



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Computer Science & Engineering

CSE2006

Microprocessor and Interfacing

LAB ASSIGNMENT 1

Submitted to **Prof. SANJAY R**

TOPIC: INTRODUCTION TO EMU8086

NAME: PUNIT MIDDHA

REG.NO: 19BCE2060

SLOT: L43+L44

DATE: 09/09/2021

➤ Task – 1 (Assessment – 1)

Let AH = the first 4 digits of your Roll no, AL = the last digits of your Roll no.

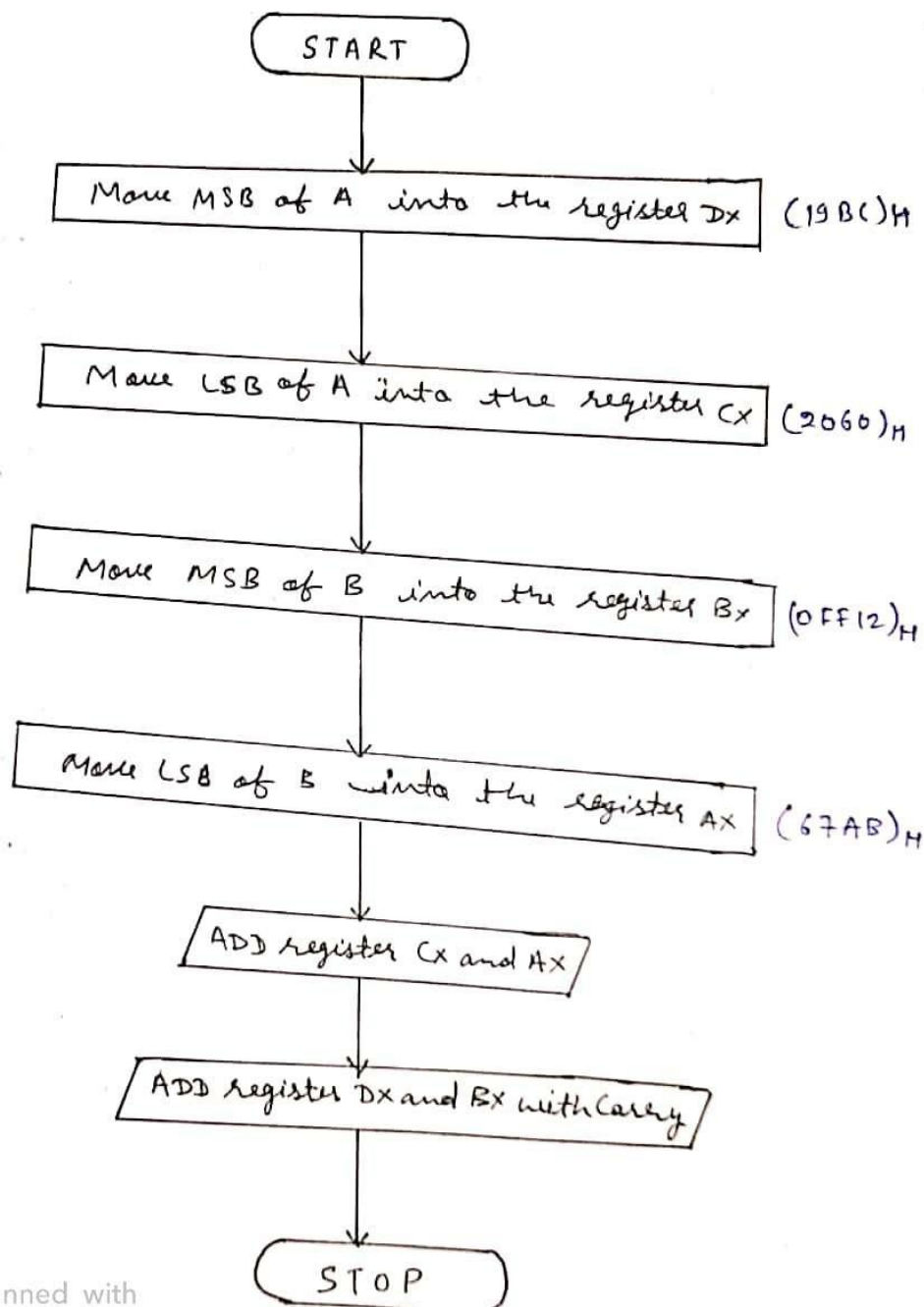
1) If B = FF1267AB, Find A + B.

Aim:

Find the addition of A and B where A=The first & last 4 digits of Register number i.e., (19BC2060)H and B=(FF1267AB)H.

