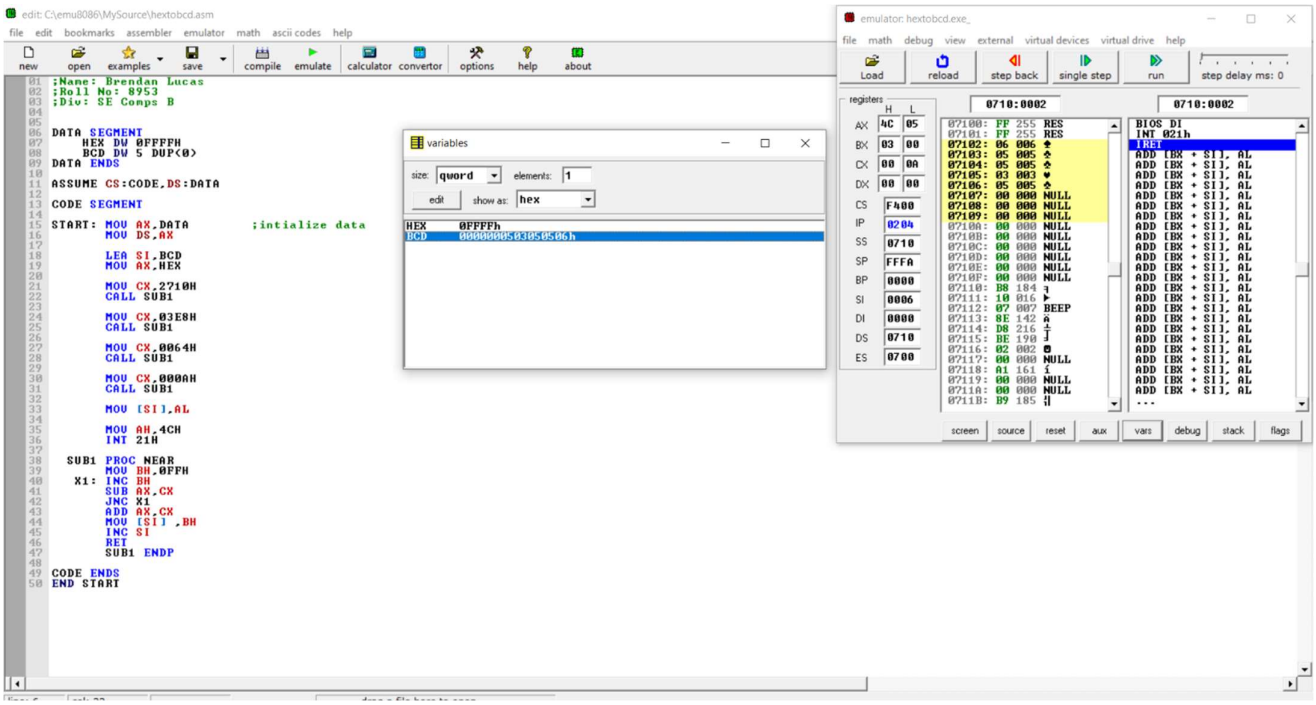
Name: Brendan Lucas, Div: SE COMP B, Roll No: 8953

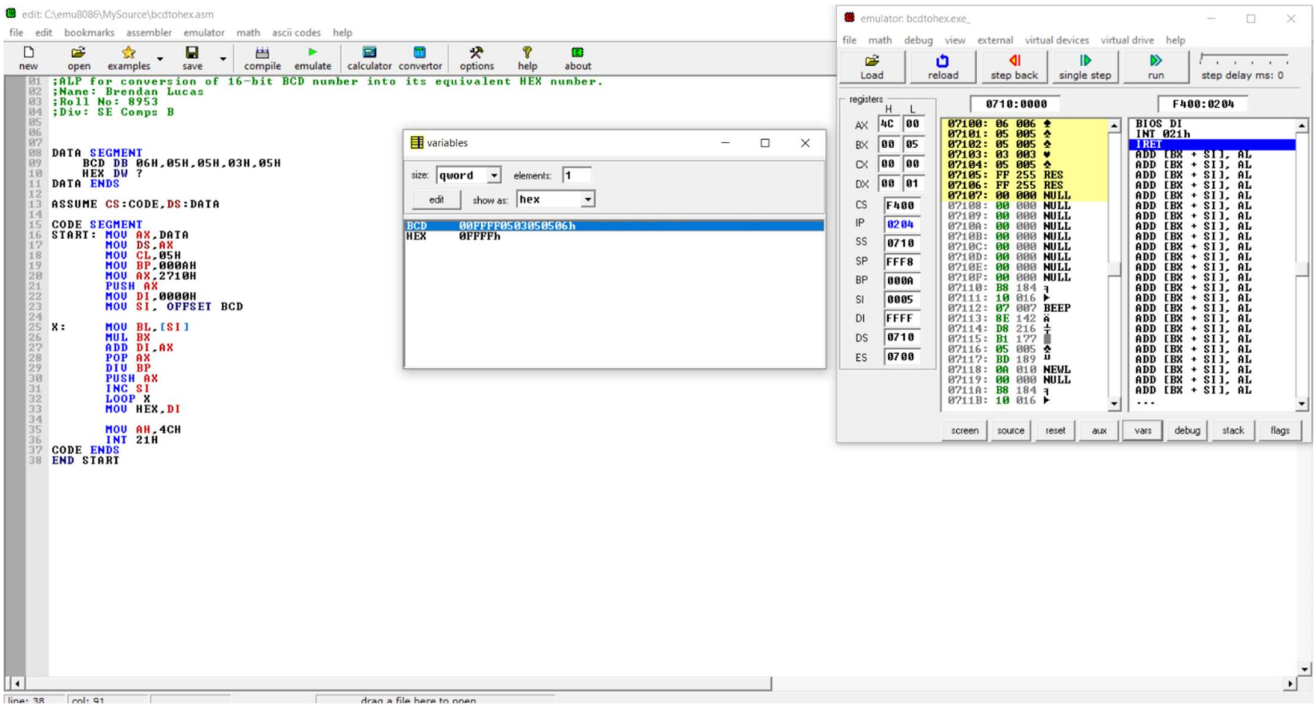
Micro Processor Practical- 4.

