CAO LAB EXP-2

UID:2019140042 Class: SE-IT

Batch: C

Aim: Implement various Arithmetic Operations through Assembly Language Programming for microprocessor 8086 (MASM).

Name:Mihir Nikam

## <u>Problem 1: Addition of 2 16-bit hexadecimal numbers</u>

## CODE:

data segment

a dw 1234h

b dw 5678h

c dw?

data ends

code segment

assume cs:code,ds:data

start:

mov ax,data

mov ds,ax //moving data into code segment

mov ax,a //moving the  $1^{st}$  number into ax register

mov bx,b //moving the 2<sup>nd</sup> number into bx register

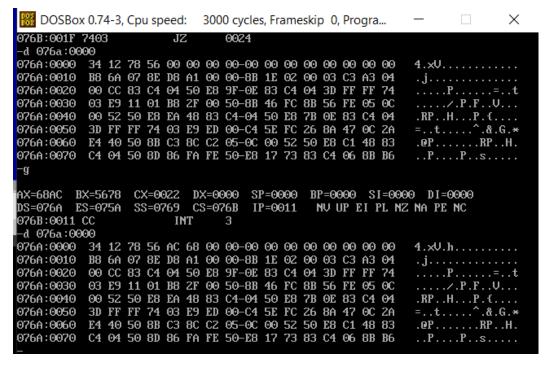
add ax,bx //adding the two numbers

mov c,ax //moving the addition into c

int 3

code ends

end start



Problem 2: Subtraction of 2 16-bit hexadecimal numbers

## CODE:

data segment

a dw 5678h

b dw 1234h

c dw?

data ends

code segment

assume cs:code,ds:data

start:

mov ax,data

mov ds,ax //moving data into code segment

mov ax,a //moving the 1st number into ax register

mov bx,b //moving the 2<sup>nd</sup> number into bx register

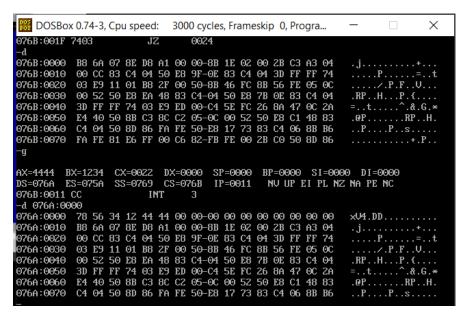
sub ax,bx //subtracting the two numbers

mov c,ax //moving the addition into c

int 3

code ends

end start



Problem 3: Multiplication of 2 numbers

## CODE:

end start

```
data segment
a dw 5h
b dw 4h
c dw?
data ends
code segment
assume cs:code,ds:data
start:
mov ax,data
              //moving data into code segment
mov ds,ax
              //moving the 1st number into ax register
mov ax,a
              //moving the 2<sup>nd</sup> number into bx register
mov bx,b
              //multiplying the two numbers
mul ax,bx
              //moving the addition into c
mov c,ax
int 3
code ends
```

```
BB DOSBox 0.74-3, Cpu speed:
                                   3000 cycles, Frameskip 0, Progra...
                                                                                              Х
076B:0008 8B1E0200
                                        BX,[0002]
076B:000C F7E0
076B:000E A30400
                             MUL
                                        ΑX
                             MOV
                                        [0004],AX
076B:0011 CC
                              INT
076B:0012 83C404
                                        SP,+04
                              ADD
076B:0015 50
                              PUSH
                                        ΑX
076B:0016 E89F0E
                                        OEB8
                             CALL
076B:0019 83C404
                             ADD
                                        SP,+04
076B:001C 3DFFFF
076B:001F 7403
                                        AX,FFFF
                             CMP
                              JZ
                                        0024
AX=0019 BX=0004 CX=0022 DX=0000
DS=076A ES=075A SS=0769 CS=076B
                                            SP=0000
                                                        BP=0000 SI=0000 DI=0000
                                             IP=0011
                                                         NU UP EI PL NZ NA PO NC
076B:0011 CC
                              INT
                                       3
-d 076A:0000
076A:0000 05 00 04 00 19 00 00 00-00 00 00 00 00 00 00
076A:0010 B8 6A 07 8E D8 A1 00 00-8B 1E 02 00 F7 E0 A3 04
                                                                            00 CC 83 C4 04 50 E8 9F-0E 83 C4 04 3D FF FF 74 03 E9 11 01 B8 2F 00 50-8B 46 FC 8B 56 FE 05 0C 00 52 50 E8 EA 48 83 C4-04 50 E8 7B 0E 83 C4 04
076A:0020
076A:0030
076A:0040
             3D FF FF 74 03 E9 ED 00-C4 5E FC 26 8A 47 OC 2A
076A:0050
             E4 40 50 8B C3 8C C2 05-0C 00 52 50 E8 C1 48 83 C4 04 50 8D 86 FA FE 50-E8 17 73 83 C4 06 8B B6
                                                                            .@P.....RP..H.
076A:0060
076A:0070
```

Result: Hence, we learn to perform different operations on hexadecimal numbers using 8086 microprocessor. Also, we learn to execute and compile the program using different commands.