Handwritten Flow chart:

1.



Handwritten Program:

1.

Addition of A & B ($A+B$) :

where

$A = (19BC2060)H$

$B = (FF1267AB)H$

CODE:-

org 100h

MOV DX, 19BCH

MOV CX, 2060H

MOV BX, 0FF12H

MOV AX, 67ABH

ADD CX, AX

ADC DX, BX

ret



Scanned with
CamScanner

Snapshots of typed program and Output:

```
edit: C:\Users\Punit Middha\Desktop\Micro\assign1_a.asm
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help
01
02 ; You may customize this and other start-up templates;
03 ; The location of this template is c:\emu8086\inc\0_com_template.txt
04
05 ; Name: PUNIT MIDDHA
06 ; RegNo: 19BCE2060
07
08 org 100h
09
10 MOV DX, 19BCH
11 MOV CX, 2060H
12 MOV BX, 0FF12H
13 MOV AX, 67ABH
14 ADD CX, AX
15 ADC DX, BX
16
17 ret
```

emulator: assign1_a.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

| registers | | H | L |
|-----------|------|----|---|
| AX | 67 | AB | |
| BX | FF | 12 | |
| CX | 88 | 0B | |
| DX | 18 | CE | |
| CS | F400 | | |
| IP | 0154 | | |
| SS | 0700 | | |
| SP | FFFA | | |
| BP | 0000 | | |
| SI | 0000 | | |
| DI | 0000 | | |
| DS | 0700 | | |
| ES | 0700 | | |

| F400:0154 | | F400:0154 | |
|-----------|-------------|------------------|--|
| F4150: | FF 255 RES | BIOS DI | |
| F4151: | FF 255 RES | INT 020h | |
| F4152: | CD 205 = | I RET | |
| F4153: | 20 032 SPA | ADD IBX + SI, AL | |
| F4154: | CF 207 ± | ADD IBX + SI, AL | |
| F4155: | 00 000 NULL | ADD IBX + SI, AL | |
| F4156: | 00 000 NULL | ADD IBX + SI, AL | |
| F4157: | 00 000 NULL | ADD IBX + SI, AL | |
| F4158: | 00 000 NULL | ADD IBX + SI, AL | |
| F4159: | 00 000 NULL | ADD IBX + SI, AL | |
| F415A: | 00 000 NULL | ADD BH, BH | |
| F415B: | 00 000 NULL | DEC BP | |
| F415C: | 00 000 NULL | SBB CL, BH | |
| F415D: | 00 000 NULL | ADD IBX + SI, AL | |
| F415E: | 00 000 NULL | ADD IBX + SI, AL | |
| F415F: | 00 000 NULL | ADD IBX + SI, AL | |
| F4160: | FF 255 RES | ADD IBX + SI, AL | |
| F4161: | FF 255 RES | ADD BH, BH | |
| F4162: | CD 205 = | DEC BP | |
| F4163: | 1A 026 → | ADD BH, CL | |
| F4164: | CF 207 ± | ADD IBX + SI, AL | |
| F4165: | 00 000 NULL | ADD IBX + SI, AL | |
| F4166: | 00 000 NULL | ADD IBX + SI, AL | |
| F4167: | 00 000 NULL | ADD IBX + SI, AL | |
| F4168: | 00 000 NULL | ADD BH, BH | |
| F4169: | 00 000 NULL | DEC BP | |
| F416A: | 00 000 NULL | ADD AL, 0CFh | |
| F416B: | 00 000 NULL | ADD IBX + SI, AL | |
| F416C: | 00 000 NULL | ADD IBX + SI, AL | |
| F416D: | 00 000 NULL | ADD IBX + SI, AL | |
| F416E: | 00 000 NULL | ADD IBX + SI, AL | |
| F416F: | 00 000 NULL | ADD IBX + SI, AL | |
| F4170: | FF 255 RES | ... | |

| flags | |
|-------|---|
| CF | 1 |
| ZF | 0 |
| SF | 0 |
| OF | 0 |
| PF | 0 |
| AF | 0 |
| IF | 0 |
| DF | 0 |

analyse

screen source reset aux vars debug stack flags

Inference:

- The result is stored in DX and CX. Msb bits are stored in DX and lsb bits in CX. Also the carry flag is 1, so there is also a carry bit.
- Hence, $A+B = \mathbf{1\ 18CE\ 880B}$.