1. a. HEX TO BCD:



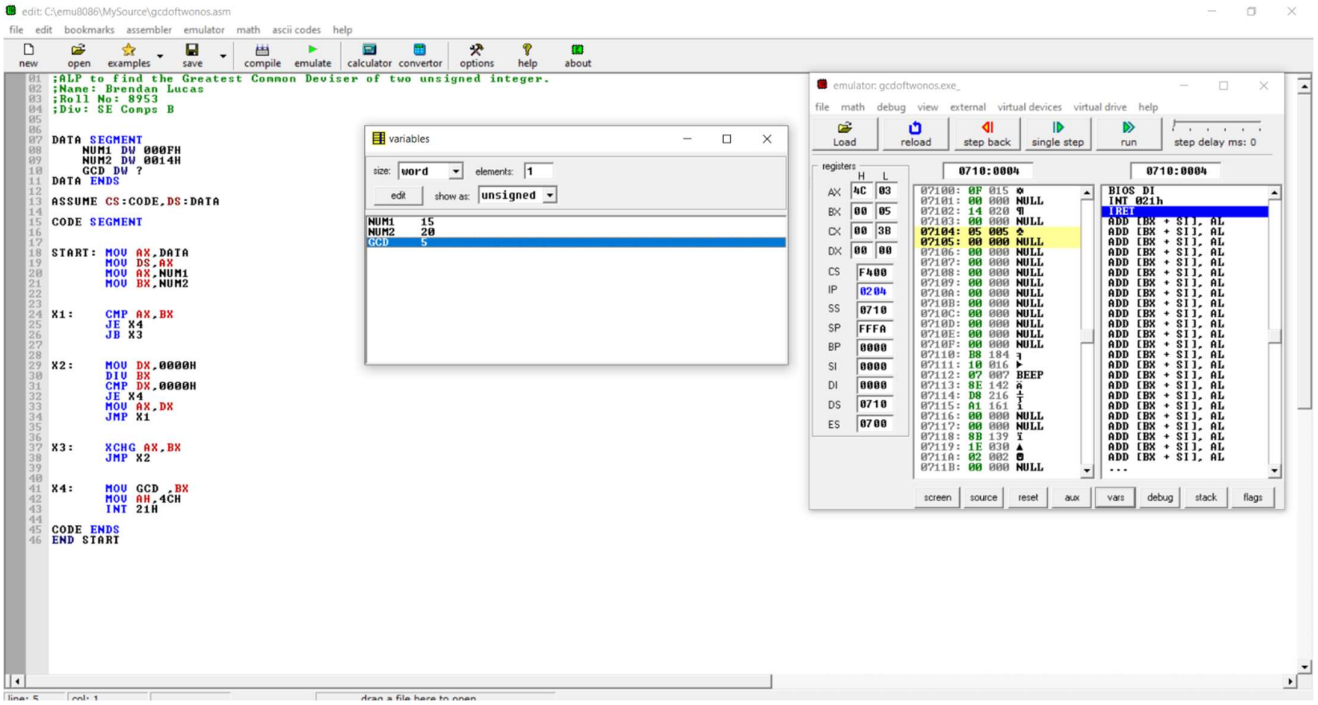
```
;Name: Brendan Lucas
;Roll No: 8953
;Div: SE Comps B

DATA SEGMENT
    HEX DW 0FFFFH
    BCD DW 5 DUP(0)
DATA ENDS

ASSUME CS:CODE,DS:DATA

CODE SEGMENT

START: MOV AX,DATA          ;intialize data
       MOV DS,AX

       LEA SI,BCD
       MOV AX,HEX

       MOV CX,2710H
       CALL SUB1

       MOV CX,03E8H
       CALL SUB1

       MOV CX,0064H
       CALL SUB1

       MOV CX,000AH
       CALL SUB1

       MOV [SI],AL

       MOV AH,4CH
       INT 21H

  SUB1 PROC NEAR
       MOV BH,0FFH
    X1: INC BH
       SUB AX,CX
       JNC X1
       ADD AX,CX
       MOV [SI],BH
       INC SI
       RET
       SUB1 ENDP

CODE ENDS
END START
```

