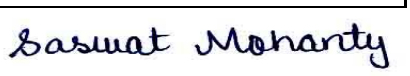
Computer Organization and Architecture (EET2211)

LAB II: Evaluate Different Arithmetic Operations on two 16 bit data

Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar

Branch: Computer Science and Engineering			**Section:** 'D'
S. No.	**Name**	**Registration No.**	**Signature**
52	Saswat Mohanty	1941012407	

Marks: _____/10

Remarks:

Teacher's Signature

I. OBJECTIVE:

1. Addition of two 16 bit numbers using direct addressing mode.
2. Subtraction of two 16 bit numbers using direct addressing mode.
3. Multiplication of two 16 bit numbers using direct addressing mode.
4. Division of two 16 bit numbers using direct addressing mode.

II. PRE-LAB

For Obj. 1:

a) Explain direct addressing mode briefly.

It is the addressing mode in which the effective address of the memory location is written directly in the instruction.

b) Examine & analyze the output obtained from addition of two 16 bit numbers.

mov ax,[1000h]

mov bx,[1002h]

add ax,bx

[1000h] = 1111h

[1002h] = 1234h

Output: 2345h

c) Write the assembly code.

```
org 100h
mov ax,0000h
mov ds,ax
mov ax,[5000h]
mov bx,[5002h]
add ax,bx
```

```
mov [5004h],ax
hlt
```

For Obj. 2:

a) Examine & analyze the output obtained from subtraction of two 16 bit numbers.

mov ax,[1000h]

mov bx,[1002h]

sub ax,bx

[1000h] = 2222h

[1002h] = 1111h

Output: 1111h

b) Write the assembly code.

```
org 100h
mov ax,0000h
mov ds,ax
mov ax,[3000h]
mov bx,[3002h]
sub ax,bx
mov [3004h],ax
hlt
```

For Obj. 3:

a) Examine & analyze the output obtained from multiplication of two 16 bit numbers.

mov ax,[1000h]

mov bx,[1002h]

mul bx,ax

[1000h] = 2222h

[1002h] = 1111h

Output: 2468642h

b) Write the assembly code.

```
org 100h
mov ax,0000h
mov ds,ax
mov ax,[3000h]
mov bx,[3002h]
mul bx
mov [3004h],ax
mov [3006h],dx
hlt
```

For Obj. 4:

a) Examine & analyze the output obtained from division of two 16 bit numbers.

mov ax,[1000h]

mov bx,[1002h]

div bx

[1000h] = 2222h

[1002h] = 1111h

Output: 2h

b) Write the assembly code.

```
org 100h
```

```
mov ax,0000h
mov ds,ax
mov ax,[3000h]
mov bx,[3002h]
div bx
mov [3004h],ax
mov [3006h],dx
hlt
ret
```

III. LAB:

Assembly Program:

For Obj. 1

```
; Saswat Mohanty
; 1941012407

; Addition  of two 16 bit numbers using direct addressing mode

org 100h

mov ax,0000h
mov ds,ax
mov ax,[5000h]      ;VALUE STORED AT 5000 = 1111
mov bx,[5002h]      ;VALUE STORED AT 5002 = 2222
add ax,bx
mov [5004h],ax
hlt

ret
```

For Obj. 2

```
; Saswat Mohanty
; 1941012407

; Subtraction of two 16 bit numbers using direct addressing mode

org 100h

mov ax,0000h
mov ds,ax
mov ax,[3000h]      ;VALUE STORED AT 3000 = 2222
mov bx,[3002h]      ;VALUE STORED AT 3002 = 1111
sub ax,bx
mov [3004h],ax
hlt

ret
```

For Obj. 3

```
; Saswat Mohanty
; 1941012407

; Multiplication of two 16 bit numbers using direct addressing mode

org 100h

mov ax,0000h
mov ds,ax
mov ax,[3000h]      ;VALUE STORED AT 3000 = 1111
mov bx,[3002h]      ;VALUE STORED AT 3002 = 2222
mul bx
mov [3004h],ax
mov [3006h],dx
hlt

ret
```

