

| | | | |
|---------------------------------|--|----------------------|-----------|
| Course Name: | Microprocessors and Peripherals (2UXC404) | Semester: | IV |
| Date of Performance: | 17-02-2021 | Batch No: | B2 |
| Faculty Name: | KCS | Roll No: | 1912052 |
| Faculty Sign & Date: | | Grade/Marks : | ___/25 |

Experiment No: 3

Title: Multiplication of 32 bit numbers

Aim and Objective of the Experiment:

Aim: Write an 8086 based ALP to

1. Multiply two 32 bit numbers stored in the data segment and store the result back in the data segment.

Objectives:

To study basic instructions and addressing modes of 8086. Understand assembler directives and concept of data and code segment

This experiment covers following instructions groups.

- a) Data transfer
- b) Arithmetic (Multiply instructions)

COs to be achieved:

CO 2. Develop 8086 based assembly language programs for various applications.

Useful links

NASM Assembler

https://www.tutorialspoint.com/compile_assembly_online.php

MASM/TASM Assembler

Work to be done

1. Upload image of handwritten algorithm/flowchart and 1st file of the program and output screenshots . Also upload results for post lab questions.

data segment

n1h dw 1234h

n1l dw 5678h

n2h dw 1234h

n2l dw 5678h

prod dw 4 dup(0)

data ends

```
code segment
    assume cs: code,ds:data
start: mov ax,data
       mov ds,ax

       mov cx,0
       mov ax,n1l
       mul n2l
       mov prod,ax
       mov prod+2,dx

       mov ax,n1h
       mul n2l
       add prod+2,ax
       adc prod+4,dx

       mov ax,n2h
       mul n1l
       add prod+2,ax
       adc prod+4,dx

       jnc it4
       inc cx
it4:  mov ax,n1h
       mul n2h
       add prod+4,ax
       adc dx,cx
       add prod+6,dx

       mov ah,4ch
       int 21h
       code ends
end start
```



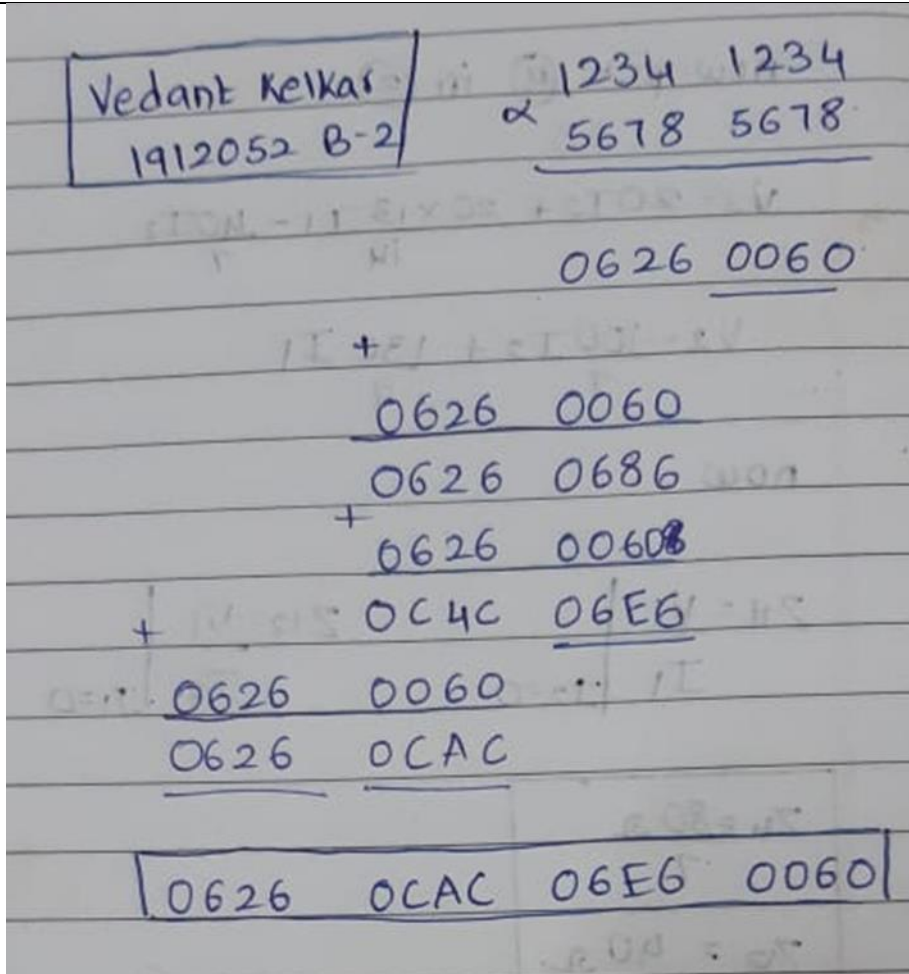
The screenshot displays the emu8086 interface with the following components:

- Assembly Code (Left Panel):**

```
01 data segment
02     n1h dw 1234h
03     n1l dw 5678h
04     n2h dw 1234h
05     n2l dw 5678h
06     prod dw 4 dup(0)
07 data ends
08
09 code segment
10     assume cs:code,ds:data
11     start: mov ax,data
12            mov ds,ax
13
14     mov cx,0
15     mov ax,n1l
16     mul n2l
17     mov prod,ax
18     mov prod+2,dx
19
20     mov ax,n1h
21     mul n2h
22     add prod+2,ax
23     adc prod+4,dx
24
25     mov ax,n2h
26     mul n1l
27     add prod+2,ax
28     adc prod+4,dx
29
30     jnc it4
31     inc cx
32     it4: mov ax,n1h
33          mul n2h
34          add prod+2,ax
35          adc prod+4,dx
36          dx,cx
37          add prod+6,dx
38
39     mov ah,4ch
40     int 21h
41     add esp,4
```
- Registers (Middle Panel):**

| Register | H | L |
|----------|------|------|
| AX | 4C | 90 |
| BX | 00 | 00 |
| CX | 00 | 00 |
| DX | 01 | 4B |
| SI | 0000 | 0000 |
| DI | 0000 | 0000 |
| ES | 0710 | 0700 |
- Memory (Bottom Panel):**

| Address | Value |
|-----------|--|
| 0710:0000 | 34 12 78 56 34 12 78 56-40 D8 F4 1D DC 66 4B 01 |
| 0710:0010 | D8 10 07 8E D8 B9 00 00-A1 02 00 F7 26 06 00 A3 |
| 0710:0020 | 08 00 09 16 00 00 A1 00-00 F7 26 06 00 01 06 00 |
| 0710:0030 | 00 11 16 0C 00 00 A1 04-00-F7 26 02 00 01 06 00 00 |
| 0710:0040 | 11 16 0C 00 73 01 41 A1-00 00 F7 26 04 00 01 06 |
| 0710:0050 | 0C 00 13 D1 61 16 0E 00-04 4C CD 21 90 90 90 90 |
| 0710:0060 | 90 90 90 90 90 90 90 90-90 90 90 90 90 90 |
| 0710:0070 | 0A 0A 0A 0A 0A 0A 0A 0A-0A 0A 0A 0A 0A 0A |



data segment
 n1h dw 1234h
 n1l dw 1234h
 n2h dw 5678h
 n2l dw 5678h
 prod dw 4 dup(0)
 data ends

code segment
 assume cs: code, ds: data
 start: mov ax, data
 mov ds, ax

 mov cx, 0
 mov ax, n1l
 mul n2l
 mov prod, ax
 mov prod+2, dx

 mov ax, n1h

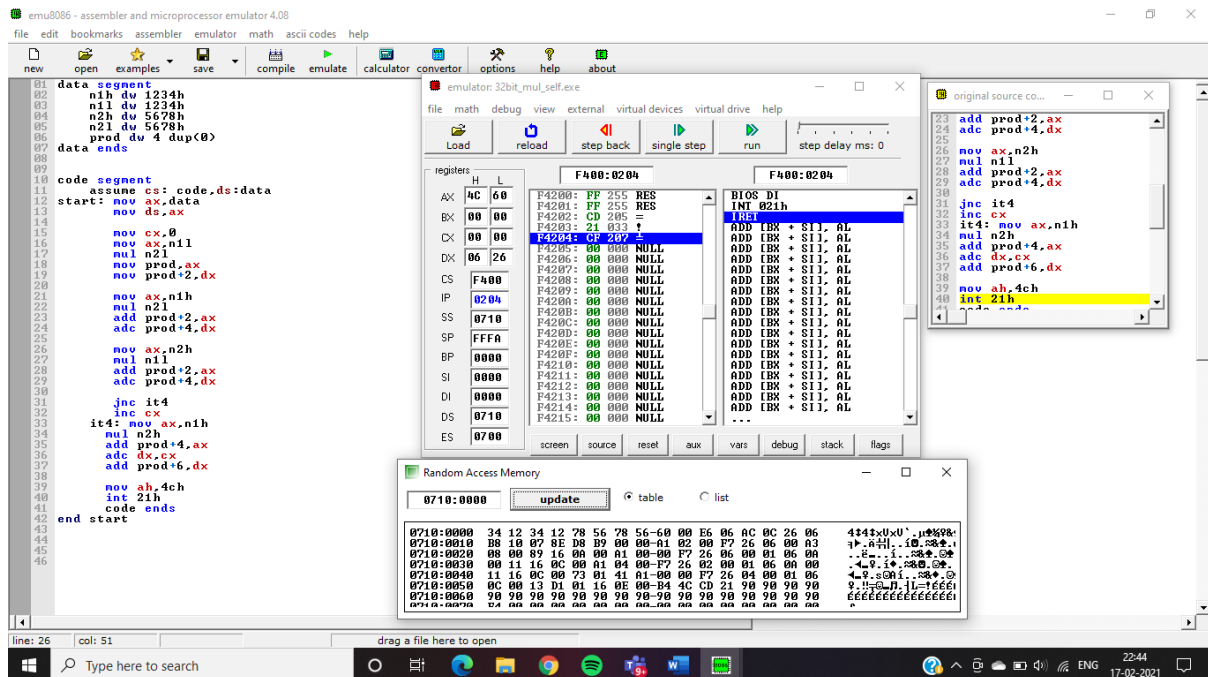
```

mul n2l
add prod+2,ax
adc prod+4,dx

mov ax,n2h
mul n1l
add prod+2,ax
adc prod+4,dx

jnc it4
inc cx
it4: mov ax,n1h
mul n2h
add prod+4,ax
adc dx,cx
add prod+6,dx

mov ah,4ch
int 21h
code ends
end start
  
