

MICROPROCESSOR – EXPT 2

WARREN FERNANDES

8940

CODE:

data segment

num1 dw 4444h

num2 dw 0002h

result dw ?

quo dw ?

rem dw ?

msg db 'Enter option: \$'

msg1 db '1)ADD \$'

msg2 db '2)SUB \$'

msg3 db '3)MUL \$'

msg4 db '4)DIV \$'

code segment

assume ds:data, cs:code

start:

mov ax,data

mov ds,ax

lea dx,msg

mov ah,09h

int 21h

lea dx,msg1

mov ah,09h

int 21h

lea dx,msg2

mov ah,09h

int 21h

```
lea dx,msg3
mov ah,09h
int 21h
lea dx,msg4
mov ah,09h
int 21h
mov ah,08h
int 21h
cmp al,31h
jnz next1
mov ax,num1
add ax,num2
mov result,ax
next1:
cmp al,32h
jnz next2
mov ax,num1
sub ax,num2
mov result,ax
next2:
cmp al,33h
jnz next3
mov ax,num1
mov bx,num2
mul bx
mov result,ax
next3:
cmp al,34h
jnz next4
```

mov ax,0000h

mov dx,0000h

mov ax,num1

div num2

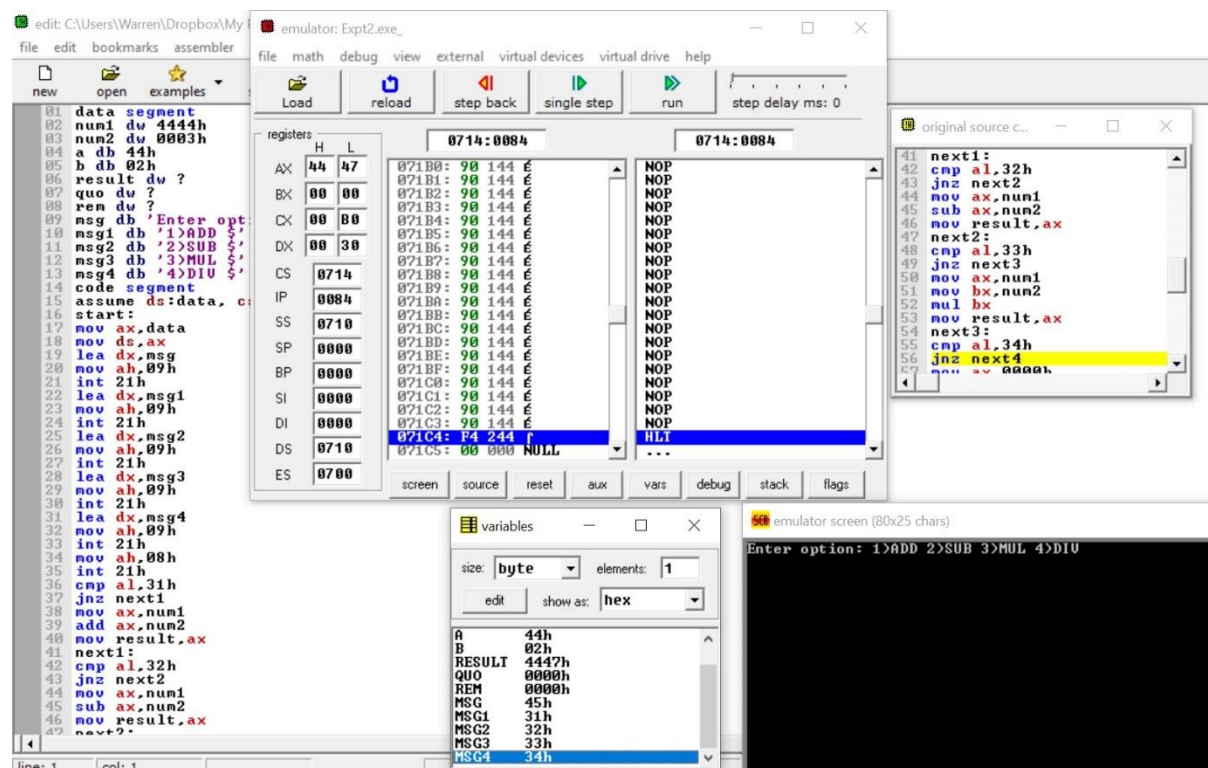
mov quo,ax

mov rem,dx

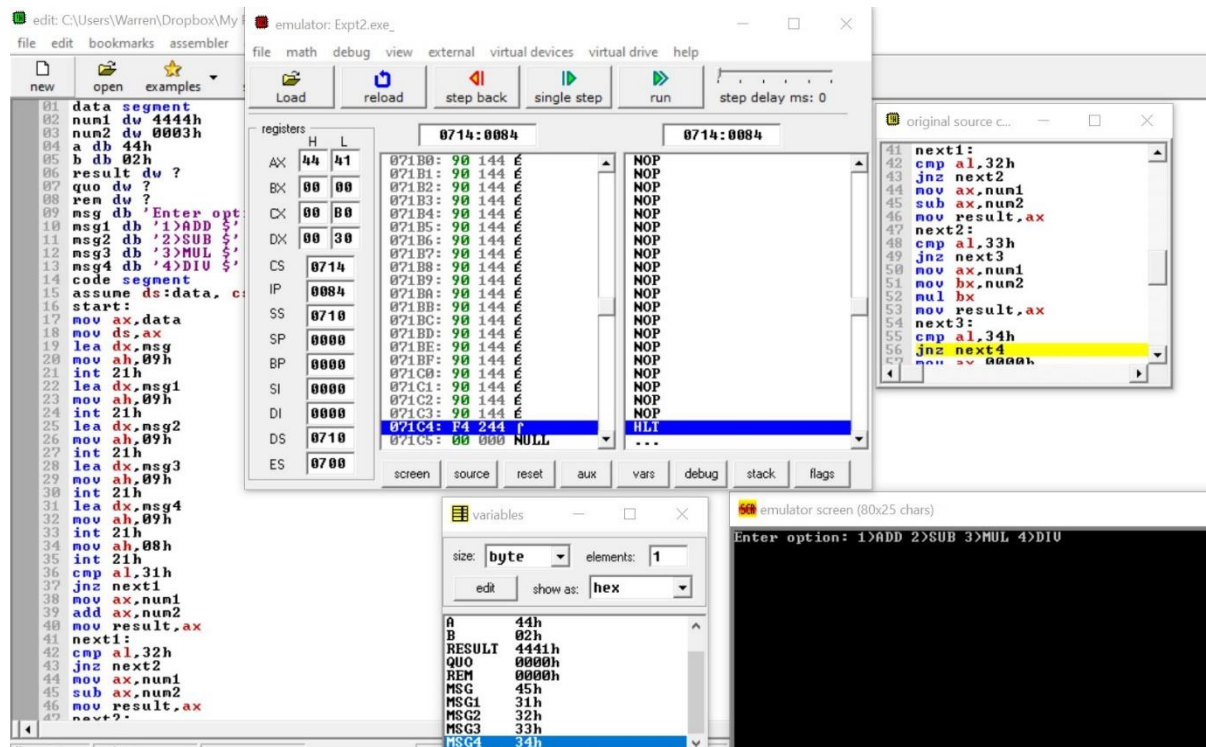
next4:

