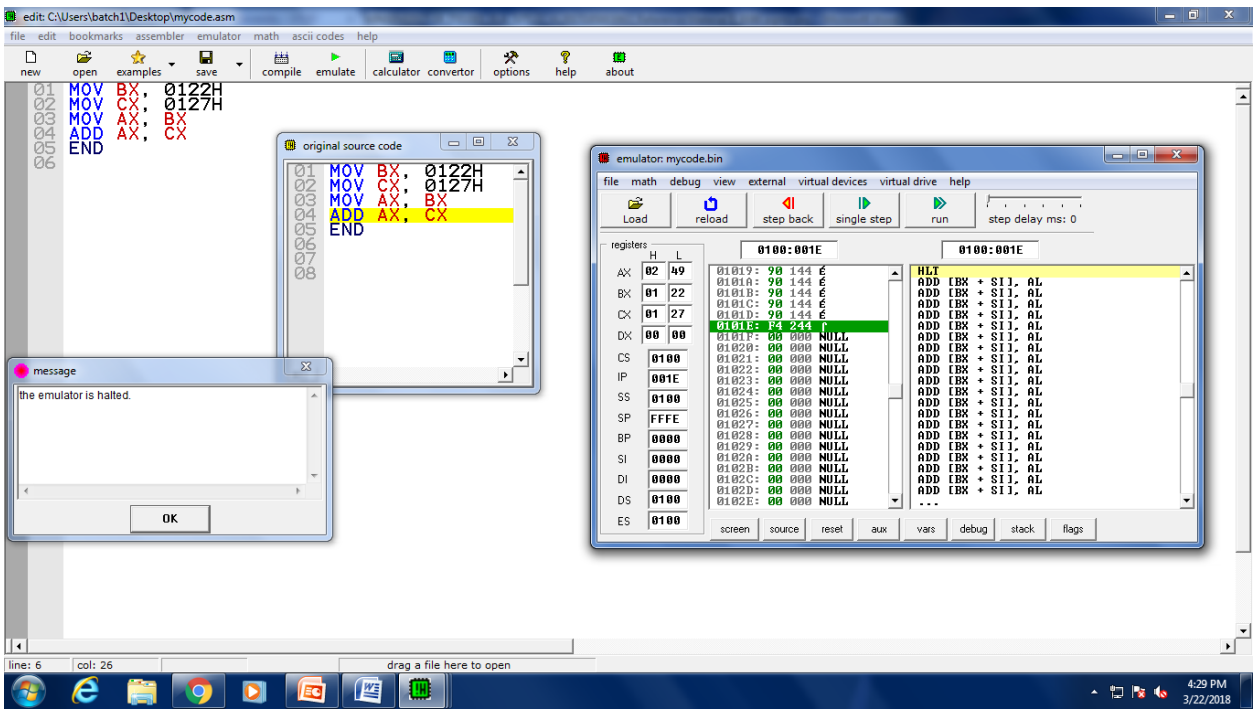
SIDDHI SINGH
17BIT0028

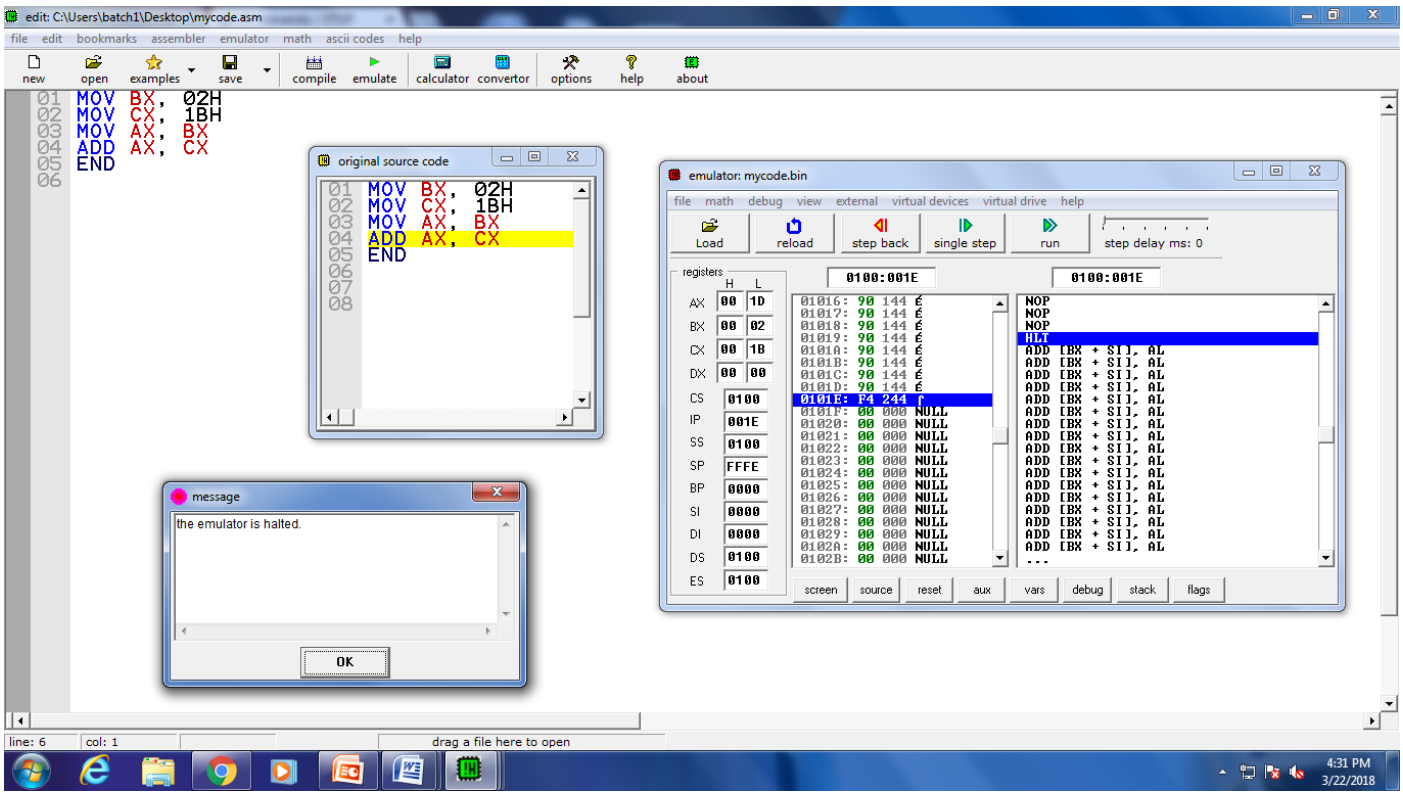
1. Write an ALP for adding two 16 bit numbers 0122H, 0127H stored in registers B and C respectively.

```
MOV BX, 0122H