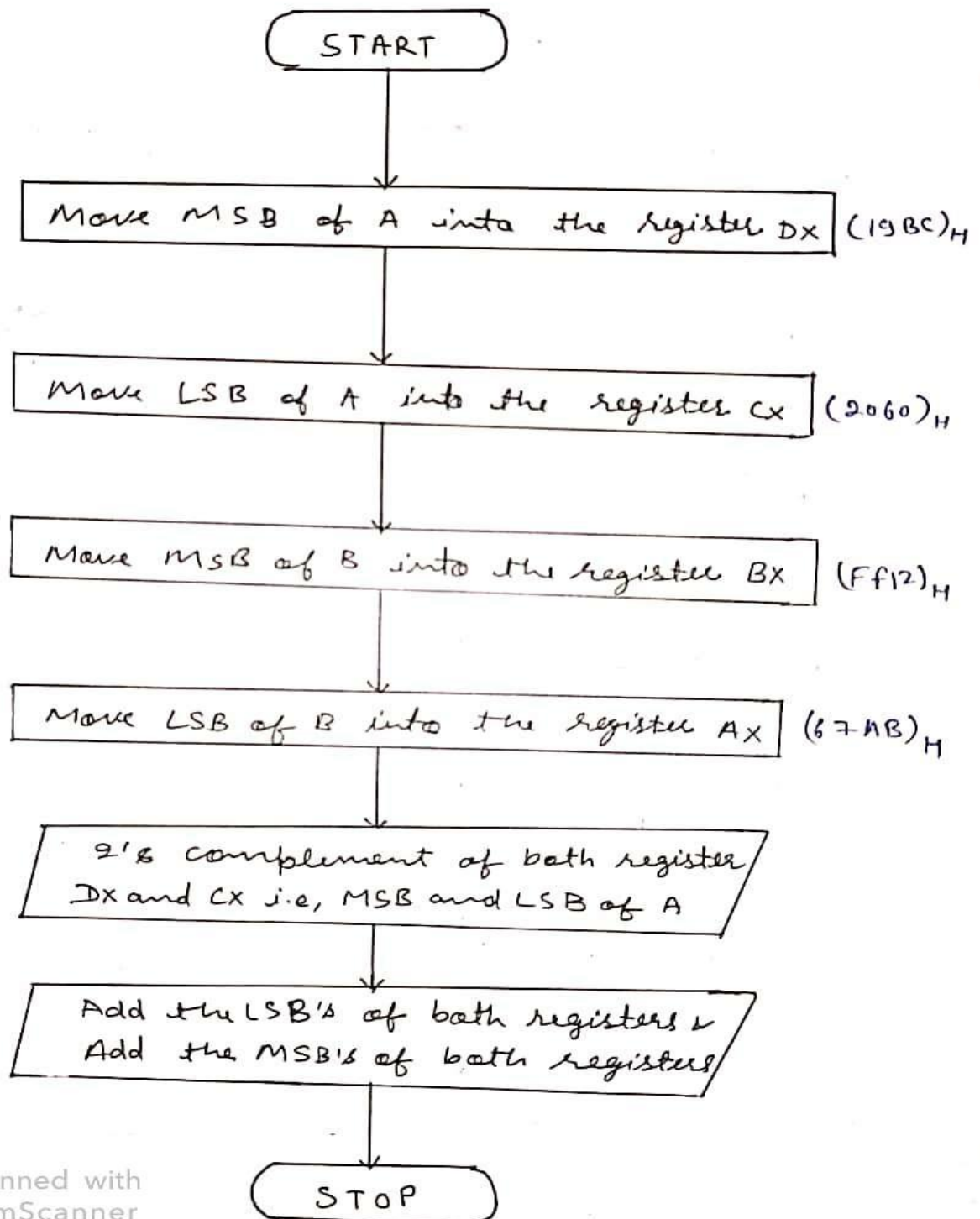
2) If $B = \text{FF1267AB}$, Find $B - A$ using 2's complement.

Aim:

Calculate Subtraction ($B - A$) where $A = (\text{19BC2060})_H$ and $B = (\text{FF1267AB})_H$.

Handwritten Flow chart:

2.



Handwritten Program:

Q.

