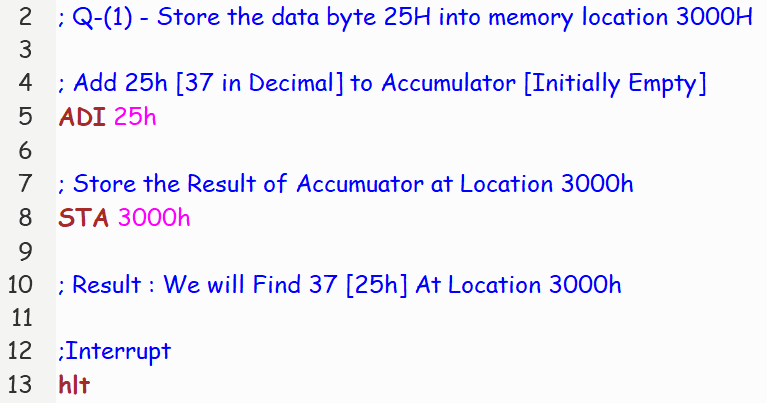
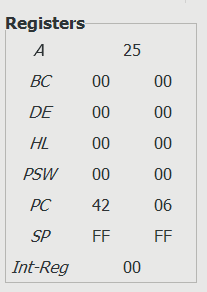
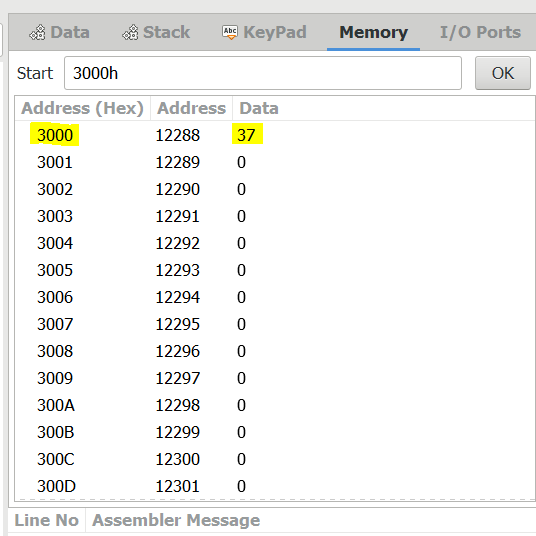
**M.I.T. LAB Assignment – 01**

**U19CS012**

(1) Store the data byte 25H into memory location 3000H

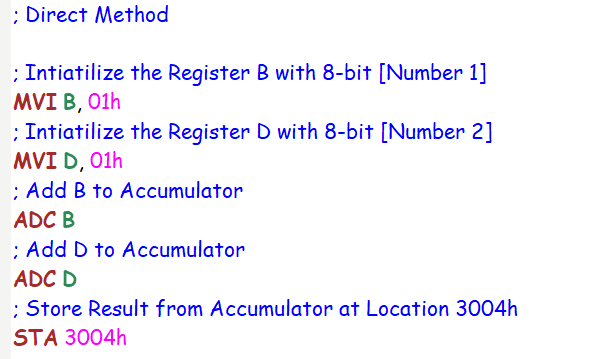
Notepad Code:

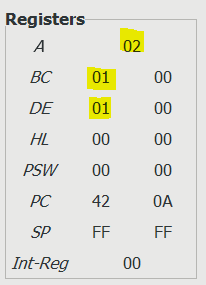
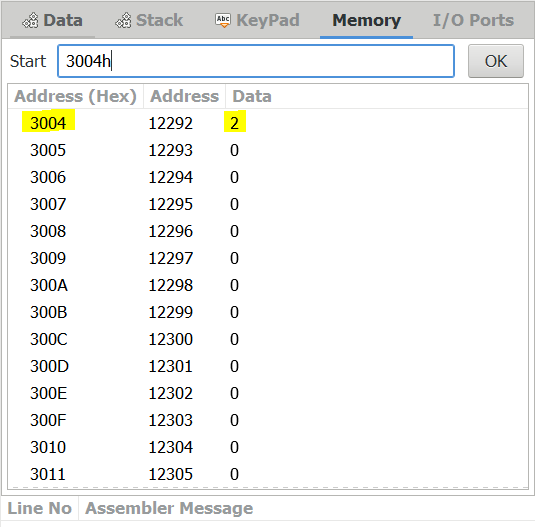