MOV CX, 0127H
MOV AX, BX
ADD AX, CX
END
```



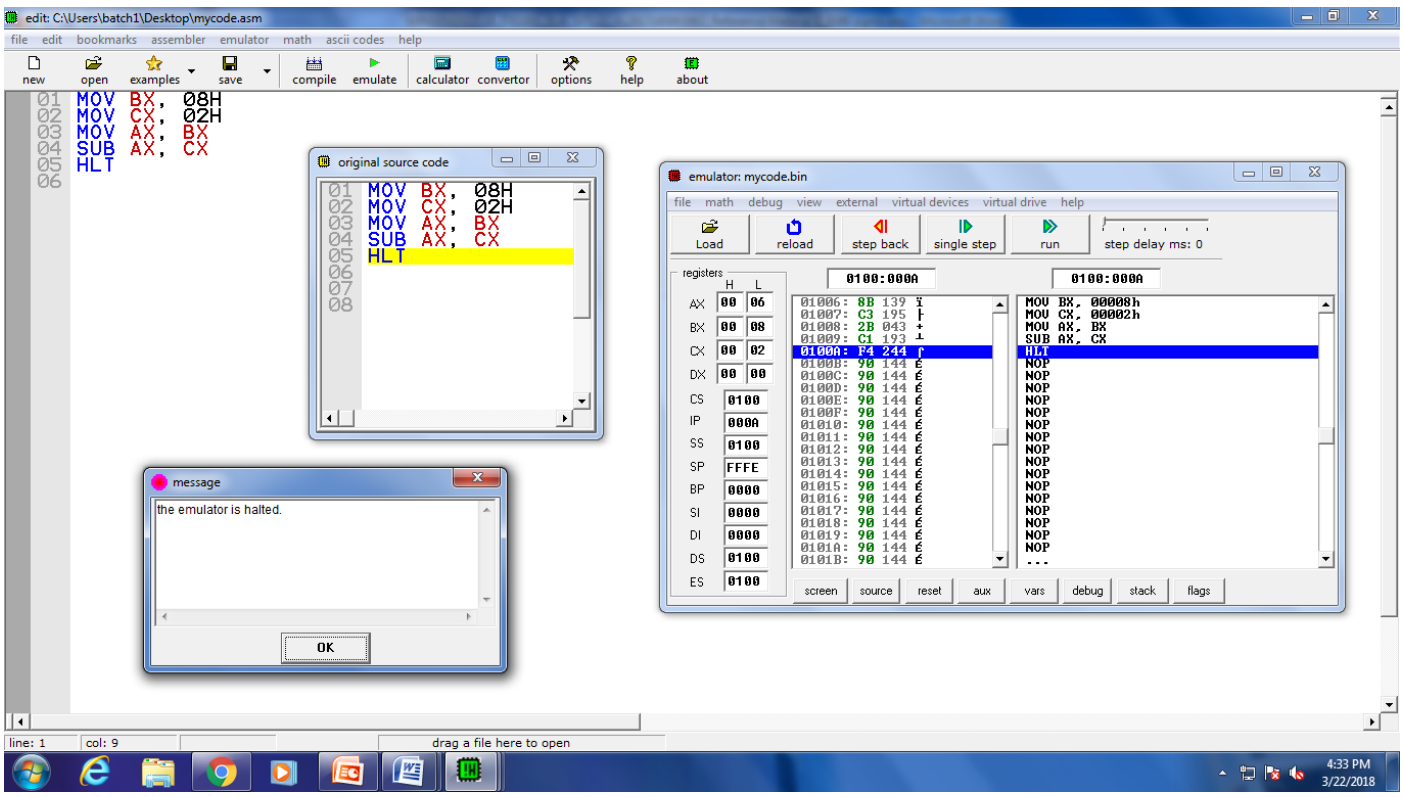
2. Write an ALP for adding two 8 bit numbers 02H, 1BH stored in registers B and C.

```
MOV BX, 02H
MOV CX, 1BH
MOV AX, BX
ADD AX, CX
END
```



