Computer Science Dept.

Department	Computer Science	القسم:
Subject Name:	Microprocessors - (9)	أسم المادة :
Year of Study:	2023-2024	السنة الدراسية:
Term:	Second Term	الفصل الدراسي:
Email	ali.sadoon@uoa.edu.iq	Email
Instructor Name:	***Ali Saadoom AHMED***	أسم التدريسي:

OUTLINE

✓ *Logical vs. physical address of the stack*

✓ *80X86 ADDRESSING MODES*

Logical vs. physical address of the stack

Calculating the physical address for the stack, the same principle is applied as was used for the code and data segments. Physical address depends on the value of stack segment (SS) register and the stack pointer (SP).

Ex: If SS=3500H and SP:FFFEH

a) Calculate the physical address: 35000+FFFE = 44FFE

b) Calculate the lower range of the stack: 35000+0000 = 35000

c) Calculate the upper range of the stack segment: 35000+FFFF = 44FFF

d) Show the logical address of the stack: 3500:FFFE

80X86 ADDRESSING MODES

The CPU can access operands (data) in various ways, called addressing modes. In 80x86 there are 7 addressing modes:

1. Register
2. Immediate
3. Direct
4. register indirect
5. based relative
6. indexed relative
7. based indexed relative

80X86 ADDRESSING MODES

1. Register addressing mode:

- ❖ involves the use of registers
- ❖ memory is not accessed, so faster
- ❖ source and destination registers must match in size.

Ex: MOV BX,DX

MOV ES,AX

ADD AL,BH

MOV AL,CX ;   not possible

80X86 ADDRESSING MODES

Assembly Language	SIZE	Operation
MOV AL , BL	8 – bits	Copies AL into BL
MOV CH , CL	8 – bits	Copies CL into CH
MOV AX , CX	16 – bits	Copies CX into AX
MOV SP , BP	16 – bits	Copies BP into SP
MOV DS , AX	16 – bits	Copies AX into DS
MOV SI , DI	16 – bits	Copies DI into SI
MOV BX , EX	16 – bits	Copies ES into BX
MOV CX , BX	16 – bits	Copies BX into CX
MOV SP , DX	16 – bits	Copies DX into SP
MOV ES , DS		Not allowed (segment-to-segment)
MOV BL , DX		Not allowed (mixed sizes)
MOV CS , AX		Not allowed (the code segment register may not be the destination register)

80X86 ADDRESSING MODES

2. Immediate addressing mode:

❖Source operand is a constant

❖Possible in all registers except segment and flag registers.

Ex: MOV **BX**,1234H ;   move 1234H into BX

MOV CX,223 ;   load the decimal value 223 into CX

ADD AL,40H ;   AL=AL+40H

MOV DS,1234H ;   illegal (Why!!!)

Not allowed (the segment register may not be the destination register)

80X86 ADDRESSING MODES

Assembly Language	SIZE	Operation
MOV BL , 44	8 – bits	Copies a 44 decimal into BL
MOV AX , 44H	16 – bits	Copies a 0044H into AX
MOV SI , 0	16 – bits	Copies a 0000H into SI
MOV CH , 100	8 – bits	Copies a 100 decimal into CH
MOV AL , ‘A’	8 – bits	Copies an ASCII A into AL
MOV AX , ‘AB’	16 – bits	Copies an ASCII BA into AX
MOV CL , 11001110B	8 – bits	Copies a 11001110 binary into CL

80X86 ADDRESSING MODES

3. Direct addressing mode:

address of the data in memory comes immediately after the instruction operand is a constant The address is the offset address. The offset address is put in a rectangular bracket

Ex: MOV DL,[2400] ; move contents of DS:2400H into DL

80X86 ADDRESSING MODES

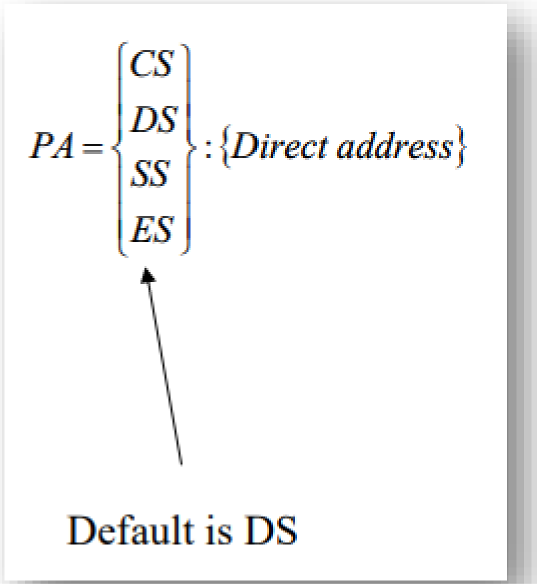
Ex: Find the physical address of the memory location and its content after the execution of the following operation. Assume DS=1512H

