

# Lab 6

## Arithmetic and Logic Instructions

### Second Group

أعداد: م.م. سمر أميل يوسف

# Second Group Instructions

## MUL, DIV

REG

Memory

**MUL** instructions affect these flags only: **CF, OF**

When result is over operand size these flags are set to **1**, when result fits in operand size these flags are set to **0**.

For **DIV** flags are undefined.

# Second Group Instructions

## MUL Source

- This instruction multiplies a byte from source with a byte in **AL** or a word from some source with a word in **AX**.
- When a byte from source is multiplied with content of AL, the result (product) will be put in **AX**.
- When a word from source is multiplied by AX, the result is put in **DX and AX**.

# MUL instruction

when operand is a **byte**:

$$AX = AL * \text{operand.}$$

when operand is a **word**:

$$(DX AX) = AX * \text{operand.}$$

# Second Group Instructions

MOV AL, 11110011B

MOV BH, 11110101B

MUL BH

RET

Q:  $Z = 5 * 3$  put the result in cx?

## Second Group Instructions

MOV AX, 4444H

MOV CX, 2222H

MUL CX ; Multiply AX with CX result high word in DX, low word in AX

RET

MOV [4444H],22H

MOV AL, 22H

MUL [4444H]

RET

# Second Group Instructions

## DIV Source

This instruction is used to divide an word by a byte or to divide an double word (32 bits) by a word.

$AX = AX / \text{Byte}$

$DXAX = AX / \text{Word}$

# DIV instruction

**DIV** - divide:

when operand is a **byte**:

$AL = AX / \text{operand}$

$AH = \text{remainder (modulus)} . .$

when operand is a **word**:

$AX = (DX AX) / \text{operand}$

$DX = \text{remainder (modulus)} . .$



# Second Group Instructions

MOV [2000h],02

MOV [2001h],06

ADD AL,[2000h]

ADD AL,[2001h]

MOV BL, 02

DIV BL

RET

# DIV instruction

Q: Write an assembly language program to compute  $Z=203 / 4$

```
MOV AX, 203 ; AX = 00CBh
MOV BL, 4
DIV BL ; AL = 50 (32h), AH = 3
RET
```

# Second Group Instructions

Q: Write an assembly language program to compute  $C=(5/9)*(F-32)$

```
MOV AX,05
MOV BX,09
DIV BX
MOV BX,0FH
MOV CX, 32H
SUB BX,CX
MUL BX
RET
```

# Second Group Instructions

Q: Write an assembly language program to compute  $Y=(5+3)* (2+1)$

```
MOV AL, 5  
ADD AL,3  
MOV BL,2  
ADD BL,1  
MUL BL  
RET
```

# IMUL /IDIV

- **MUL** Unsigned multiply: and **IMUL** Signed multiply:
- **DIV** Unsigned divide and **IDIV** Signed divide

```
mov al, 02
```

```
mov bl,-4
```

```
imul bl
```

```
ret
```

```
mov al, 08
```

```
mov bl,-2
```

```
idiv bl
```

```
ret
```