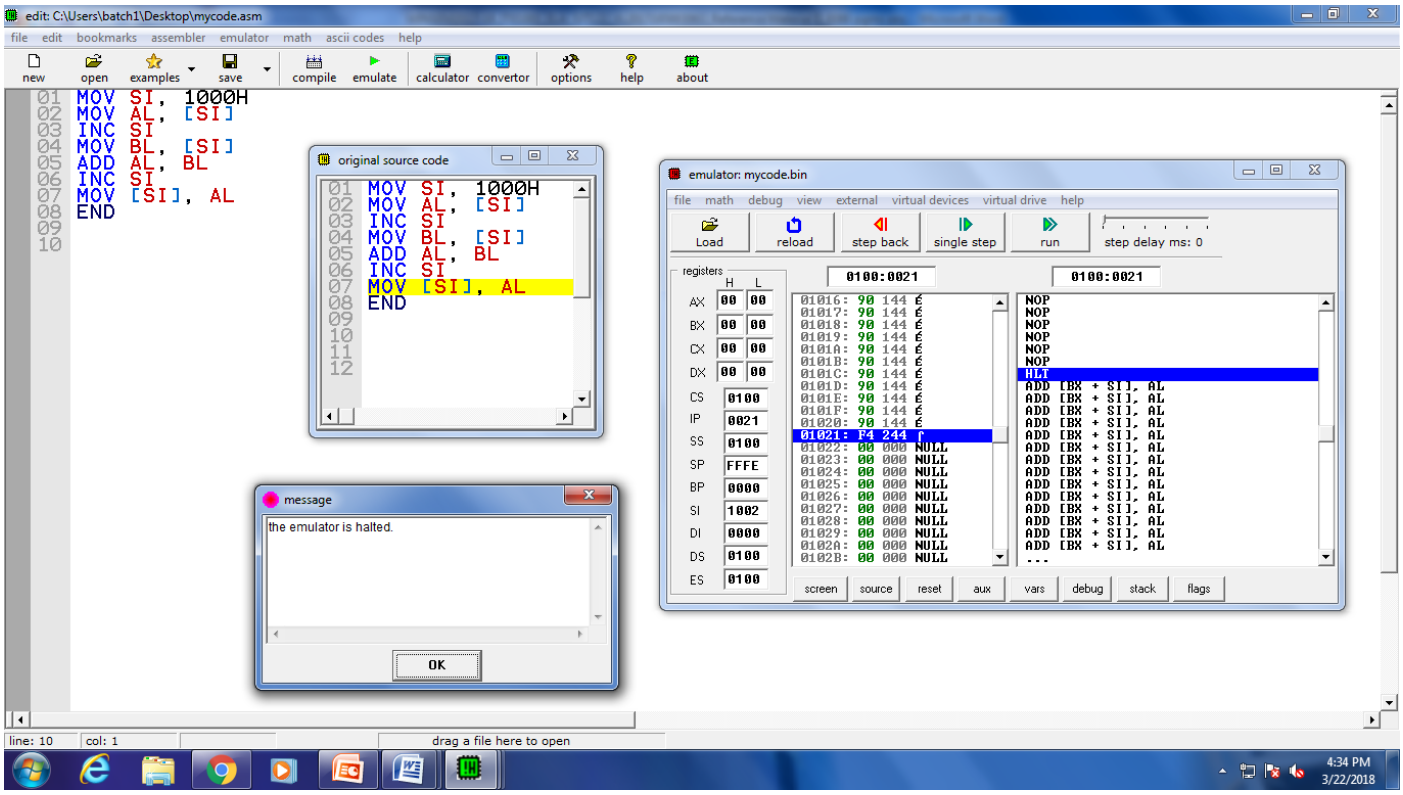
3. Write an ALP for subtracting two 8 bit numbers stored in registers

```
MOV BX.08H
MOV CX, 02H
MOV AX, BX
SUB AX, CX
HLT
```



4. Write an ALP for storing the 16 bit data in memory address 1000 and 1001. Then perform addition of these two numbers and store the result in the memory address 1002.