Subtraction ($B-A$) using 2's complement
where

$A = (19BC2060)H$

$B = (FF1267AB)H$

CODE :-

```
org 100h
; for A
MOV DX, 19BCH
MOV CX, 2060H
; for B
MOV BX, 0FF12H
MOV AX, 67ABH
; Negation of A
NEG CX
NEG DX
; Adding both A and B
ADD AX, CX
ADD BX, DX
```



Scanned with
CamScanner

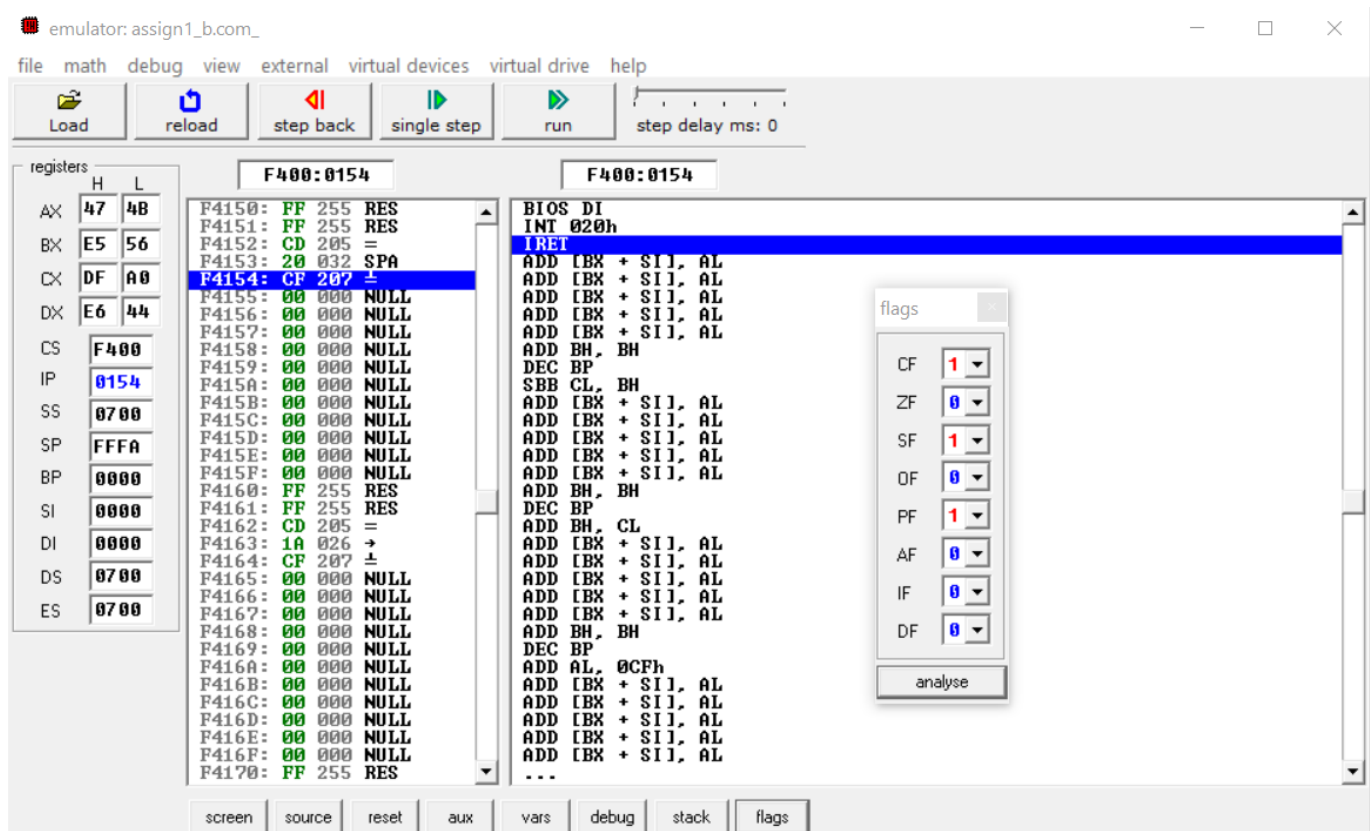
Snapshots of typed program and Output:

edit: C:\Users\Punit Middha\Desktop\Micro\assign1_b.asm

file edit bookmarks assembler emulator math ascii codes help

| | | | | | | | | | |
|-----|------|----------|------|---------|---------|------------|-----------|---------|------|
| new | open | examples | save | compile | emulate | calculator | convertor | options | help |
|-----|------|----------|------|---------|---------|------------|-----------|---------|------|

```
01
02 ; You may customize this and other start-up templates;
03 ; The location of this template is c:\emu8086\inc\0_com_template.txt
04
05 ; Name: PUNIT MIDDHA
06 ; RagNo: 19BCE2060
07
08 org 100h
09
10 ; For A
11 MOV DX, 19BCH
12 MOV CX, 2060H
13
14 ; For B
15 MOV BX, 0FF12H
16 MOV AX, 67ABH
17
18 ; Negation of A
19 NEG CX
20 NEG DX
21
22 ; Adding both A and B
23 ADD AX, CX
24 ADD BX, DX
25
26 ret
```



