

COE 205 Computer Organization & Assembly Language – Spring 2005

Assignment 3: Basic Instructions and Addressing Modes

Professor: Muhamed Mudawar

Due Date: Tuesday, March 22, 2005

Q1. (5 pts) Consider a program that has the following data segment:

```
I      EQU      07Fh
J      DB       '1234'
K      EQU      250
L      DW       0FFh
```

Indicate whether the following are valid or invalid 8086 instructions. If valid, give the result of the operation. If invalid, give the reason.

1. MOV AL, I+1

Valid: AL = 80h

2. MOV AX, J

Invalid: different sizes, J is a byte while AX is a word

3. MOV AL, J[2]

Valid: AL='3'=33h

4. MOV AX, J*3

Invalid: J is a variable, assembler cannot do the multiplication

5. MOV BH, I*2

Valid: BH = 0FEh, I is a constant, assembler does multiplication

6. MOV I, L

Invalid: Destination cannot be a constant

7. MOV DS, I

Invalid: Cannot move constant into a Segment register

8. MOV L, WORD PTR J

Invalid: Both operands cannot be memory locations

9. MOV L, OFFSET J

Valid: L is now 0 assuming that OFFSET J is 0

10. INC [J+1]

Valid: J[1] was '2' and is now '3' = 33h

Q2. (10 pts) The initial content of some registers and memory locations is given below:

AX=FE01H BX=7FEDH CX=F1A4H DX=00FFH

SI =0010H DI =0020H DS=2000H

| <u>Memory Address (hex)</u> | <u>Contents (hex)</u> |
|-----------------------------|-----------------------|
| 2000: 0010 | FF |
| 0011 | 1A |
| 0012 | 01 |
| 0013 | 06 |
| 0014 | FE |
| 0015 | 50 |
| 0016 | 40 |

Show the content of the destination operand and the state of the flag bits (OF, SF, ZF, AF, PF, and CF) after the execution of the following instructions. Use the initial content of the registers and memory locations for the execution of each instruction.

1. ADD BX, CX

$BX = BX + CX = 7FEDh + F1A4h = 7191h$

OF=0, SF=0, ZF=0, AF=1, PF=0, CF=1

2. INC Byte PTR [DI-10]

$DI - 10 = 20h - 10 = 32 - 10 = 22 = 16h$

This will increment byte at address 2000:0016h from 40h to 41h

OF=0, SF=0, ZF=0, AF=0, PF=1, CF = unchanged

3. DEC Byte PTR 4[SI]

$SI + 4 = 0010h + 4 = 0014h$

This will decrement byte at address 2000:0014h from FEh to FDh

OF=0, SF=1, ZF=0, AF=0, PF=0, CF = unchanged

4. NEG Word PTR [BX-7FDBh]

$BX - 7FDBh = 7FEDh - 7FDBh = 12h$

This will negate word at address 2000:0012h from 0601h to F9FFh

OF=0, SF=1, ZF=0, AF=1, PF=1, CF=1

5. SUB AL, 2+[SI]

$SI + 2 = 0010h + 2 = 0012h$

$AL = AL - \text{byte at address } 2000:0012h = 01h - 01h = 0$

OF=0, SF=0, ZF=1, AF=0, PF=1, CF=0

Q3. (5 pts) Write an assembly language program that reads one hex digit with values from A to F and displays its decimal equivalent on the next line.

```
.model small
.stack 100
.data
    msg1    DB    'Enter a hex digit (A-F): $'
    msg2    DB    13,10,'Decimal equivalent: $'
.code
.startup
    ; Display first message
    lea DX, msg1
    mov AH, 9
    int 21h
    ; Read entered digit
    mov AH,1
    int 21h
    ; Subtract ASCII code of 'A' and add '0'
    add AL, '0' - 'A'
    mov BL, AL
    ; Display second message
    lea DX, msg2
    mov AH, 9
    int 21h
    ; Display '1'
    mov DL, '1'
    mov AH, 2
    int 21h
    ; Display second char
    mov DL, BL
    int 21h
.exit
end
```