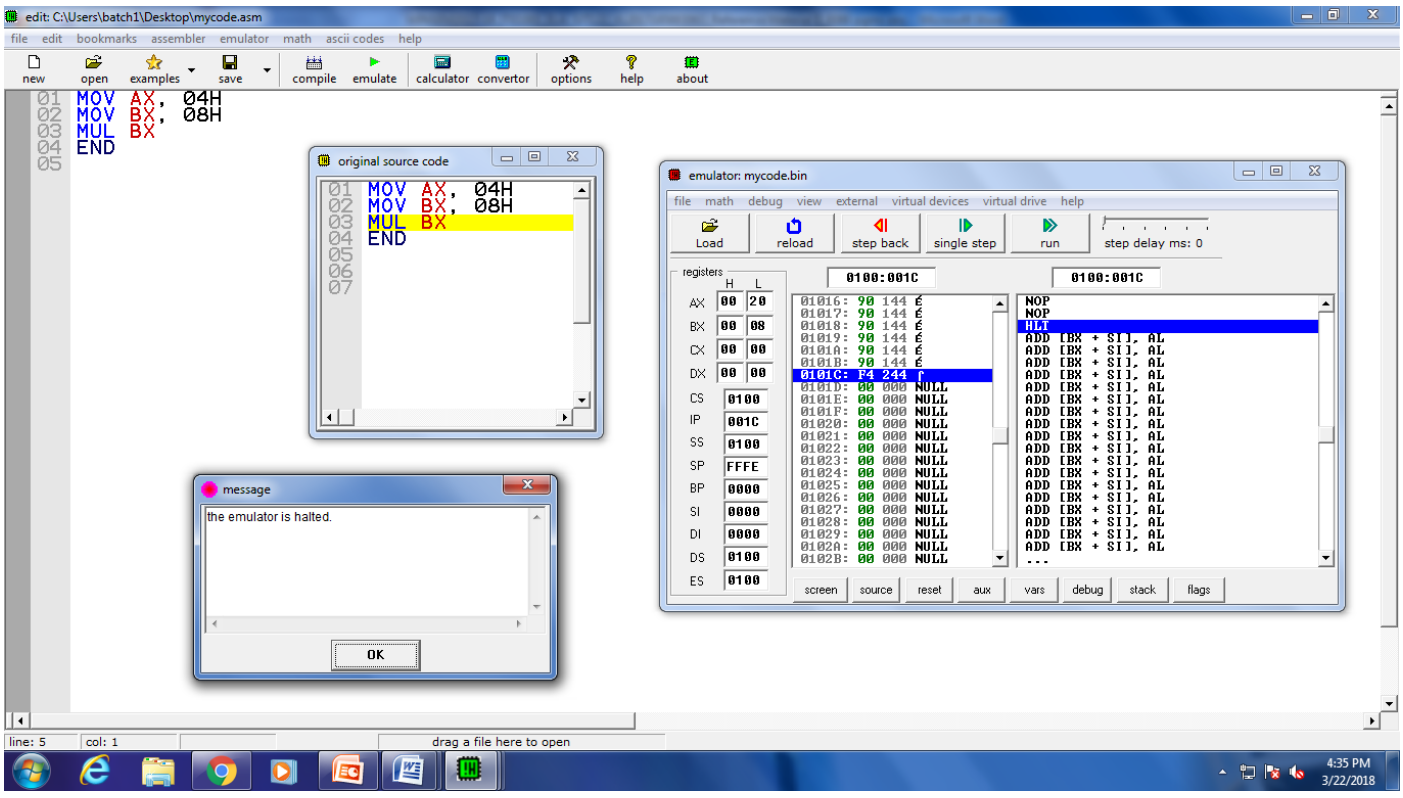
```
MOV SI, 1000H
MOV AL, [SI]
INC SI
MOV BL, [SI]
ADD AL, BL
INC SI
MOV [SI], AL
END
```