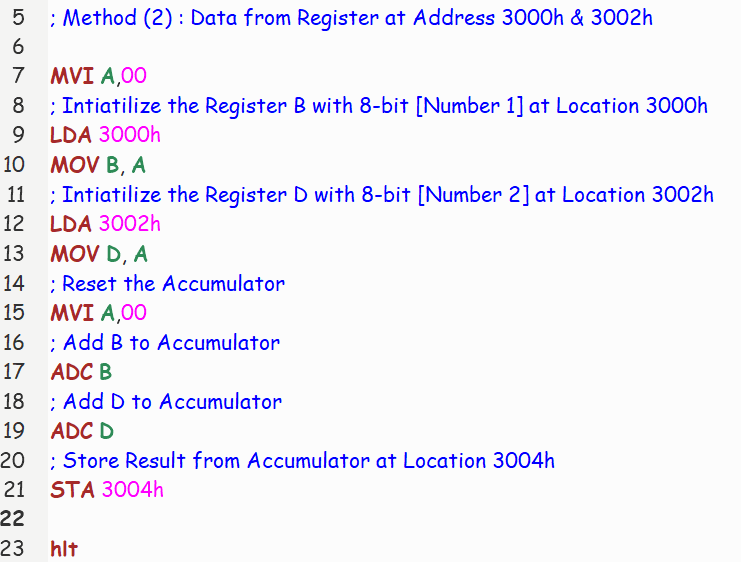
(2) Write a program to add two 8-bit numbers. Store the result at one memory location.

Notepad Code:

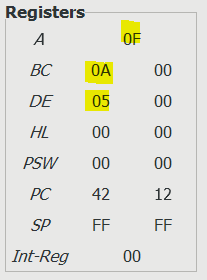
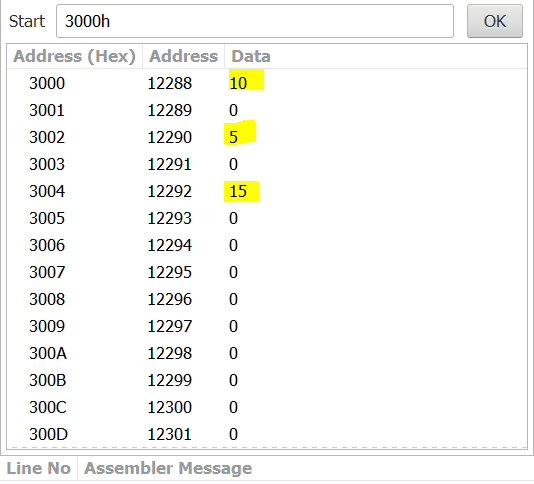


Notepad Code:

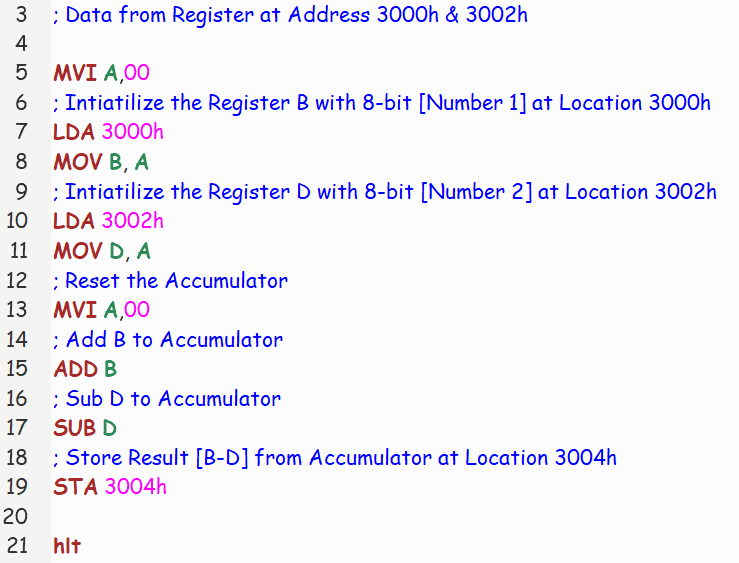


Let’s Input B = **10** [(0A)h] & D = **5** [(05)h] -> Output : **15** [(0F)h]

