Computer Organization and Architecture (EET2211)

LAB VI: Find 1's and 2's complement of a number

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	Saswat Mohanty

Marks: _____/10

Remarks:

Teacher's Signature

I. OBJECTIVE:

1. Write a program to find 1's and 2's complement of a given number without using logical instructions.

II. PRE-LAB

For Obj. 1:

a) Find 1's and 2's complement of a given number.

[3000h] = 2222h

Output: 1s Complement = EEEEh

2s Complement = EEEFh

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
inc ax
mov [3006h],ax
hlt
ret
```

III. LAB:

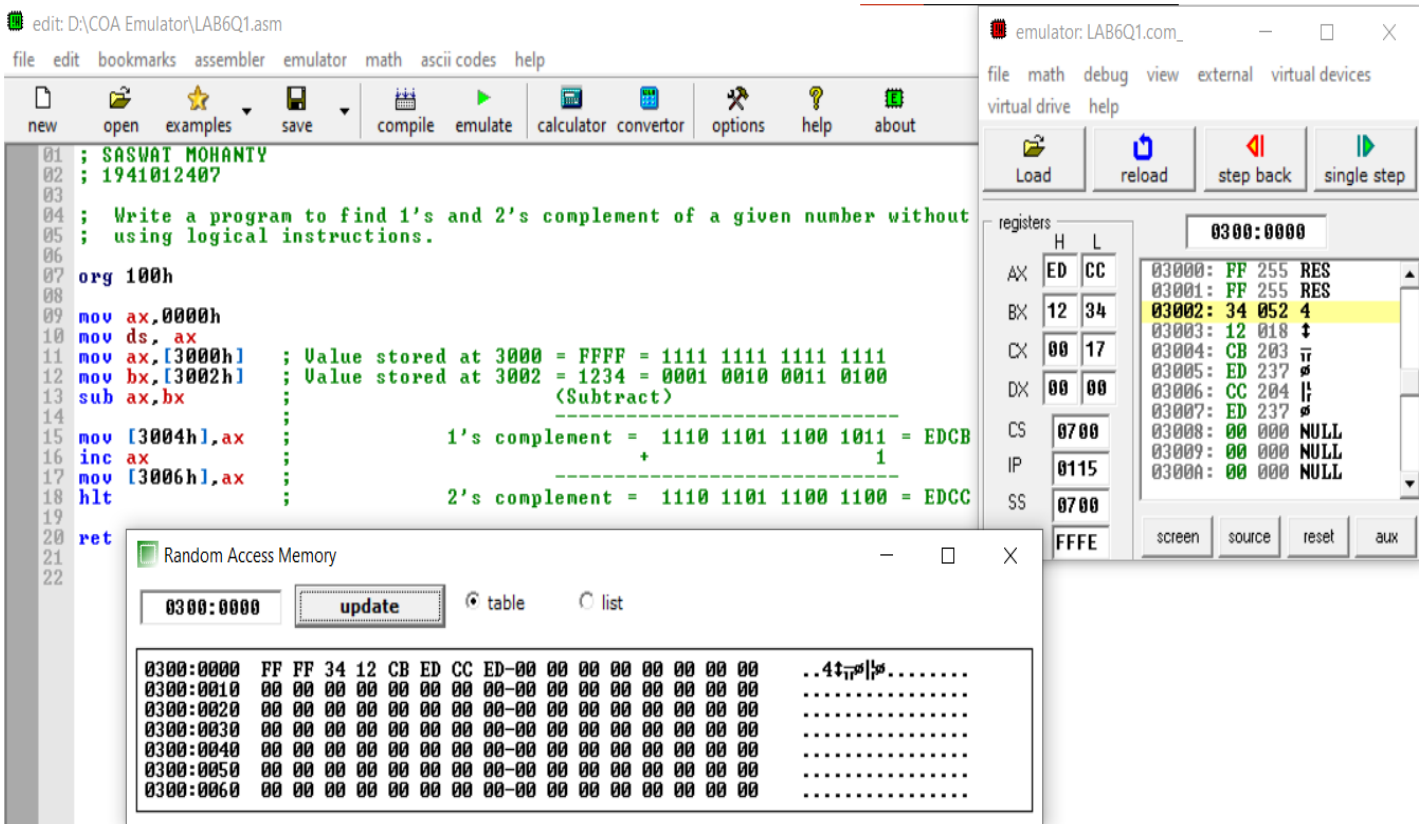
Assembly Program:

For Obj. 1:

```
; SASWAT MOHANTY
; 1941012407

;  Write a program to find 1's and 2's complement of a given number without
;  using logical instructions.

org 100h

mov ax,0000h
mov ds, ax
mov ax,[3000h]   ; Value stored at 3000 = FFFF = 1111 1111 1111 1111
mov bx,[3002h]   ; Value stored at 3002 = 1234 = 0001 0010 0011 0100
sub ax,bx        ;                              (Subtract)
                 ;                              ---------------------------------------------
mov [3004h],ax   ;                 1's complement =  1110 1101 1100 1011 = EDCB
inc ax           ;                                      +                              1
mov [3006h],ax   ;                                ---------------------------------------------
hlt              ;                 2's complement =  1110 1101 1100 1100 = EDCC

ret
```

Observations (with screen shots):

For Obj. 1:



Conclusion:

It can be concluded that 1's and 2's complement of a given number when dry run and executed in system found to be same. Thus, the program to find the 1's and 2's complement of a given number was executed.

IV. POST LAB:

1. Briefly explain the logical address, base segment address and physical address?

Logical address is contained in the 16-bit IP, BP, SP, BX, SI or DI. It is also known as the offset address or the effective address.

The base segment address is contained in one of the 16bit contents of the segment registers CS, DS, ES, SS.

The physical address or the real address is formed by combining the offset and base segment addresses. This address is 20bit and is primarily used for the accessing of the memory.

2. Differentiate between CISC and RISC.

CISC	RISC
Stands for Complex Instruction Set Computers	Stands for reduced Instruction Set Computers
A full set of computer instructions that intends to provide the necessary capabilities in an efficient way	An instruction set architecture that is designed to perform a smaller number of computer instructions so that it can operate at a higher speed
Instruction cycles can take several clock cycles to execute	Single cycle instructions execution takes place
Hardware centric design	Software centric design

3. Explain briefly the advantages of pipelining in 8086.

Advantages of pipelining:

The EU always reads the next instruction byte from the queue in BIU. This is much faster than sending out an address to the memory and waiting for the next instruction byte to come.

In short pipelining eliminates the waiting time of EU and speeds up the processing.

The 8086 BIU will not initiate a fetch unless and until there are two empty bytes in its queue.

4. Briefly explain the following:

a) Stack Pointer (SP) **b) Base Pointer (BP)**

c) Destination Index (DI) **d) Source Index (SI)**

a) **Stack Pointer (SP): -** The Stack Pointer (SP) register is used to indicate the location of the last item put onto the stack.

b) **Base Pointer (BP): -** The base pointer refers to the bottom of the stack, which normally refers to higher addresses as it grows towards lower.

c) **Destination Index (DI): -** The Destination Index register used as a pointer to the current character being written or compared in a string instruction.

d) **Source Index (SI): -** The Source Index register is used as source index for string operations.