CMPE 310
Systems Design and Programming

L18: Chapter 3 – *Addressing Modes (Basic)*

UMBC
AN HONORS UNIVERSITY IN MARYLAND

L18 Objectives

* Addressing modes
  * Recognize the addressing modes of the x86

Addressing Modes

* Instructions perform the operation they specify on elements of data that are called its operand
* Types of operands
  * Source operand
  * Destination operand
  * Content of source operand combined with content of destination operand → Result saved in destination operand location
* Operands may be
  * Part of the instruction—source operand only
  * Held in one of the internal registers—both source and destination operands
  * Stored at an address in memory—either the source or destination operand
  * Held in an input/output port—either the source or destination operand

Addressing Modes

* Types of addressing modes
  * Register addressing modes
  * Immediate operand addressing
  * Memory operand addressing
* Each operand can use a different addressing mode

Register Operand Addressing Mode

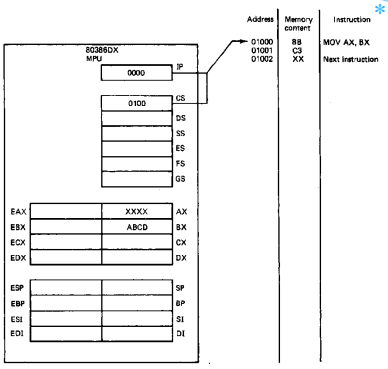
Register	Operand size		
	Byte (Reg8)	Word (Reg16)	Double word (Reg32)
Accumulator	AL, AH	AX	EAX
Base	BL, BH	BX	EBX
Count	CL, CH	CX	ECX
Data	DL, DH	DX	EDX
Stack pointer	–	SP	ESP
Base pointer	–	BP	EBP
Source index	–	SI	ESI
Destination index	–	DI	EDI
Code segment	–	CS	–
Data segment	–	DS	–
Stack segment	–	SS	–
E data segment	–	ES	–
F data segment	–	FS	–
G data segment	–	GS	–

* Register addressing mode operands
  * Source operand and destination operands are both held in internal registers of the x86/88
  * Only the data registers can be accessed as bytes, words, or double words
* Ex. AL,AH → bytes
* AX → word
* EAX →double word
* Index and pointer registers as words or double words

  Ex. SI →word pointer
  ESI → double word pointer
* Segment registers only as words

  Ex. DS → word pointer

Register Operand Addressing Mode

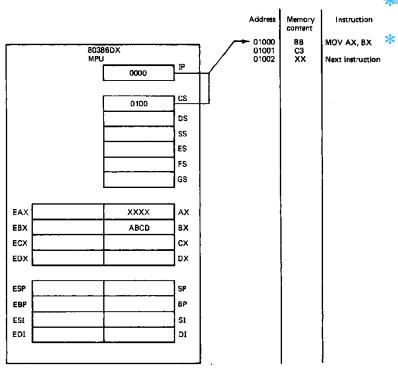


* Example

  MOV AX,BX

Register Operand Addressing Mode

* Example (continued)
* State after execution

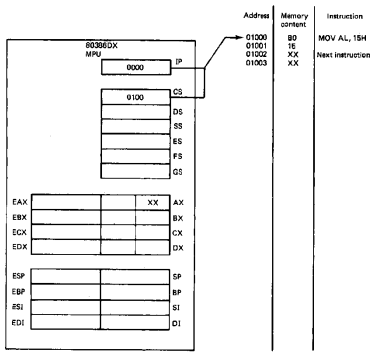


Immediate Operand Addressing Mode

Opcode	Immediate operand

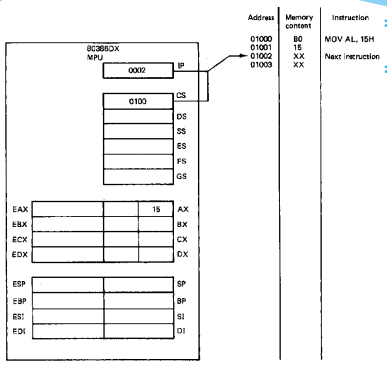
* Immediate operand
  * Operand is coded as part of the instruction
  * Applies only to the source operand
  * Destination operand uses register addressing mode or a memory addressing mode
* Types
  * Imm8 = 8-bit immediate operand
  * Imm16 = 16-bit immediate operand
  * Imm32 = 32-bit immediate operand
* General instruction structure and operation
* MOV Rx,ImmX
* ImmX → (Rx)

Immediate Operand Addressing Mode



* Example
  MOV AL,15H

Immediate Operand Addressing Mode



* Example (continued)
* State after execution

16-bit Memory Operand Addressing Modes

PA = SBA : EA

PA = Segment base : Base + Index + Displacement

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

* Accessing operands in memory
  * Only one operand can reside in memory—either the source or destination
  * Calculate the 20-bit physical address (PA) at which the operand in stored in memory
  * Perform a read or write to this memory location
  * 16-bit memory addressing modes produce 8088/8086/80286 compatible code

16-bit Memory Operand Addressing Modes

PA = SBA : EA

PA = Segment base : Base + Index + Displacement

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

* Physical address computation
  * Given in general as
    PA = SBA:EA
    SBA = Segment base address
    EA = Effective address
* Components of a effective address
  * Base → base registers BX or BP
  * Index → index register SI or DI
  * Displacement → 8 or 16-bit displacement
  * Not all elements are used in all computations—results in a variety of addressing modes

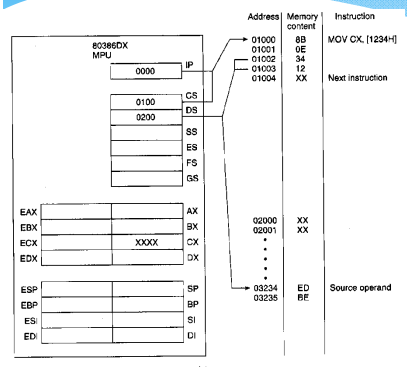
Direct Addressing Mode

PA = Segment base:Direct address

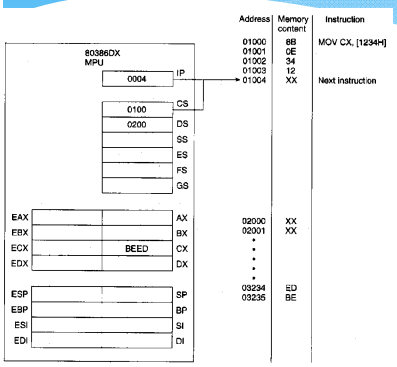
PA = { CS, DS, SS, ES, FS, GS } : { Direct address }

* Direct addressing mode
  * Similar to immediate addressing in that information coded directly into the instruction
  * Immediate information is the effective address—called the direct address
* Physical address computation
  - PA = SBA:EA → 20-bit address
  - PA = SBA:[DA] → immediate 8-bit or 16 bit displacement
* Segment base address is DS by default
  - PA = DS:[DA]
* Segment override prefix (SEG) is required to enable use of another segment register
  - PA = ES:[DA]

Direct Addressing Mode (Example: MOV CX, [1234H])



Direct Addressing Mode (Example: MOV CX, [1234H])



* State after execution

Next time

* Register indirect
* Base-plus-index

STOP