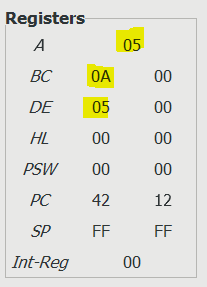
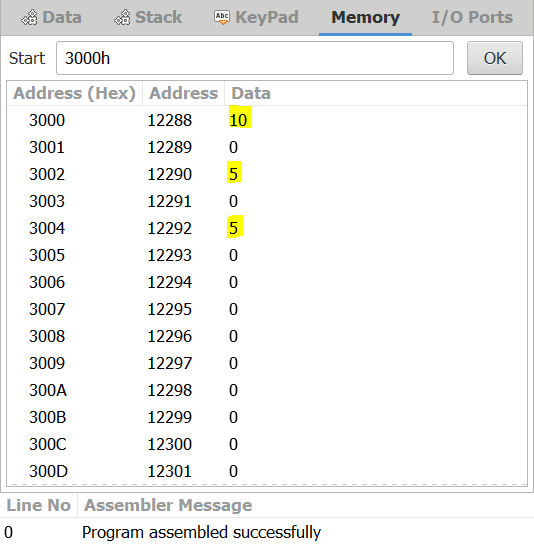
 

(3) Write a program to subtract two 8-bit numbers. Store the result at one memory location.

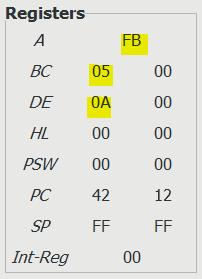
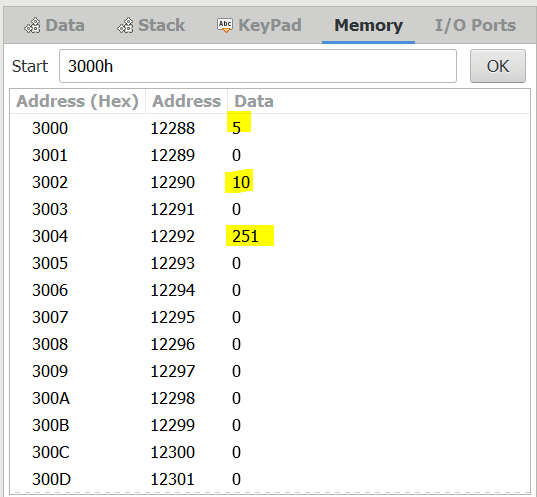
Notepad Code:



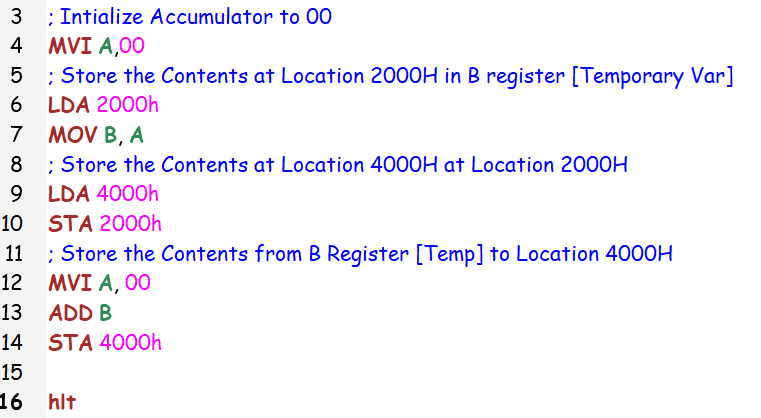
Let’s Input B = **10** [(0A)h] & D = **5** [(05)h] -> Output : **5** [(05)h]

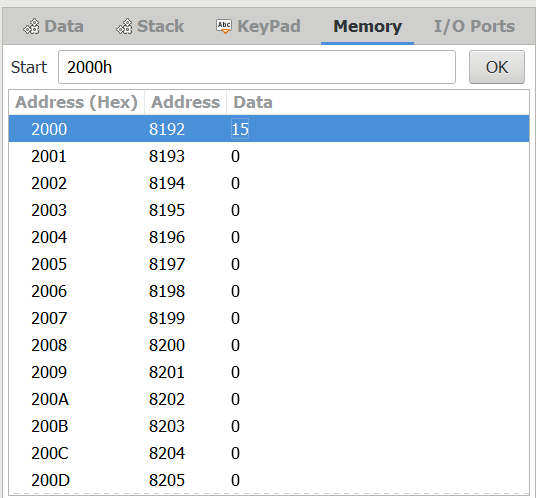
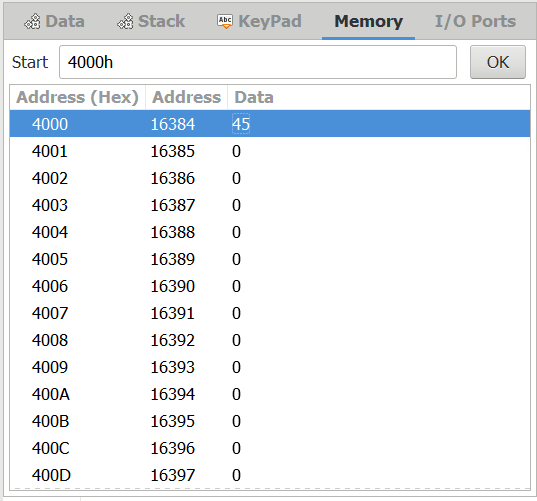
Let’s Input B = **5** [(05)h] & D = **10** [(0A)h] -> Output : **5-10 = -5 = -5+256 = 251** [(FB)h]