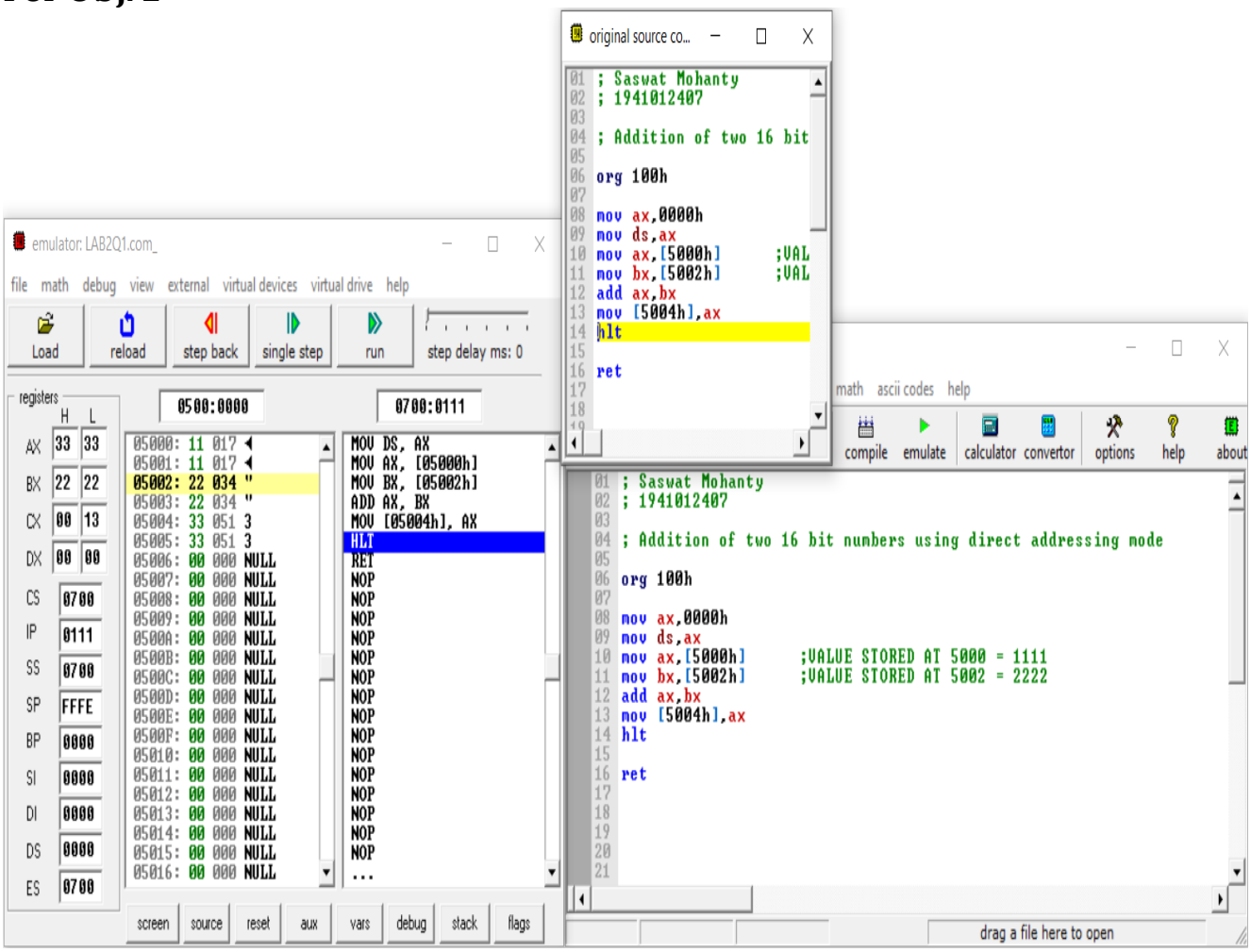
For Obj. 4

```
; Saswat Mohanty
; 1941012407

; Division of two 16 bit numbers using direct addressing mode
```