MOV AL,99H

MOV [3518],AL

Physical address of DS:3518 => 15120+3518=18638H

The memory location 18638H will contain the value 99H


$$PA = \begin{Bmatrix} CS \\ DS \\ SS \\ ES \end{Bmatrix} : \{Direct\ address\}$$

80X86 ADDRESSING MODES

4. Register indirect addressing mode:

- ❖ The address of the memory location where the operand resides is held by a register.
- ❖ SI, DI and BX registers are used as the pointers to hold the offset addresses.
- ❖ They must be combined with DS to generate the 20-bit physical address

$$PA = \begin{Bmatrix} CS \\ DS \\ SS \\ ES \end{Bmatrix} : \begin{Bmatrix} BX \\ BP \\ SI \\ DI \end{Bmatrix}$$

Default is DS

Ex: MOV AL,[BX] ; moves into AL the contents of the memory location

pointed to by DS:BX

Ex: MOV CL,[SI] ; move contents of DS:SI into CL

MOV [DI],AH ; move the contents of AH into DS:DI

80X86 ADDRESSING MODES

5. Based relative addressing mode:

❖BX and BP are known as the base registers. In this mode base registers as well as a displacement value are used to calculate the ***effective address*.**

❖The default segments used for the calculation of Physical address (PA) are DS for BX, and SS for BP.

80X86 ADDRESSING MODES

Ex: MOV CX,[BX]+10 ; move DS:BX+10 and DS:BX+11 into CX

; PA = DS (shifted left) +BX+10

- Note that, the content of the low address will go into CL and the

high address contents will go into CH.

- There are alternative coding: MOV CX,[BX+10],

  MOV CX,10[BX]

- BX+10 is *effective address*

Ex: MOV AL,[BP]+5 ; PA = SS (shifted left) +BP+5

80X86 ADDRESSING MODES

6. Indexed relative addressing mode:

❖Indexed relative addressing mode works the same as the based relative addressing mode.

❖Except the registers DI and SI holds the offset address.

Ex: MOV DX,[SI]+5 ;PA=DS(shifted left)+SI+5

MOV CL,[DI]+20 ;PA=DS(shifted left)+DI+20

$$PA = \begin{Bmatrix} CS \\ DS \\ SS \\ ES \end{Bmatrix} : \begin{Bmatrix} SI \\ DI \end{Bmatrix} + \begin{Bmatrix} 8-bit\ displacement \\ 16-bit\ displacement \end{Bmatrix}$$

80X86 ADDRESSING MODES

7. Based Indexed addressing mode:

Combining the based addressing mode and the indexed addressing mode results in a new, mere powerful mode known as the based-indexed addressing mode. This addressing mode can be used to access complex data structures such as two-dimensional arrays. (One base register and one index register are used).

$$PA = \left\{\begin{matrix} CS \\ DS \\ SS \\ ES \end{matrix}\right\} : \left\{\begin{matrix} BX \\ BP \end{matrix}\right\} + \left\{\begin{matrix} SI \\ DI \end{matrix}\right\} + \left\{\begin{matrix} 8-bit\ displacement \\ 16-bit\ displacement \end{matrix}\right\}$$

80X86 ADDRESSING MODES

Ex: MOV CL,[BX][DI]+8 ; PA=DS(shifted left)+BX+DI+8

MOV CH,[BX][SI]+20 ; PA=DS(shifted left)+BX+SI+20

MOV AH,[BP][DI]+12 ; PA=SS(shifted left)+BP+DI+12

MOV AL,[BP][SI]+29 ; PA=SS(shifted left)+BP+SI+29

Alternative coding

MOV CL,[BX+DI+8]

MOV CL,[DI+BX+8]

Computer Science Dept.

THANK YOU

By:

Ali Saadoon AHMED