Inference:

- As the bits are 32 bits long. Each number have to be stored in 2 registers. So, 4 registers have to be used. The result is stored in BX and AX. MSB in BX and LSB bit in AX.
- Hence, B-A = **E556 474B**

➤ Task – 2 (Assessment – 1)

Write the ALP for the following arithmetic operations:

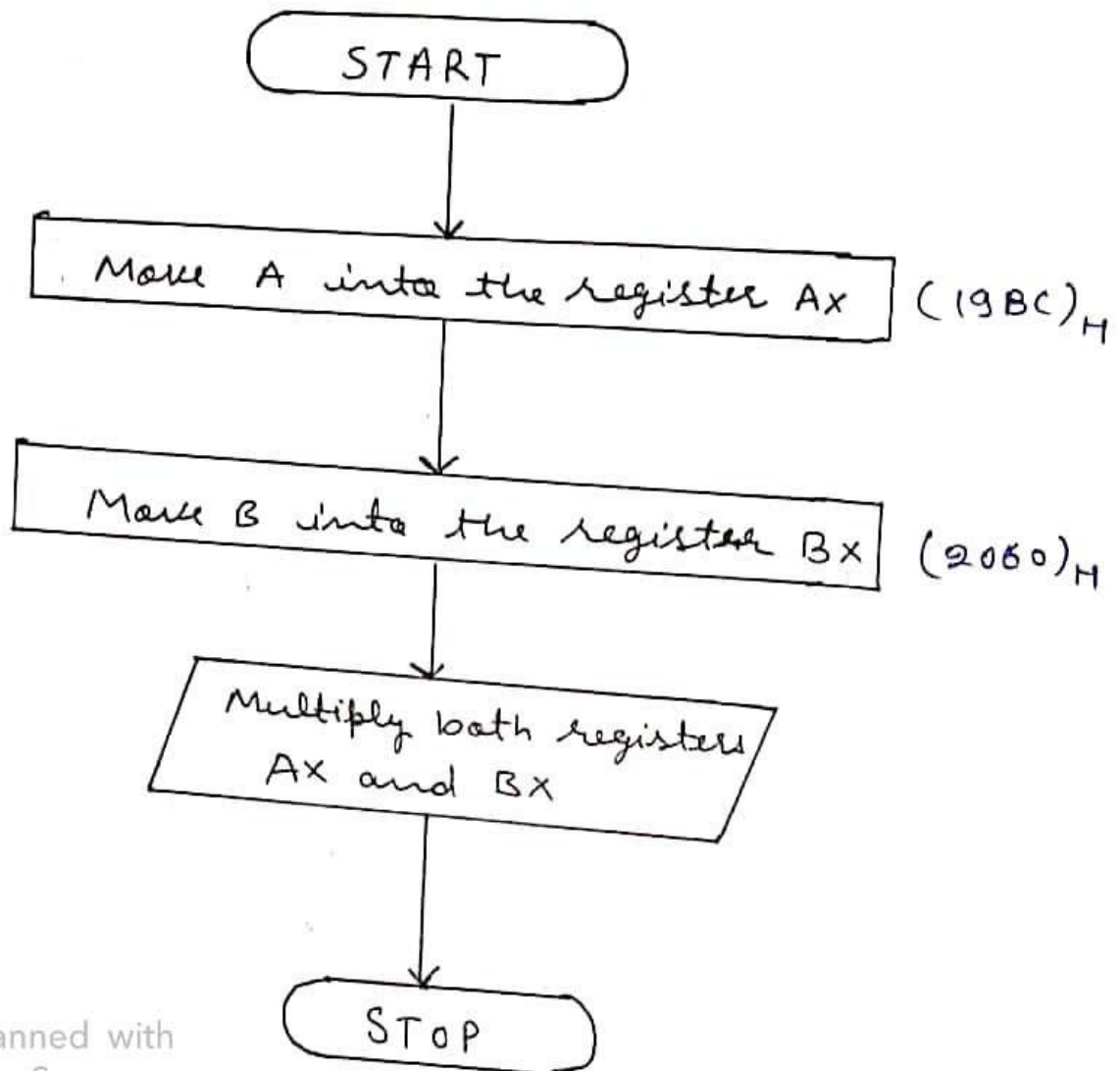
- 1) If $AX = (\text{first 4 digits of your Roll no})_H$ and $BX = (\text{last 4 digits of your Roll no})_H$, find multiplication of AX and BX .

