revisited Instruction:

The MOV instruction introduces the machine language instructions available with various addressing modes and instructions. Machine code is introduced because it may occasionally be necessary to interpret machine language programs generated by an assembler. Interpretation of the machine's native language (machine language) allows debugging or modification at the machine language level.

The MOV instruction copies a word or a byte of data from a specified source to a specified destination. Data can be moved between general purpose-registers, between a general purpose-register and a segment register, between a general purpose-register or segment register and memory, or between a memory location and the accumulator. Note that memory-to-memory transfers are not allowed here.

Mnemonic	Meaning	Format	Operation	Flags Effected
MOV	Move	MOV D,S	(S) → (D)	none

Example:

```
MOV AL,6+3      ;Copies 9 into AL
MOV AL,6-3      ;Copies 3 into AL
MOV AL,4*3      ;Copies 12 into AL
MOV AX,12/5     ;Copies 2 into AX (remainder is lost)
```



MOV CX, 037AH ; Move 037AH into the CX; 037A → CX
 MOV AX, BX ; Copy the contents of register BX to AX ; BX → AX
 MOV DL, [BX] ; Copy byte from memory at BX to DL ; DS*10+BX → DL

4.4 XCHG Instruction - Exchange XCHG destination, source:

The Exchange instruction exchanges the contents of the register with the contents of another register (or) the contents of the register with the contents of the memory location. Direct memory to memory exchanges are not supported. The both operands must be the same size and one of the operand must always be a register.

Mnemonic	Meaning	Format	Operation	Flags Effected
XCHG	Excgange	MOV D↔S	(D) ↔ (S)	none

Example: Reg→Reg

MOV AL, 5
 MOV AH, 2
 XCHG AL, AH ; AL = 2, AH = 5
 XCHG AL, AH ; AL = 5, AH = 2

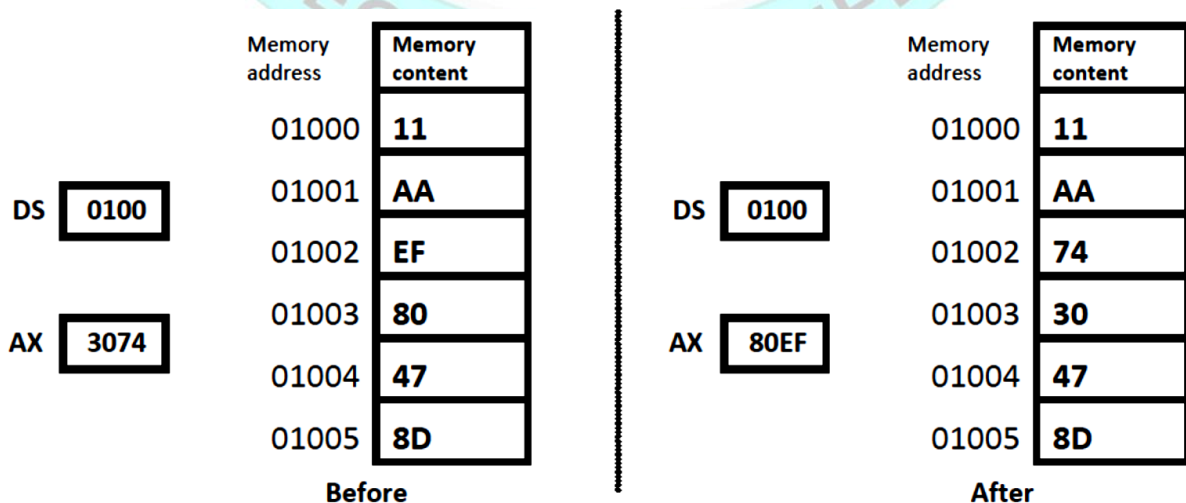
Example: Mem→Reg or Reg→Mem

MOV AL, 12 ; copy 12 (decimal) into AL
 MOV DI, 2000H ; copy 2000H memory offset address into DI
 XCHG AL, [DI] ; copy the memory content addressed by [(DS* 10H)+ 2000H] into AL
 XCHG [DI], AL ; copy AL content into memory addressed by [(DS* 10H)+ 2000H]
 XCHG CX, [037AH] ; [(DS* 10)+ 037A]↔ CX
 XCHG AX, [BX] ; [(DS* 10)+ BX]↔ AX
 XCHG DL, [BP+200H] ; [(SS* 10)+ BP+200]↔DL

Example: For the figure below. What is the result of executing the following instruction?

XCHG AX, [0002]

Solution





4.5 LEA, LDS, and LES (Load Effective Address) Instructions:

These instructions load a segment and general purpose registers with an address directly from memory. The general forms of these instructions are as shown below:

Mnemonic	Meaning	Format	Operation	Flags Effected
LEA	Load register with Effective Address	LEA reg16, EA	EA → (reg16)	None
LDS	Load register and Ds with words from memory	LDS reg16, EA	[PA] → (reg16) [PA+2] → (DS)	None
LES	Load register and ES with words from memory	LES reg16, EA	[PA] → (reg16) [PA+2] → (ES)	None

The LEA instruction loads a 16-bit register with the offset address of the data specified by the operand. The LDS and LES instructions load any 16-bit register with an offset address, and the DS, ES, or SS segment register with a segment address. These instructions use any of the memory-addressing modes to access a 16-bit section of memory that contains both the segment and offset address.

Example:

LEA BX, PRICE ; Load BX with offset of PRICE in DS
 LEA BP, SS:STAK ; Load BP with offset of STACK in SS
 LEA CX, [BX][DI] ; Load CX with EA=BX + DI
 LDS BX, [4326] ; copy the contents of the memory at displacement 4326H in DS to BL, contents of the 4327H to BH. Copy contents of 4328H and 4329H in DS to DS register.

Example 5: Assuming that (BX)=100H, DI=200H, DS=1200H, SI= F002H, AX= 0105H, and the following memory content. what is the result of executing the following instructions?

- LEA SI , [DI + BX +2H]
- MOV SI , [DI+BX+2H]
- LDS CX , [300]
- LES BX , [DI+AX]

Solution:

- LEA SI , [DI + BX +2H]
 $SI = (DI) + (BX) + 2H = 0200H + 0100H + 0002H = 0302H$
- MOV SI , [DI+BX+2H]
 $EA = (DI + BX + 2H) = 0302H$
 $PA = DS * 10 + EA = 1200H * 10 + 0302 = 12302$
 SI = 80EFH
- LDS CX , [300]
 $PA = DS * 10 + EA = 1200H * 10 + 300H = 12300H$
 CX = AA11H and DS = 80EFH
- LES BX , [DI+AX]
 $EA = (DI + AX) = 0200H + 0105H = 0305H$
 $PA = DS * 10 + EA = 1200H * 10 + 0305H = 12305H$
 BX = 5A8DH and ES = C592H

Memory address	Memory content
12300	11
12301	AA
12302	EF
12303	80
12304	47
12305	8D
12306	5A
12307	92
12308	C5