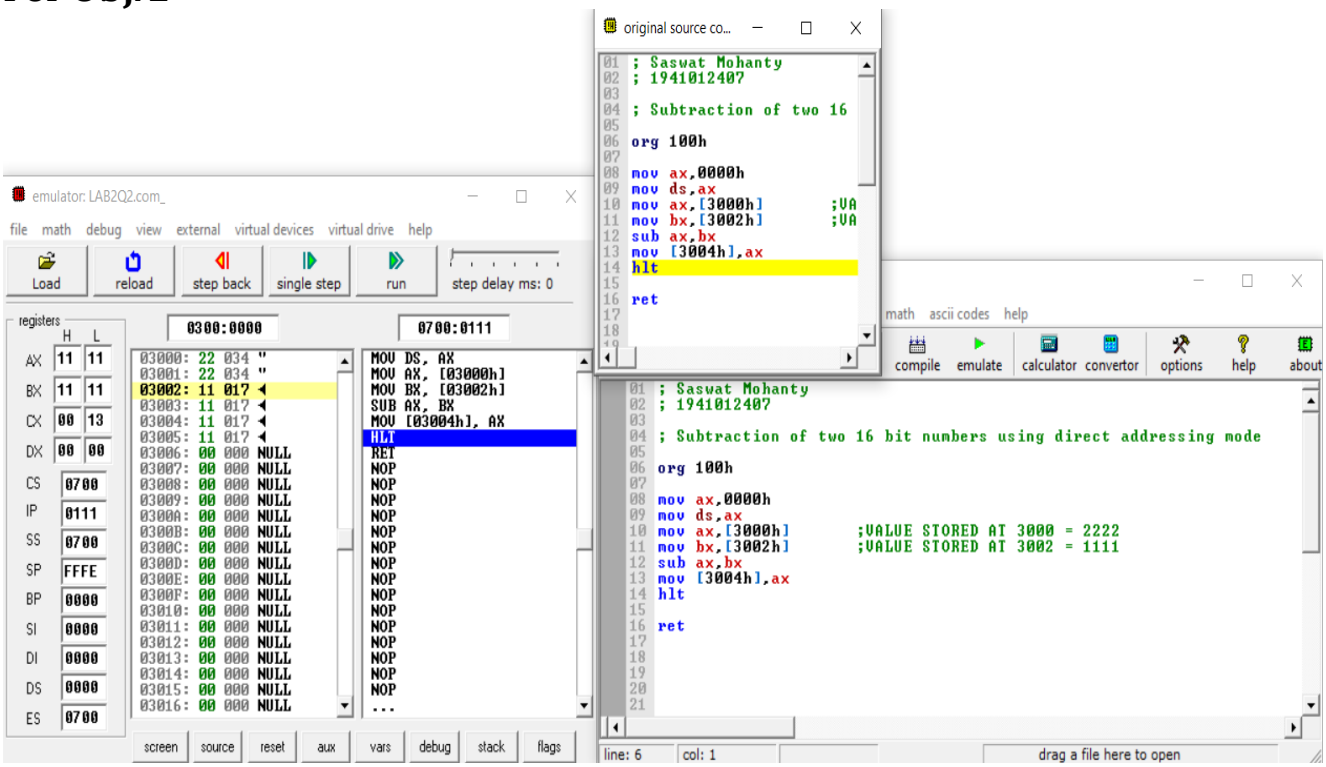
```
org 100h

mov ax,0000h
mov ds,ax
mov ax,[3000h]      ;VALUE STORED AT 3000 = 6666
mov bx,[3002h]      ;VALUE STORED AT 3002 = 2222
div bx
mov [3004h],ax
mov [3006h],dx
hlt

ret
```

Observations (with screen shots):