Aim:

Calculate Multiplication of A and B where $A=(19BC)_H$ and $B=(2060)_H$.

Handwritten Flow chart:

1.



Handwritten Program:

1.

Multiplication ($A * B$)

where

$A = (19BC)H$

$B = (2060)H$

CODE :-

org 100h

; for A

MOV AX, 19BCH

; for B

MOV BX, 2060H

; Multiply both A and B

MUL BX

ret



Snapshots of typed program and Output:

```
edit: C:\Users\Punit Middha\Desktop\Micro\assign1_c.asm
file edit bookmarks assembler emulator math ascii codes help

new open examples save compile emulate calculator convertor options help

01 |
02 ; You may customize this and other start-up templates;
03 ; The location of this template is c:\emu8086\inc\0_com_template.txt
04
05 ; Name: PUNIT MIDDHA
06 ; RegNo: 19BCE2060
07
08 org 100h
09
10 ; For A
11 MOV AX, 19BCH
12
13 ; For B
14 MOV BX, 2060H
15
16 ; Multiply both A and B
17 MUL BX
18
19 ret
```

emulator: assign1_com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

| registers | | F400:0154 | | F400:0154 | |
|-----------|------|-----------|--------------------|-------------------|--|
| | H | L | | | |
| AX | 26 | 80 | F4150: FF 255 RES | BIOS DI | |
| BX | 20 | 60 | F4151: FF 255 RES | INT 020h | |
| CX | 00 | 09 | F4152: CD 205 = | IRET | |
| DX | 03 | 41 | F4153: 20 032 SPA | ADD [BX + SI], AL | |
| CS | F400 | | F4154: CF 207 ± | ADD [BX + SI], AL | |
| IP | 0154 | | F4155: 00 000 NULL | ADD [BX + SI], AL | |
| SS | 0700 | | F4156: 00 000 NULL | ADD [BX + SI], AL | |
| SP | FFFA | | F4157: 00 000 NULL | ADD [BX + SI], AL | |
| BP | 0000 | | F4158: 00 000 NULL | ADD [BX + SI], AL | |
| SI | 0000 | | F4159: 00 000 NULL | ADD [BX + SI], AL | |
| DI | 0000 | | F415A: 00 000 NULL | ADD [BX + SI], AL | |
| DS | 0700 | | F415B: 00 000 NULL | ADD [BX + SI], AL | |
| ES | 0700 | | F415C: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F415D: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F415E: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F415F: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F4160: FF 255 RES | ADD [BX + SI], AL | |
| | | | F4161: FF 255 RES | ADD [BX + SI], AL | |
| | | | F4162: CD 205 = | ADD [BX + SI], AL | |
| | | | F4163: 1A 026 → | ADD [BX + SI], AL | |
| | | | F4164: CF 207 ± | ADD [BX + SI], AL | |
| | | | F4165: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F4166: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F4167: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F4168: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F4169: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F416A: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F416B: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F416C: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F416D: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F416E: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F416F: 00 000 NULL | ADD [BX + SI], AL | |
| | | | F4170: FF 255 RES | ADD [BX + SI], AL | |

flags

CF 1 ZF 0 SF 0 OF 1 PF 0 AF 0 IF 0 DF 0

analyse

screen source reset aux vars debug stack flags

Inference:

- The result of multiplication of A and B will be stored in 2 registers as it would be 32 bits long. The msb bits in DX and lsb in AX.
- Hence, A x B = **0341 2680**