```



Post Lab Subjective/Objective type Questions:

Q.1 Write an 8086 based ALP to find the factorial of a number in data segment and store the result back in data segment

data segment

A db 05h

fact dw 4 dup(0)

data ends

code segment

assume cs: code,ds:data

start: mov ax,data

mov ds,ax

mov ah,00

mov al,A

X:dec A

mov cl,A

cmp cl,01

jz stop

mul A

jmp X

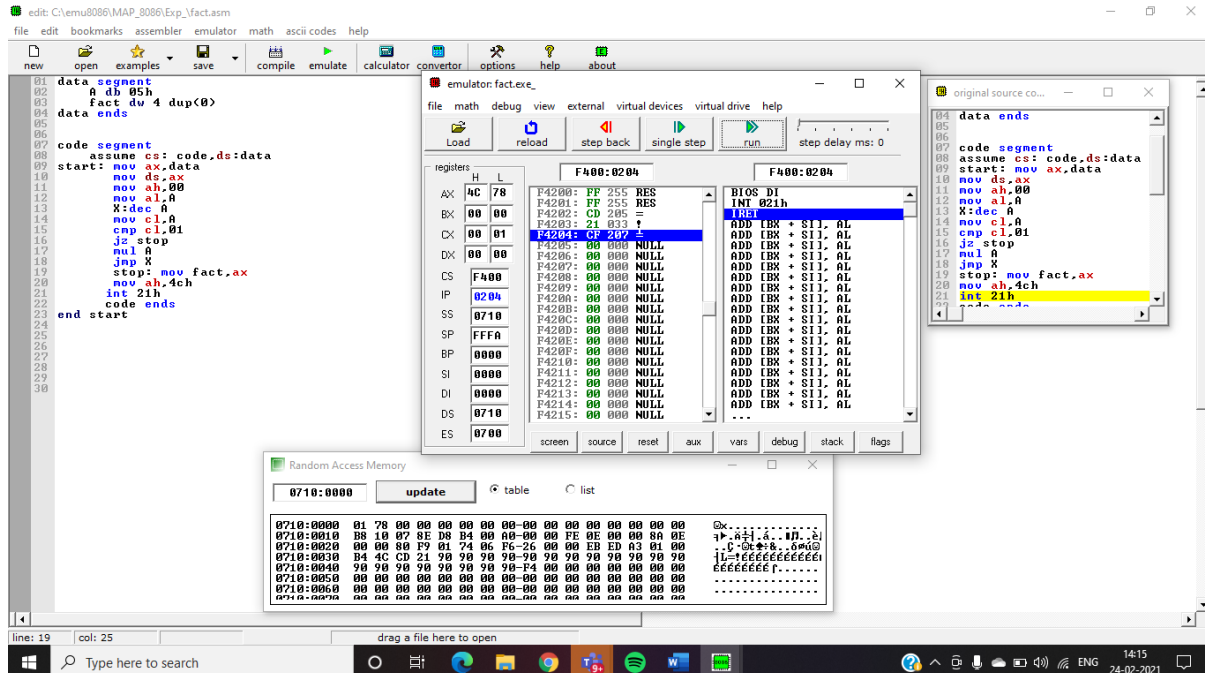
stop: mov fact,ax

mov ah,4ch

int 21h

code ends

end start



Q.2. What is the output of the following instruction?



AX = 37D7H, BH = 151 decimal
DIV BH

Remainder will be stored in AH
Quotient in AL
AH=65H=10 decimal
AL=5EH=94 decimal

Q.3 What is the difference between MUL and IMUL? Explain with example

MUL Multipl byte or word (unsigned)
MUL,Integer multiply byte or word (signed)

Conclusion:

Wrote 8086 program to multiply two 32 bit numbers stored in the data segment and store the result back in the data segment.

**Signature of faculty in-charge with
Date:**