end start

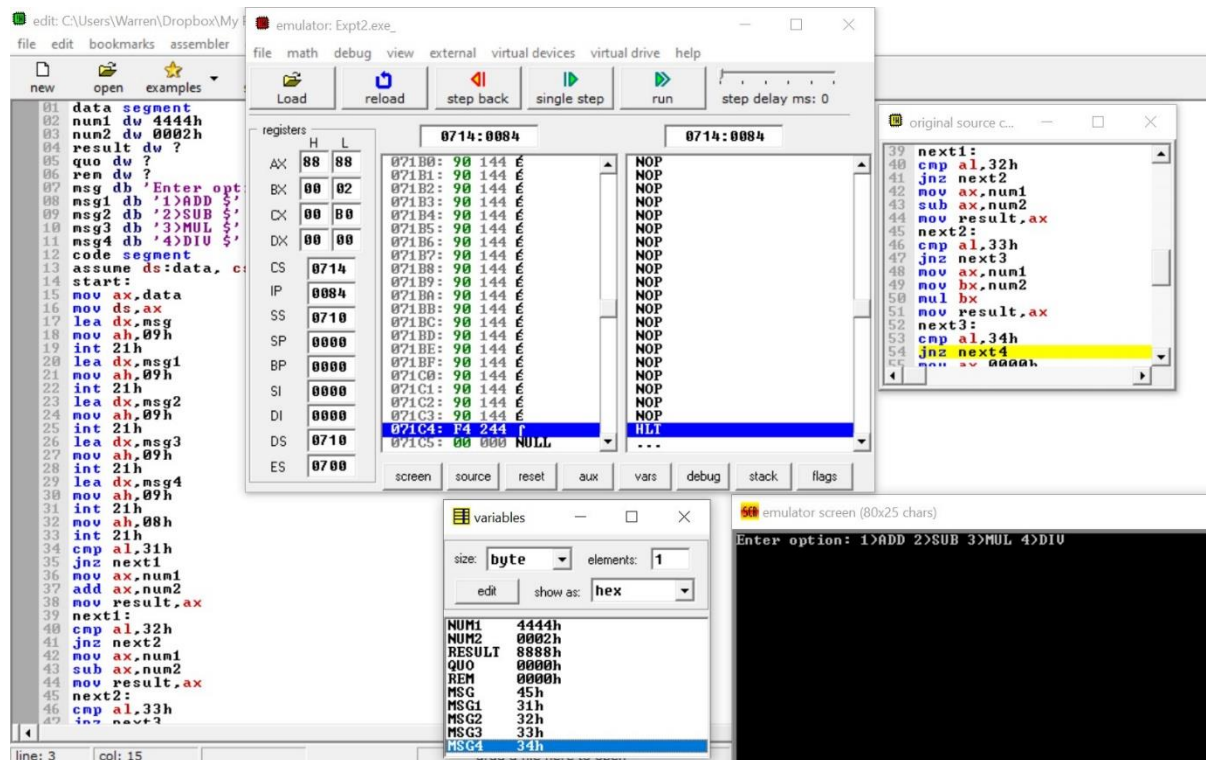
16 Bit Addition



16 Bit Subtraction



16 Bit Multiplication



The image displays the Expt2 emulator interface, which is used for executing and debugging assembly code. The interface is divided into several panels:

- Top Panel:** Contains the file menu, a toolbar with icons for Load, reload, step back, single step, run, and a step delay slider set to 0 ms.
- Left Panel (Registers):** A table showing the current values of CPU registers.

Register	H	L
AX	22	22
BX	00	00
CX	00	00
DX	00	00
CS	0714	
IP	0084	
SS	0710	
SP	0000	
BP	0000	
SI	0000	
DI	0000	
DS	0710	
ES	0700	
- Center Panel (Memory):** Displays memory contents at address 0714:0084. It shows two columns of memory data, with the first column containing hexadecimal values and the second column containing their ASCII representations.

Address	Hex	ASCII
071B5:	90 144	€
071B6:	90 144	€
071B7:	90 144	€
071B8:	90 144	€
071B9:	90 144	€
071BA:	90 144	€
071BB:	90 144	€
071BC:	90 144	€
071BD:	90 144	€
071BE:	90 144	€
071BF:	90 144	€
071C0:	90 144	€
071C1:	90 144	€
071C2:	90 144	€
071C3:	90 144	€
071C4:	90 144	€
071C5:	00 000	NULL
071C6:	00 000	NULL
071C7:	00 000	NULL
071C8:	00 000	NULL
071C9:	00 000	NULL
071CA:	00 000	NULL
- Right Panel (Original Source):** Displays the original assembly source code.


```

45 next2:
46 cmp al,33h
47 jnz next3
48 mov ax,num1
49 mov bx,num2
50 mul bx
51 mov result,ax
52 next3:
53 cmp al,34h
54 jnz next4
55 mov ax,0000h
56 mov dx,0000h
57 mov ax,num1
58 div num2
59 mov quo,ax
60 mov rem,dx
61 next4:

```
- Bottom Panel (Variables):** A table showing the current values of variables.

Variable	Value
NUM1	4444h
NUM2	0002h
RESULT	0000h
QUO	2222h
REM	0000h
MSG	45h
MSG1	31h
MSG2	32h
MSG3	33h
MSG4	34h
- Bottom Right Panel (Emulator Screen):** Displays the output of the program, showing the prompt "Enter option:" followed by the user input "1>ADD 2>SUB 3>MUL 4>DIV".

POSTLAB:

Q1. Registers of 8086.

In 8086 Microprocessor, the registers are categorized into mainly four types.

1) General Purpose Registers.

The use of general purpose registers is to store temporary data. While the instructions are executed in the control unit they may work on some numeric value or some operands.

- 1) AX 2) BX 3) CX 4) DX

2) Segment Registers.

1) Code Segment.

Users cannot modify the content.

2) Data Segment.

User can modify the content.

3) Stack Segment.

SS is used to store the information about the memory segment.

4) Extra Segment.

If there is less space in DS segment then ES is used.

3) Point & Index Registers.

1) Instruction Pointer.

2) Base Pointer.

3) Stack Pointer.

4) Source Index.

5) Destination Index.

4) Flag OR Status Registers.

The flag or status register which contains 9 flags, and the remaining 7 bits are idle in this register. These flags tell about the status of the process after any arithmetic or logical operation.

2. Logical Address

It is generated by CPU while a program is running. The logical address is virtual address as it does not exist physically therefore, it is also known as Virtual Address. This address is used as a reference to access the physical memory location by CPU. The term logical address space is used for the set of all logical addresses. The hardware device, called Memory Management Unit is used for mapping logical address.

Physical Address

It identifies a physical location of required data in a memory. The user never directly deals with the physical address but can access by its corresponding logical address must be mapped by the physical address by MMU before they are used.