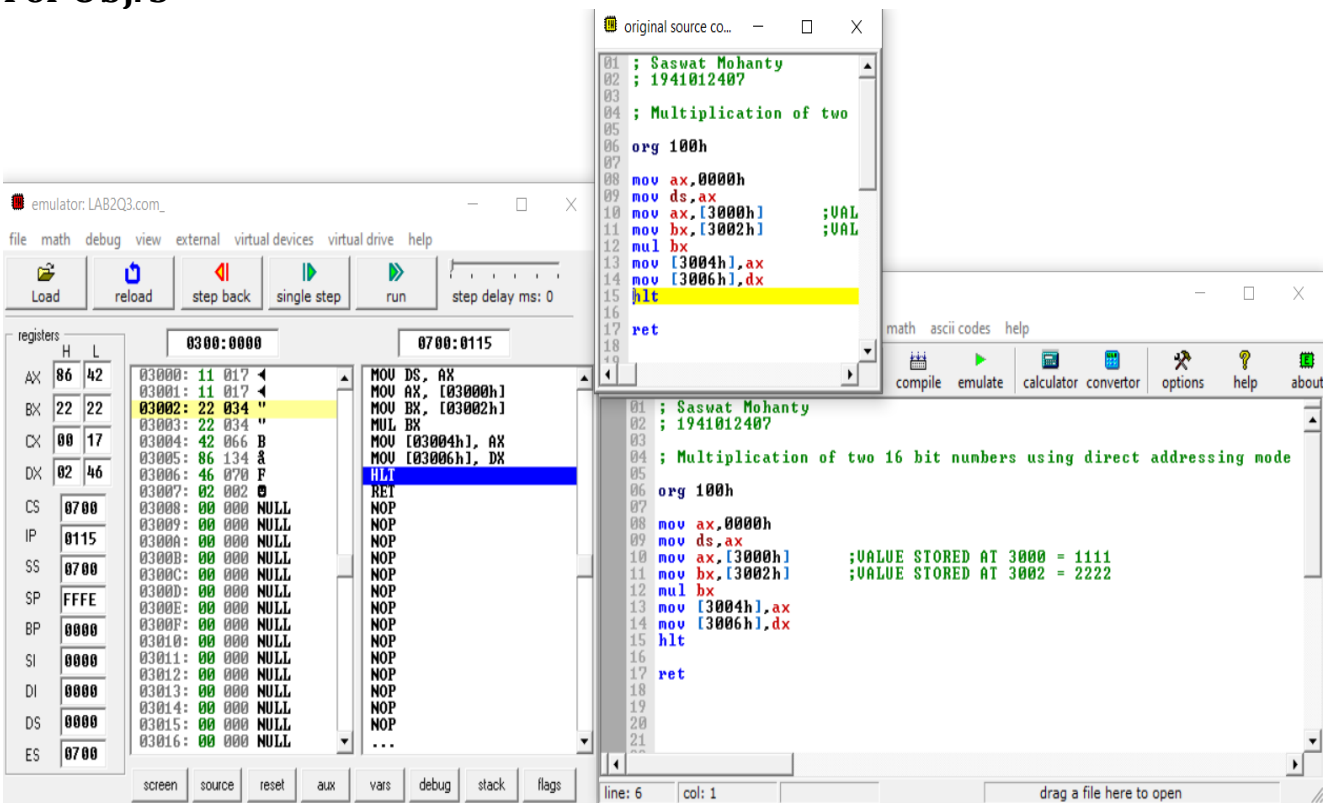
For Obj. 1



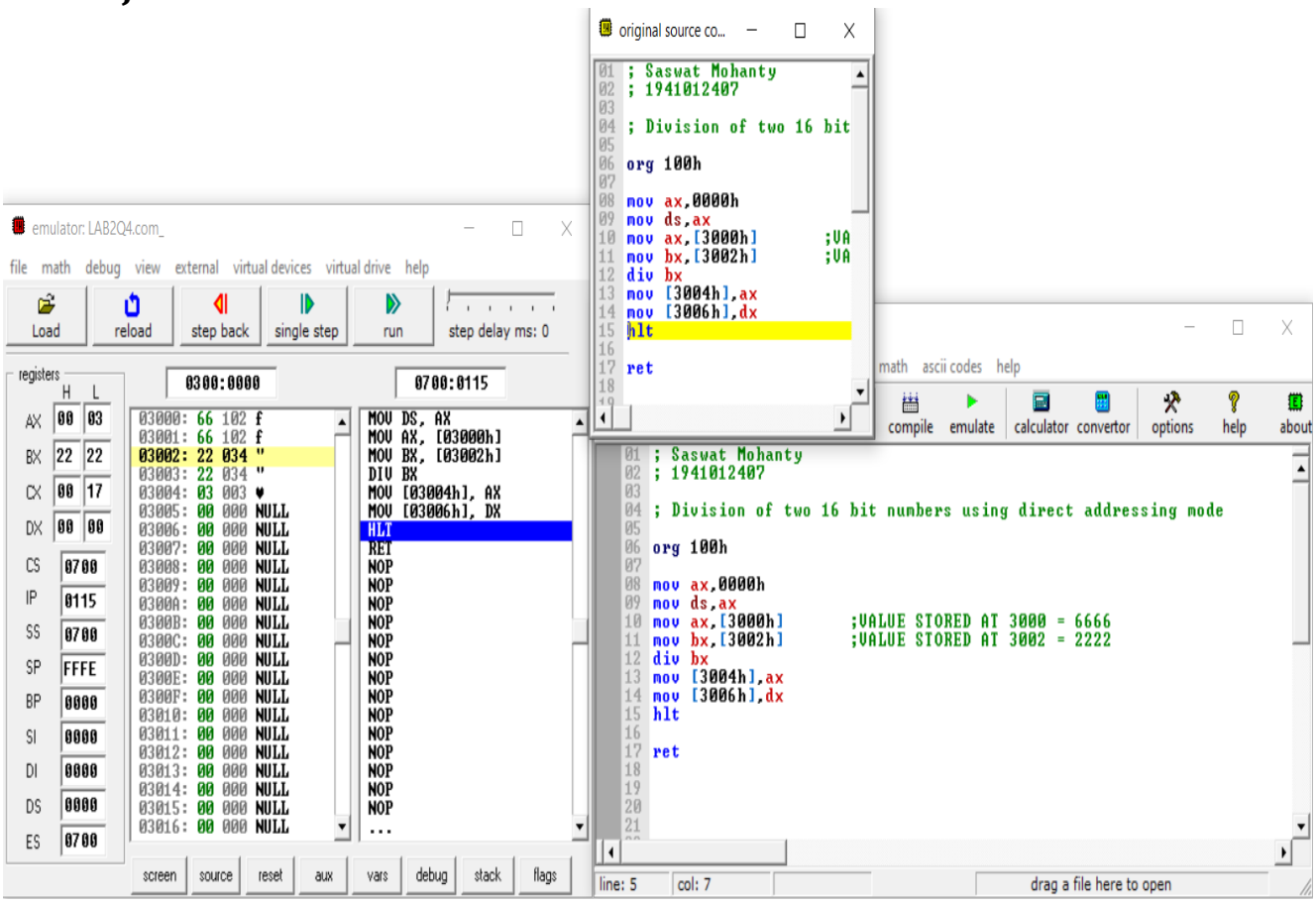
For Obj. 2



For Obj. 3



For Obj. 4



Conclusion:

For Obj. 1:

It can be concluded that the sum of numbers when dry run and executed in system found to be same. Thus, the program to add two 16-bit numbers was executed.

For Obj. 2:

It can be concluded that the difference of numbers when dry run and executed in system found to be same. Thus, the program to subtract two 16-bit numbers was executed.

For Obj. 3:

It can be concluded that the product of numbers when dry run and executed in system found to be same. Thus, the program to multiply two 16-bit numbers was executed.

For Obj. 4:

It can be concluded that the division of numbers when dry run and executed in system found to be same. Thus, the program to divide two 16-bit numbers was executed.

IV. POST LAB:

1. State and explain the different logical instructions of 8086.

Opcode	Operand	Description
AND	D,S	Used for adding each bit in a byte/word with the corresponding bit in another byte/word.
OR	D,S	Used to multiply each bit in a byte/word with the corresponding bit in another byte/word.
NOT	D	Used to invert each bit of a byte or word.
XOR	D,S	Used to perform Exclusive-OR operation over each bit in a byte/word with the corresponding bit in another byte/word.