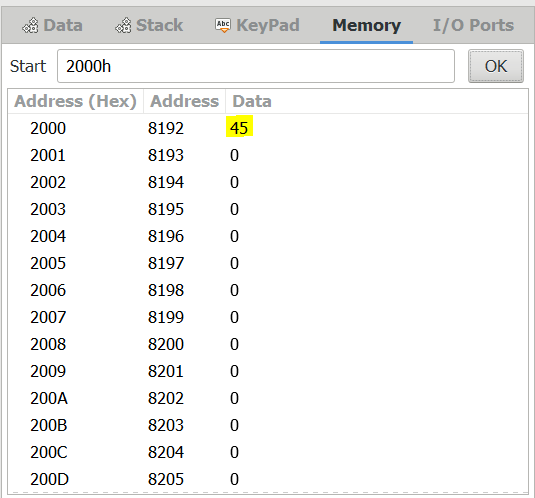
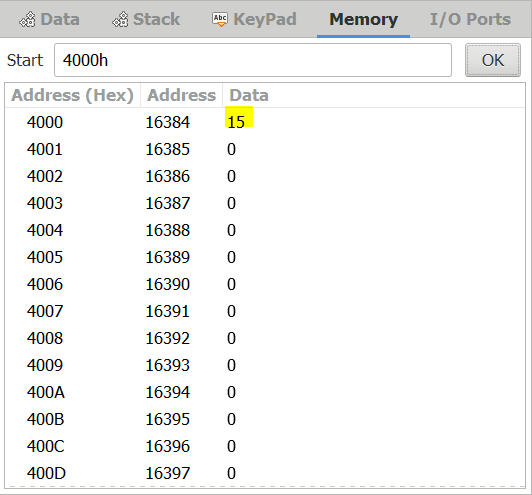
(4) Exchange the contents of memory locations 2000H and 4000H



INPUT: 2000h = 15 & 4000h = 45

OUTPUT: 2000h = 45 & 4000h = 15

(5) Write a program to add two 16 bit numbers. Numbers are stored in four consecutive memory location as 8-bit numbers. (Use instruction ADC)



Let’s Suppose

Number 1 = (msb) **10** 10 (lsb) [**4112** in Decimal]

Input 1: Addr(3000h) = 10h = 16 && Addr(3001h) = **10**h = 16

Number 2 = (msb) 01 10 (lsb) [**272** in Decimal]

Input 2: Addr(3002h) = 10h = 16 && Addr(3003h) = **01**h = 1

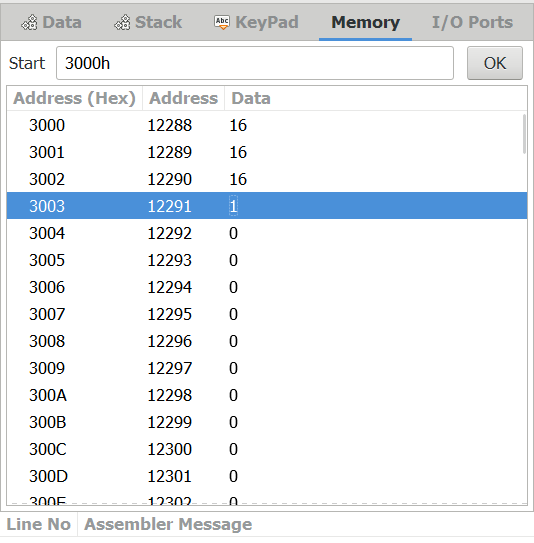
Expected Output = 4112 + 272 = **4384** = (msb) **11** 20 (lsb) [Hexadecimal]

