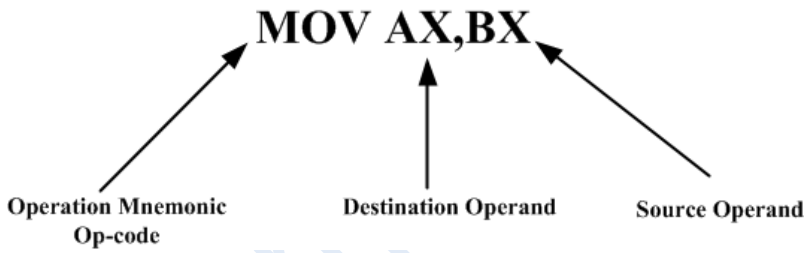
Chapter Two

Addressing Modes

Computer instructions are made up of an Operation Code (op-code) and either one or two operands. The op-code identifies the action to be taken; the operands identify the source and destination of the data operated on. Op-codes are usually written in an abbreviated form called mnemonic. For example the instruction **MOVE** becomes **MOV**, and **JUMP** becomes **JMP** and so on. The operands identify the data that the CPU needs it for processing. In 8086 the general format of an instruction is as follows:



Addressing modes tell where and how to locate data to be accessed. When the CPU executes an instruction, it performs the specified function (Operation) on data. These data are called operands and may be part of the instruction, reside in one of the internal registers of the microprocessor, stored at an address in memory or held at an I/O port. Addressing modes are ways for the microprocessor to access operands.

There are 7 addressing modes types divided into three categories:

1. Register Addressing Mode.
2. Immediate Addressing Mode.
3. Memory Addressing Modes, and can be divided into
    - Direct Addressing Mode.
    - Register Indirect Addressing Mode.
    - Register Relative Addressing Mode.

- Base Plus Index Addressing Mode.
- Base Relative Plus Index Addressing Mode.

3.1 Register Addressing Mode.

In register addressing mode, data is transferred between two registers.

MOV AX, BX ; the data is transferred from register BX to register AX.

MOV CL, DH ; the data is transferred from register DH to register CL.

MOV DS, SI ; the data is transferred from register SI to segment register DS.

3.2 Immediate Addressing Mode.

in immediate addressing mode the data is directly injected into the destination.

MOV AX, 1234h

MOV [2500h], 20h

3.3 Memory Addressing Modes

3.3.1 Direct Addressing Mode

In the direct addressing mode the memory location is known and used directly as an operand.

MOV [1234h], AX; put data from AX to Memory location [1234h]. The physical address for this type of memory can be calculated as follows:

PA= DS*10h+offset.

PA= DS*10h+1234h.

3.3.2 Register Indirect Addressing Mode

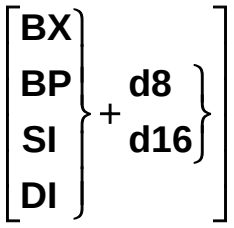
In register indirect addressing mode the memory location is stored in a register.

MOV CX, [BX]; data is transferred from memory location [BX]. The PA can be calculated as follows:

PA= DS*10h+BX.

3.3.3 Register Relative addressing mode.

In this type of memory addressing mode we use one of the registers **(BX,BP,SI,DI)** plus displacement either **(d8)** or **(d16)**:



Ex.s:

1. **MOV AX, [BX+30h]**; Register relative with d8.

In this example when we want to calculate the physical address the **offset** will be the value of **(BX)**+**30**h:

so the physical address is:

PA=DS*10h+(BX+30h)

and if we assume (DS=10006, BX=2000h) then:

=1000*10+(2000+30) = 12030h is the physical address of the instruction above.

2. **MOV [SI+3546h],AX** ; Register relative with d16.

In this example when we want to calculate the physical address the **offset** will be the value of **(SI)**+**3546**h:

so the physical address is:

PA=DS*10h+(SI+3546h)

and if we assume (DS=1000h, SI=2000h) then:

=1000*10+(2000+3546) = 15546h is the physical address of the instruction above

3.3.4 Base plus index addressing mode.

In this addressing mode we use one of the base registers **(BX,BP)** plus one of the index registers **(SI,DI)**:

$$\left[\begin{matrix}BX\\BP\end{matrix}\right\} + \left.\begin{matrix}SI\\DI\end{matrix}\right\}\Bigg]$$

Ex.s:

1. **MOV [BP+DI], 3040h**; Base plus index.

In this example when we want to calculate the physical address the **offset** will be the value of **(BP)**+**(DI)**:

so the physical address is:

PA=SS*10h+(BP+DI)

and if we assume (SS=1000h, BP=2000h, DI=0030h) then:

=1000*10+(2000+0030) = 12030h is the physical address of the instruction above

2. **MOV DL, [BX+SI]**; Base plus index.

In this example when we want to calculate the physical address the **offset** will be the value of **(BX)**+**(SI)**:

so the physical address is:

PA=DS*10h+(BX+SI)

and if we assume (DS=1000h, BX=2000h, SI=1000h) then:

=1000*10+(2000+1000) = 13000h is the physical address of the instruction above

3.3.5 Base relative plus index addressing mode.

In this addressing mode we use one of the base registers **(BX,BP)** plus one of the index registers **(SI,DI)** plus displacement either **(d8)** or **(d16)**.

$$\left[\begin{matrix}BX\\BP\end{matrix}\right\} + \left.\begin{matrix}SI\\DI\end{matrix}\right\} + \left.\begin{matrix}d8\\d16\end{matrix}\right\}\right]$$

Ex:

MOV [BX+SI+1300h], DX; Base relative plus index.

In this example when we want to calculate the physical address the **offset** will be the value of **(BX)**+**(SI)+1300**h:

so the physical address is:

PA=DS*10H+(BX+SI+1300h)

and if we assume (DS=1000h, BX=2000h, SI=1000h) then:

=1000*10+(2000+1000+1300) = 14300h is the physical address of the instruction above.

IMPORTANT NOTE:

Finally in some cases the displacement will be in negative value so when we want to write the code for it we must take the **2's complement** to the negative displacement, for example:

MOV [BP-30H], BX

The **2's** complement of **(-30)** will be **(D0)**, so the instruction will be:

MOV [BP+D0h], BX

3.4 An Easy Way to Remember the 8086 Memory Addressing Modes

You could memorize all the memory forms so that you know which are valid (and, by omission, which forms are invalid) by using the following chart

: