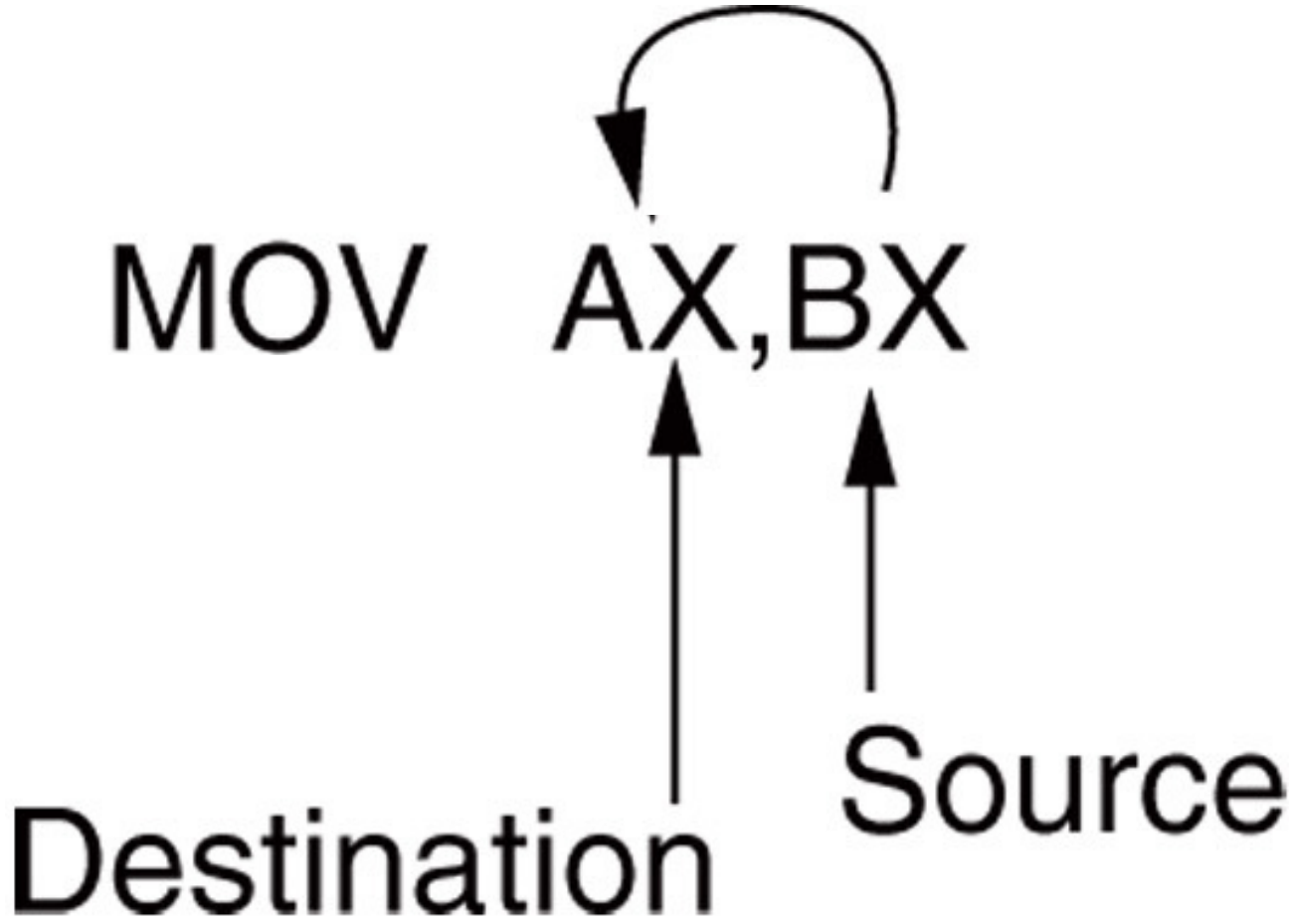
ADDRESSING MODES/8086

Mrs. JYOTSNA A. KULKARNI

DATA ADDRESSING MODES



Variations of data-addressing modes using MOV.