5. Write an ALP for performing multiplication and division of 2 16 bit numbers.

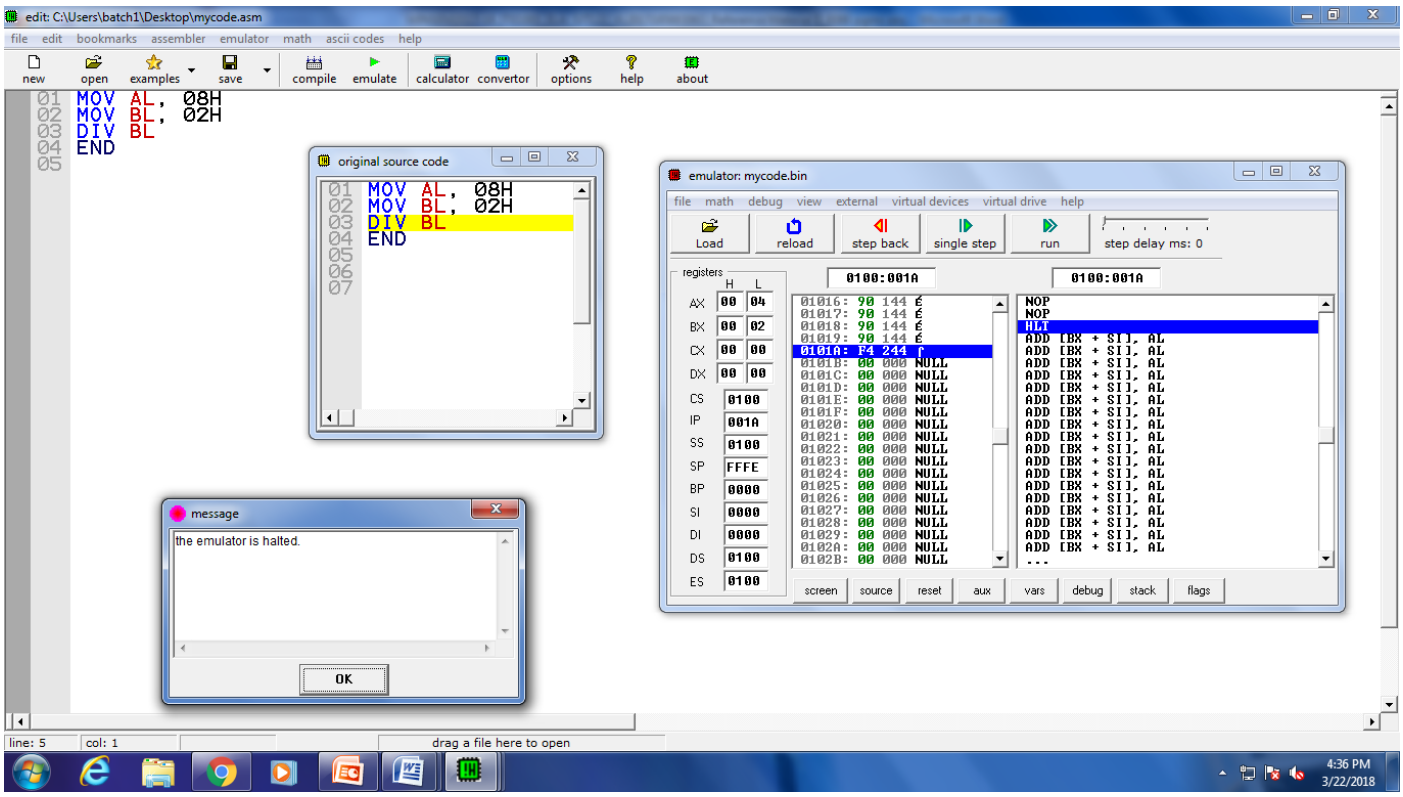
FOR MULTIPICATION

```
MOV AX, 04H
MOV BX, 08H
MUL BX
END
```