**Expected Output:**

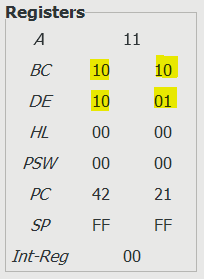
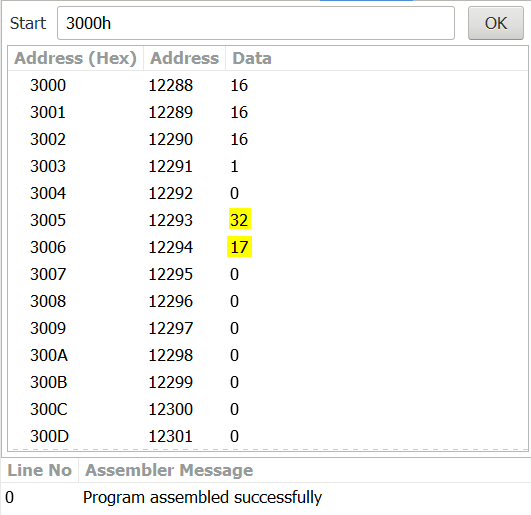
(LSB)Addr(3005h) = 20h = 32

(MSB)Addr(3006h) = **11h** = 17

INPUT :



OUTPUT:

(6) Write a program to subtract two 16 bit numbers. Numbers are stored in four consecutive memory location as 8-bit numbers. (Use instruction SBB)



Let’s Suppose

Number 1 = (msb) **10** 10 (lsb) [**4112** in Decimal]

Input 1: Addr(3000h) = 10h = 16 && Addr(3001h) = **10**h = 16

Number 2 = (msb) 01 10 (lsb) [**272** in Decimal]

Input 2: Addr(3002h) = 10h = 16 && Addr(3003h) = **01**h = 1

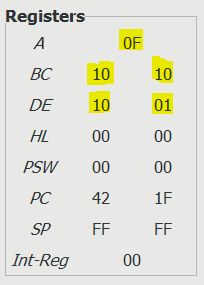
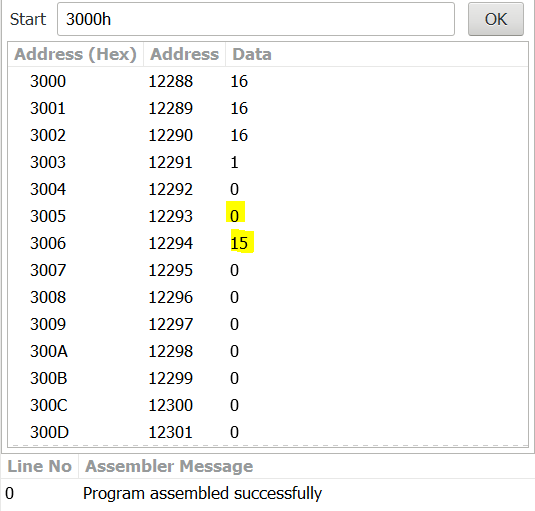
Expected Output = 4112 - 272 = **3840** = (msb) **0F** 00 (lsb) [Hexadecimal]

**Expected Output:**

(LSB)Addr(3005h) = 00h = 0

(MSB)Addr(3006h) = **0Fh** = 15

OUTPUT:

Let’s Suppose

Number 1 = (msb) **10** 10 (lsb) [**4112** in Decimal]

Input 1: Addr(3000h) = 10h = 16 && Addr(3001h) = **10**h = 16

Number 2 = (msb) 01 11 (lsb) [**273** in Decimal]

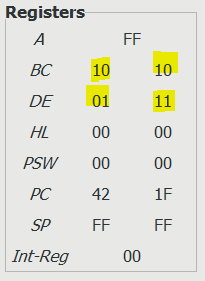
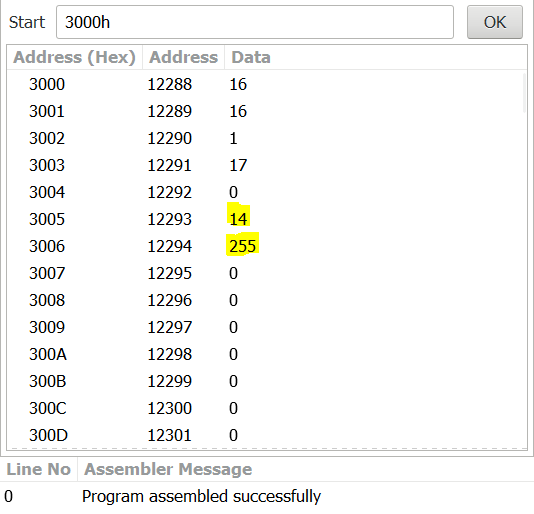
Input 2: Addr(3002h) = 10h = 1 && Addr(3003h) = **01**h = 17

Expected Output = 4112 - 273 = **3839** = (msb) **0E** FF (lsb) [Hexadecimal]

**Expected Output:**

(LSB)Addr(3005h) = 0Eh = 14

(MSB)Addr(3006h) = **FFh** = 255

**Submitted By:**

**BHAGYA VINOD RANA**

**U19CS012**