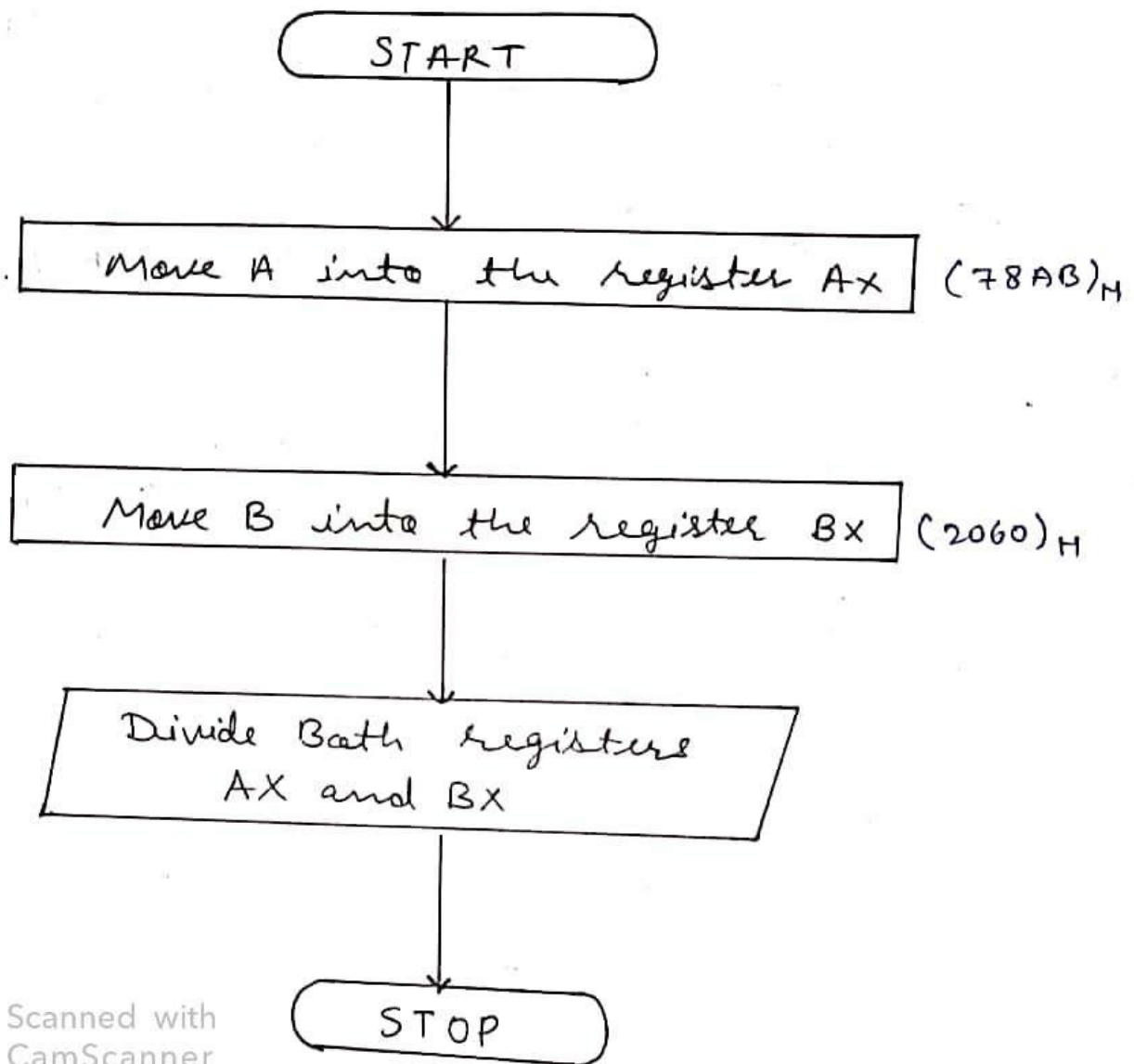
2) Find $(78AB)H \div (\text{last 4 digits of your Roll no})H$

Aim:

Calculate Division($A \div B$) where $A=(78AB)H$ and $B=(2060)H$.

Handwritten Flow chart:

2.



Handwritten Program:

2.

Division (A/B)

where

A = (78AB)H

B = (2060)H

CODE:-

org 100h

;A

MOV AX, 78ABH

;B

MOV BX, 2060H

; A/B

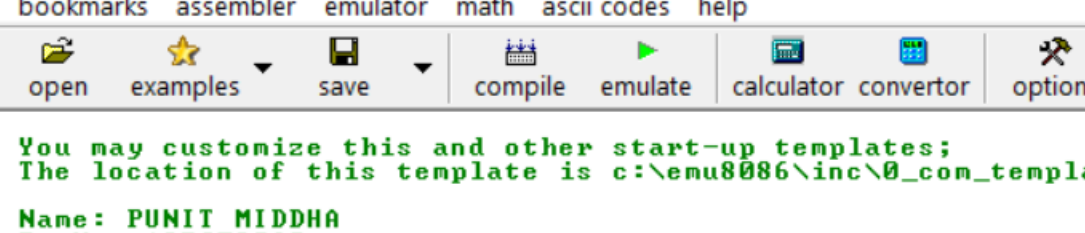
DIV BX



Scanned with
CamScanner

ret

Snapshots of typed program and Output:

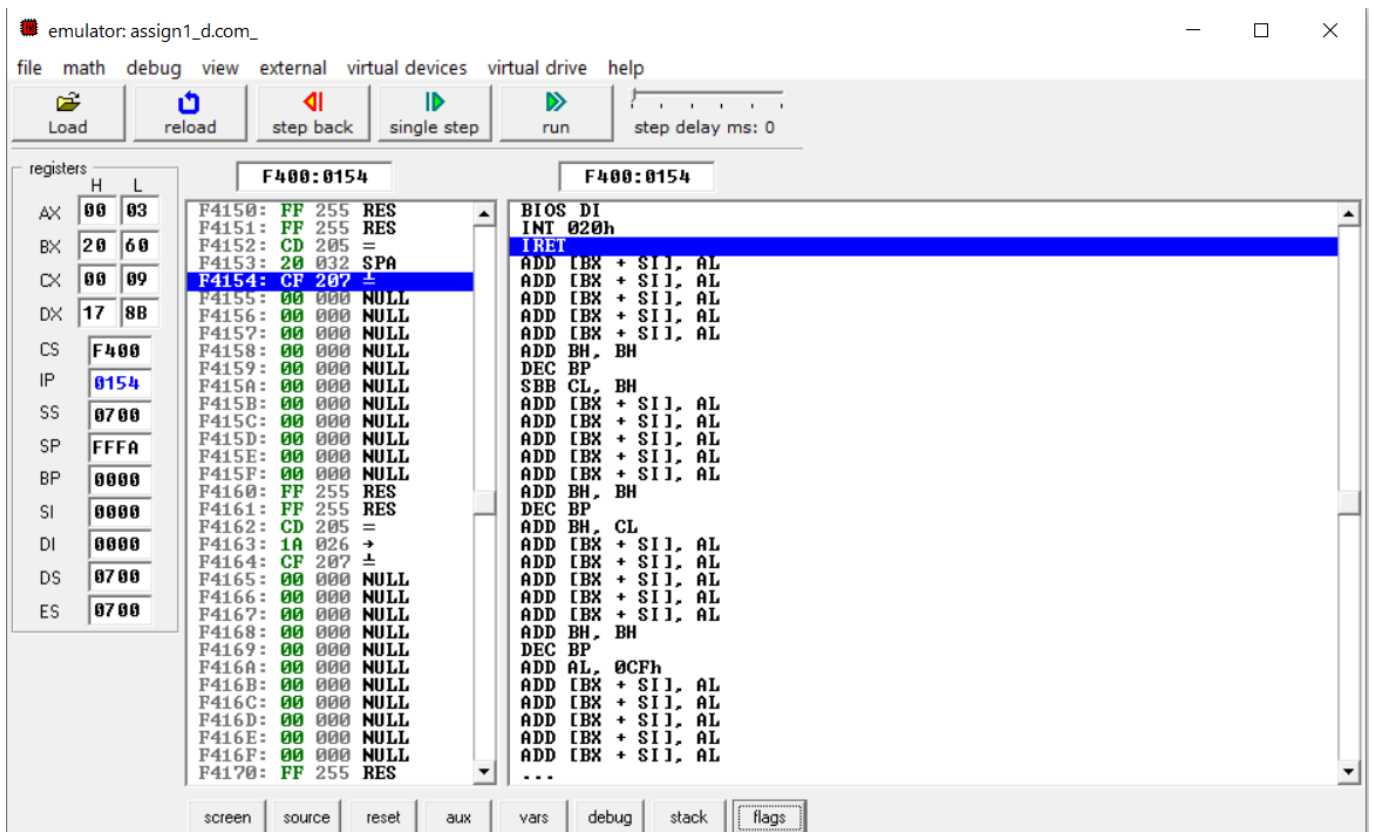


The screenshot shows the Micro Assembler v2.0.10.0 application window. The title bar indicates the file being edited is 'C:\Users\Punit Middha\Desktop\Micro\assign1_d.asm'. The menu bar includes 'file', 'edit', 'bookmarks', 'assembler', 'emulator', 'math', 'ascii codes', and 'help'. The toolbar contains icons for 'new', 'open', 'examples', 'save', 'compile', 'emulate', 'calculator', 'converter', 'options', and 'help'. The main text area displays the following assembly code:

```

01 |
02 ; You may customize this and other start-up templates;
03 ; The location of this template is c:\emu8086\inc\0_com_template.txt
04
05 ; Name: PUNIT MIDDHA
06 ; RegNo: 19BCE2060
07
08 org 100h
09
10 ;A
11 MOV AX, 78ABH
12
13 ;B
14 MOV BX, 2060H
15
16 ;A/B
17 DIV BX
18
19 ret

```



Inference:

- The result of division of A and B is stored in 2 registers AX and DX. The quotient in AX and the remainder in DX.
- Hence, $A/B = 3$ with remainder = **178B**