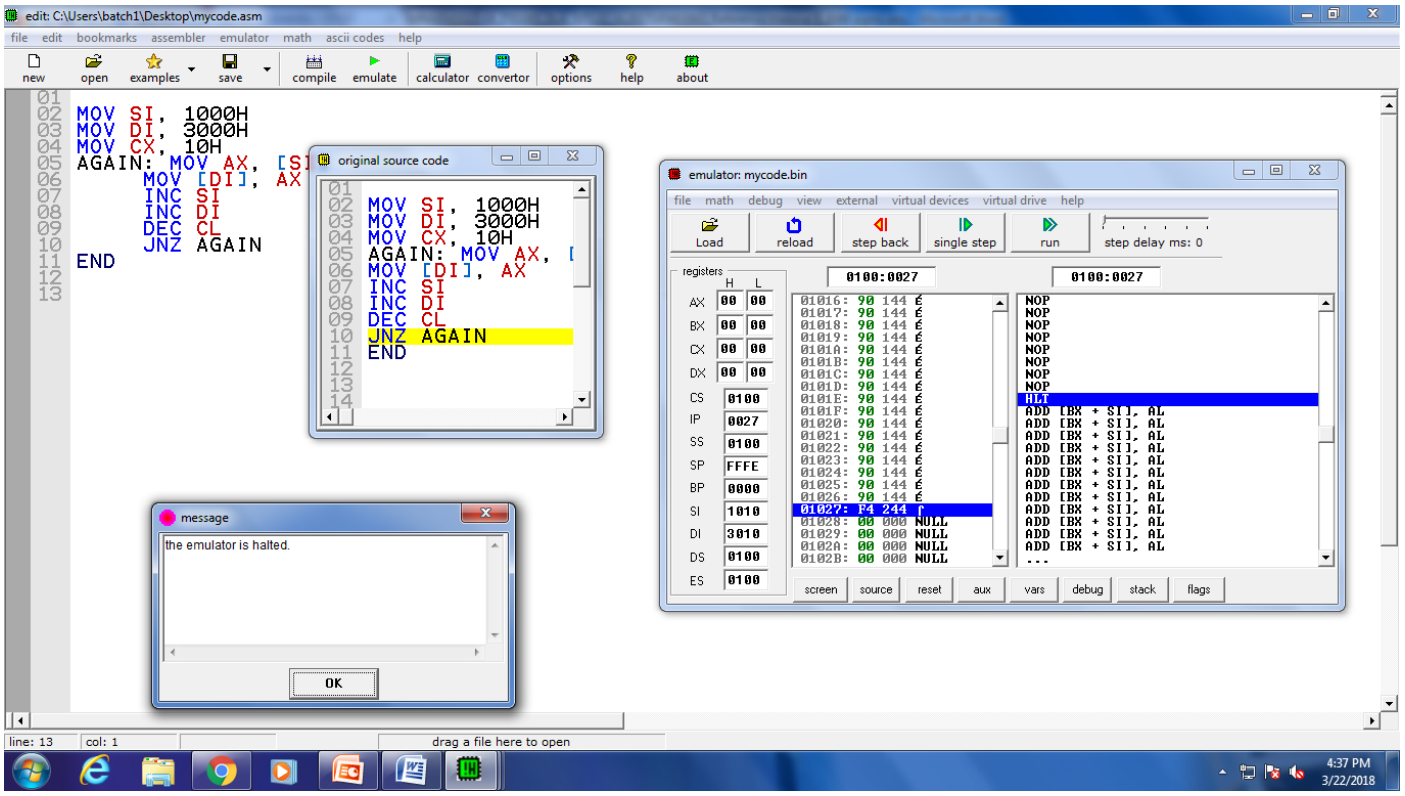
FOR DIVISION

```
MOV AL, 08H
MOV BL, 02H
DIV BL
END
```



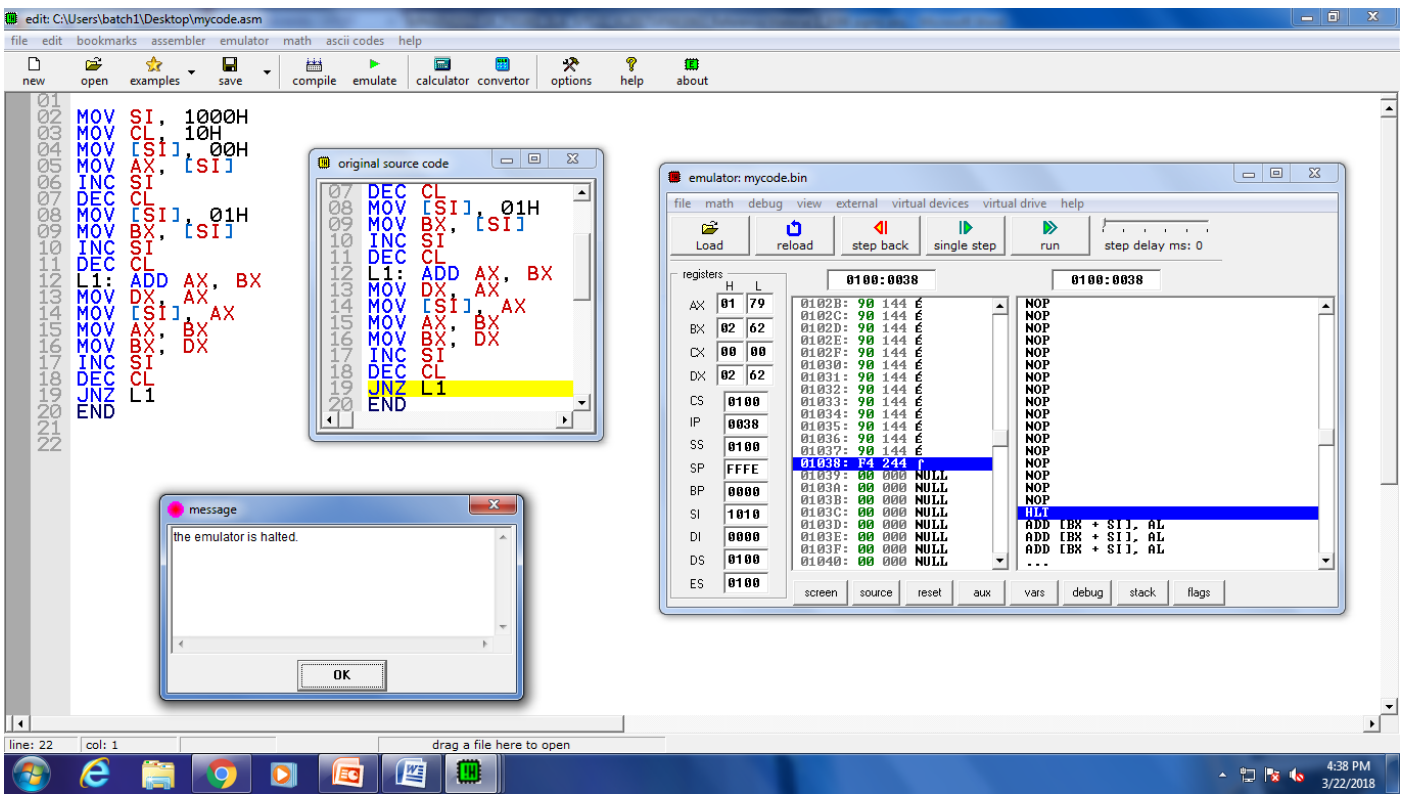
6. Write an ALP for transferring a block of ten 16 bit data from memory address starting at 1000 to a memory location 3000.