None of the MOV instruction affects FLAG register

- Register Addressing
- Immediate Addressing
- Direct Addressing
- Register Indirect Addressing
- Base-plus-Index Addressing
- Register relative Addressing
- Base relative-plus-index Addressing
- Scaled index Addressing

Register Addressing

MOV AX,BX

- Act of moving data changes only the destination register, never the source register.
- 8-bit registers: AH, AL, BH, BL, CH, CL, DH, and DL.
- 16-bit registers: AX, BX, CX, DX, SP, BP, SI, and DI.
- Important for instructions to use registers that are the same size.
  - *never* mix an 8-bit \with a 16-bit register, an 8- or a 16-bit register with a 32-bit register
  - this is not allowed by the microprocessor and results in an error when assembled

Check validity of following instructions

- MOV CL,DH -Valid
- MOV CS, AX - NOT ALLOWED

(Causes Problem)

- MOV DS,CS - NOT ALLOWED
- MOV SP,BP -Valid

NOTE

- Data transferred from a register : variable data
- Immediate data are constant data

- Register Addressing
- Immediate Addressing
- Direct Addressing
- Register Indirect Addressing
- Base-plus-Index Addressing
- Register relative Addressing
- Base relative-plus-index Addressing
- Scaled index Addressing

Immediate Addressing

MOV AX,44H

- Term *immediate* implies that data immediately follow the hexadecimal opcode in the memory.
- Immediate addressing operates upon a byte or word of data.

Immediate Data in symbolic assembly Language

- The letter **H** appends hexadecimal data.
  - MOV AX,44H
- If hexadecimal data begin with a letter, the assembler requires the data start with a **0**.
  - to represent a hexadecimal F2, 0F2H is used
- Decimal data are represented as is and require no special codes or adjustments
  - MOV AL,44

Immediate Data in symbolic assembly Language

- An ASCII-coded character or characters may be depicted in the immediate form if the ASCII data are enclosed in apostrophes.
  - be careful to use the apostrophe (‘) for ASCII data and not the single quotation mark (‘)
  - MOV AX,`AB`
- Binary data are represented if the binary number is followed by the letter B.
  - MOV CL,11001110B

Check validity of following instructions

- MOV SI,0
- MOV AH,1
- MOV AL,`A`
- MOV CH,100

- Register Addressing
- Immediate Addressing
- Direct Addressing
- Register Indirect Addressing
- Base-plus-Index Addressing
- Register relative Addressing
- Base relative-plus-index Addressing
- Scaled index Addressing

Direct Data Addressing

Two basic forms of direct data addressing:

- Direct addressing

  between a memory location and AL & AX.

- Displacement addressing

  Address is formed by adding the displacement to the default data segment address or an alternate segment address.

Direct Addressing

- MOV AX,[1234H]
- This instruction transfers a copy contents of memory location 11234H into AX.
  - the effective address is formed by adding 1234H (the offset address) and 10000H (the data segment address of 1000H times 10H) in a system operating in the real mode

Direct Addressing

- Direct addressing with a MOV instruction transfers data between a memory location, located within the data segment.
- MOV AX,[1234H]
- MOV [1234H],AX

Displacement Addressing

- Almost identical to direct addressing, except the instruction is 4 bytes wide instead of 3.
- This type of direct data addressing is much more flexible because most instructions use it.

- Register Addressing
- Immediate Addressing
- Direct Addressing
- Register Indirect Addressing
- Base-plus-Index Addressing
- Register relative Addressing
- Base relative-plus-index Addressing
- Scaled index Addressing

Register Indirect Addressing

- Allows data to be addressed at any memory location through an offset address held in any of the following registers: BP, BX, DI, and SI.