TEST	D,S	Used to add operands to update flags, without affecting operands.

SHR	D,C	Used to shift bits of a byte/word towards the right and put zero(S) in MSBs.
SHL/SAL	D,C	Used to shift bits of a byte/word towards left and put zero(S) in LSBs.
ROR	D,C	Used to rotate bits of byte/word towards the right, i.e. LSB to MSB and to Carry Flag [CF].
ROL	D,C	Used to rotate bits of byte/word towards the left, i.e. MSB to LSB and to Carry Flag [CF].
RCR	D,C	Used to rotate bits of byte/word towards the right, i.e. LSB to CF and CF to MSB.
RCL	D,C	Used to rotate bits of byte/word towards the left, i.e. MSB to CF and CF to LSB.

2. Subtract two 16 bit numbers 20H and 06H, and store the difference.

-20

-06

=1A

3. Explain briefly any five arithmetic instructions.

- **ADD** − Used to add the provided byte to byte/word to word.
- **SUB** − Used to subtract the byte from byte/word from word.
- **MUL** − Used to multiply unsigned byte by byte/word by word.
- **DIV** − Used to divide the unsigned word by byte or unsigned double word by word.
- **INC** − Used to increment the provided byte/word by 1.

4. Write the function of the following machine control instructions

a) **WAIT** - Event Wait.

b) **HLT** - Halt CPU.

c) **NOP** - No Operation.

d) **ESC** - Escape.