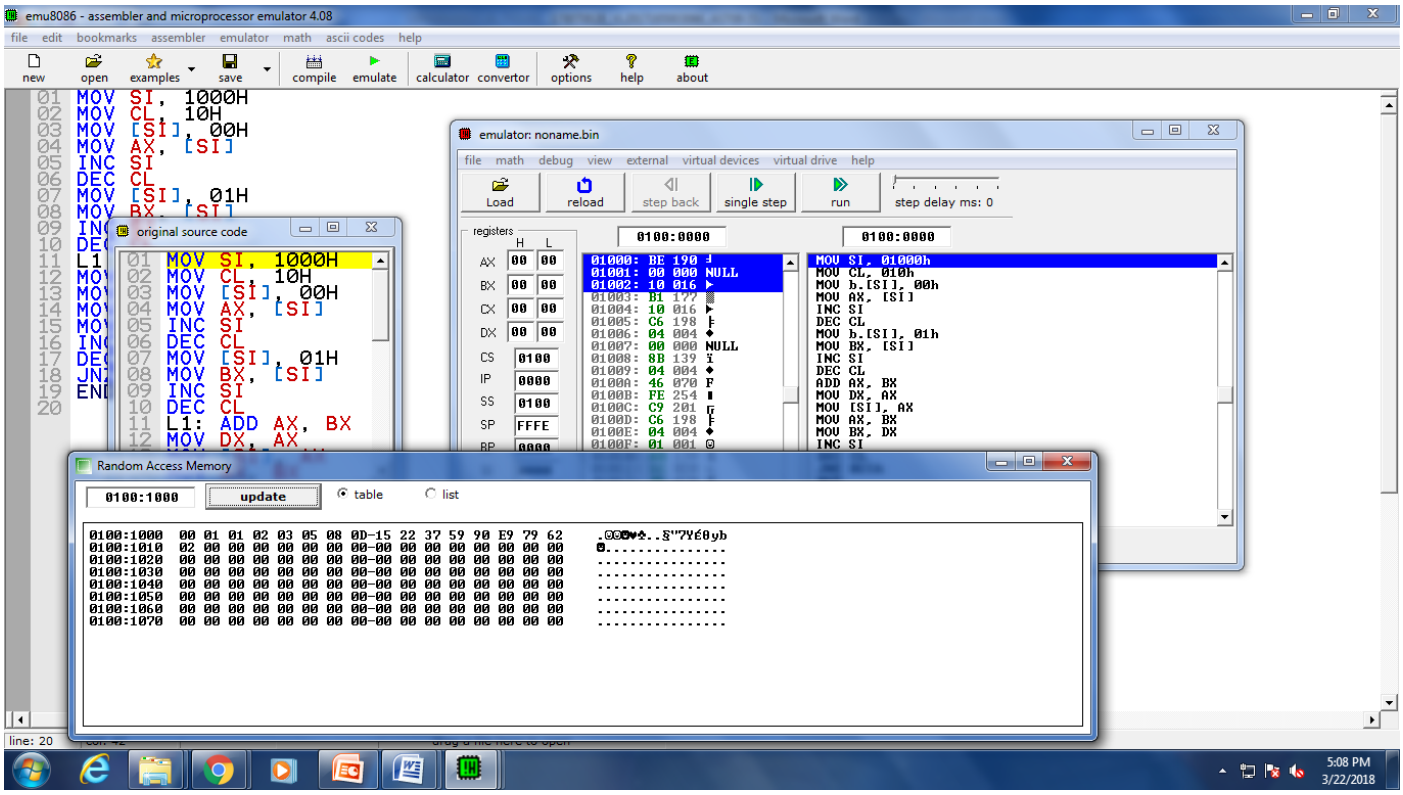
```
MOV SI, 1000H
MOV DI, 3000H
MOV CX, 10H
AGAIN: MOV AX, [SI]
      MOV [DI], AX
      INC SI
      INC DI
      DEC CL
      JNZ AGAIN
END
```



7. Write an ALP to display 13 numbers of the fibonacci series in a memory location

```
MOV SI, 1000H
MOV CL, 10H
MOV [SI], 00H
MOV AX, [SI]
INC SI
DEC CL
MOV [SI], 01H
MOV BX, [SI]
INC SI
DEC CL
L1: ADD AX, BX
MOV DX, AX
MOV [SI], AX
MOV AX, BX
MOV BX, DX
INC SI
DEC CL
JNZ L1
END
```




emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
01 MOV SI, 1000H
02 MOV CL, 10H
03 MOV [SI], 00H
04 MOV AX, [SI]
05 INC SI
06 DEC CL
07 MOV [SI], 01H
08 MOV BX, [SI]
09 IN
10 DE
11 L1
12 MO
13 MO
14 MO
15 MO
16 IN
17 DE
18 JN
19 EN
20
original source code
01 MOV SI, 1000H
02 MOV CL, 10H
03 MOV [SI], 00H
04 MOV AX, [SI]
05 INC SI
06 DEC CL
07 MOV [SI], 01H
08 MOV BX, [SI]
09 INC SI
10 DEC CL
11 L1: ADD AX, BX
12 MOV DX, AX
emulator: noname.bin
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 00 00
BX 00 00
CX 00 00
DX 00 00
CS 0100
IP 0000
SS 0100
SP FFFE
0100:0000
01000: BE 190
01001: 00 000 NULL
01002: 10 016
01003: B1 177
01004: 10 016
01005: C6 198
01006: 04 004
01007: 00 000 NULL
01008: 8B 139
01009: 04 004
0100A: 46 070 F
0100B: FE 254
0100C: C9 201
0100D: C6 198
0100E: 04 004
0100F: 01 001
0100:0000
MOV SI, 01000h
MOV CL, 010h
MOV b.[SI], 00h
MOV AX, [SI]
INC SI
DEC CL
MOV b.[SI], 01h
MOV BX, [SI]
INC SI
DEC CL
ADD AX, BX
MOV DX, AX
MOV [SI], AX
MOV AX, BX
MOV BX, DX
INC SI
Random Access Memory
0100:1000
update
table
list
0100:1000 00 01 01 02 03 05 08 0D-15 22 37 59 90 E9 79 62
0100:1010 02 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
0100:1020 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
0100:1030 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
0100:1040 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
0100:1050 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
0100:1060 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
0100:1070 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00
line: 20
5:08 PM
3/22/2018

8086 MICROPROCESSOR

1) Write an ALP for adding two 16 bit numbers 0122H, 0127H stored in register B and C respectively.

```
MOV BX, 0122H
MOV CX, 0127H
MOV AX, BX
ADD AX, CX
END
```

2) Write an ALP for adding two 8 bit numbers 02H, 1BH stored in register B and C.

```
MOV BX, 02H
MOV CX, 1BH
MOV AX, BX
ADD AX, CX
END
```

3) Write an ALP for subtracting two 8 bit number stored in register.

```
MOV BX, 08H
MOV CX, 02H
MOV AX, BX
SUB AX, CX
```

4.) Write an ALP for storing the 16 bit data in memory address 1000 and 1001. Then perform addition of these two numbers & store the result in memory address 1002

```
MOV SI, 1000H
MOV AL, [SI]
INC SI
MOV BL, [SI]
ADD AL, BL
INC SI
MOV [SI], AL
END
```

5) Write an ALP for performing multiplication and division of two 16 bit numbers.

FOR MULTIPLICATION

```
MOV AX, 04H
MOV BX, 08H
MUL BX
END
```

FOR DIVISION

```
MOV AL, 08H
MOV BL, 02H
DIV BL
```

6) Write an ALP for transferring a block of ten 16 bit data from memory address starting at 1000 to a memory location 3000.

```
MOV SI, 1000H
MOV DI, 3000H
MOV CX, 10H
AGAIN: MOV AX, [SI]
       MOV [DI], AX
       INC SI
       INC DI
       DEC CL
       JNZ AGAIN
END
```

7) Write an ALP to display 13 number of the fibonacci series in a memory location

```
MOV SI, 1000H
MOV CL, 10H
MOV [SI], 00H
MOV AX, [SI]
INC SI
DEC CL
MOV [SI], 01H
MOV BX, [SI]
INC SI
DEC CL
L1: ADD AX, BX
```