1. b. BCD TO HEX:

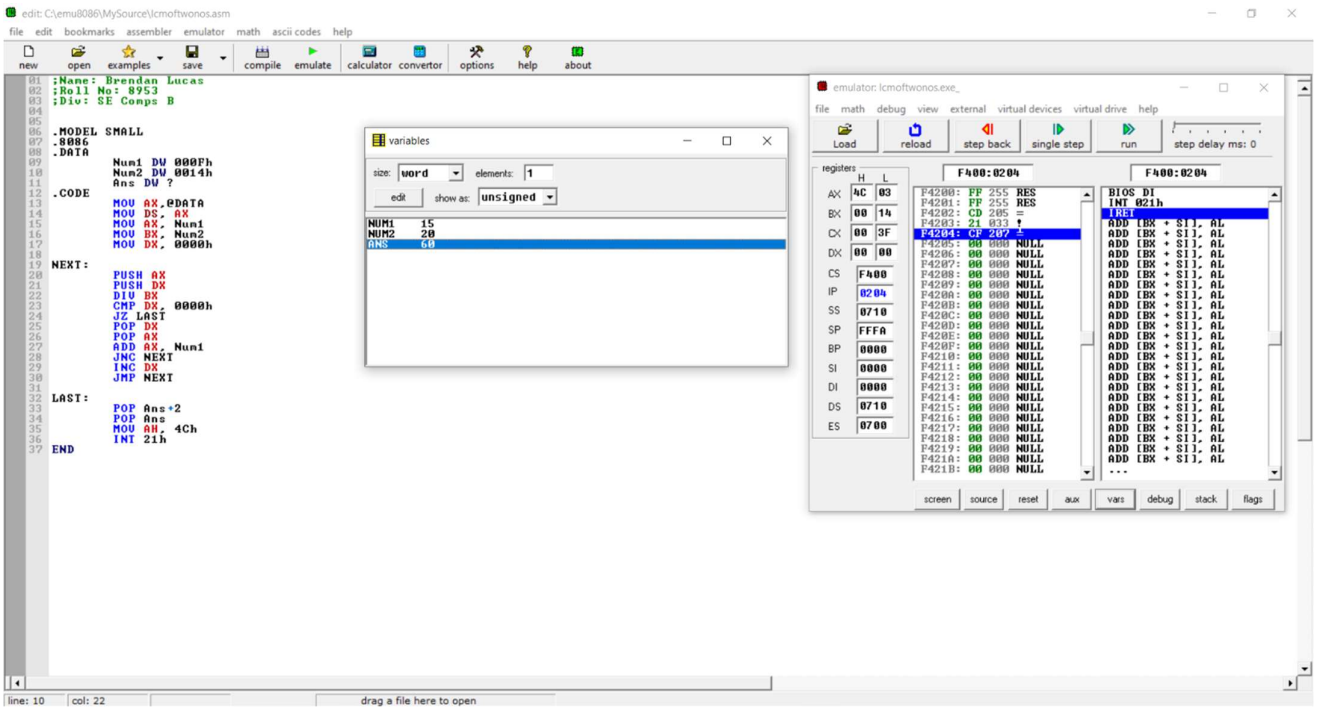


```
;ALP for conversion of 16-bit BCD number into its equivalent HEX number.
;Name: Brendan Lucas
;Roll No: 8953
;Div: SE Comps B


DATA SEGMENT
    BCD DB 06H,05H,05H,03H,05H
    HEX DW ?
DATA ENDS

ASSUME CS:CODE,DS:DATA

CODE SEGMENT
START: MOV AX,DATA
       MOV DS,AX
       MOV CL,05H
       MOV BP,000AH
       MOV AX,2710H
       PUSH AX
       MOV DI,0000H
       MOV SI, OFFSET BCD

X:     MOV BL,[SI]
       MUL BX
       ADD DI,AX
       POP AX
       DIV BP
       PUSH AX
       INC SI
       LOOP X
       MOV HEX,DI

       MOV AH,4CH
       INT 21H
CODE ENDS
END START
```

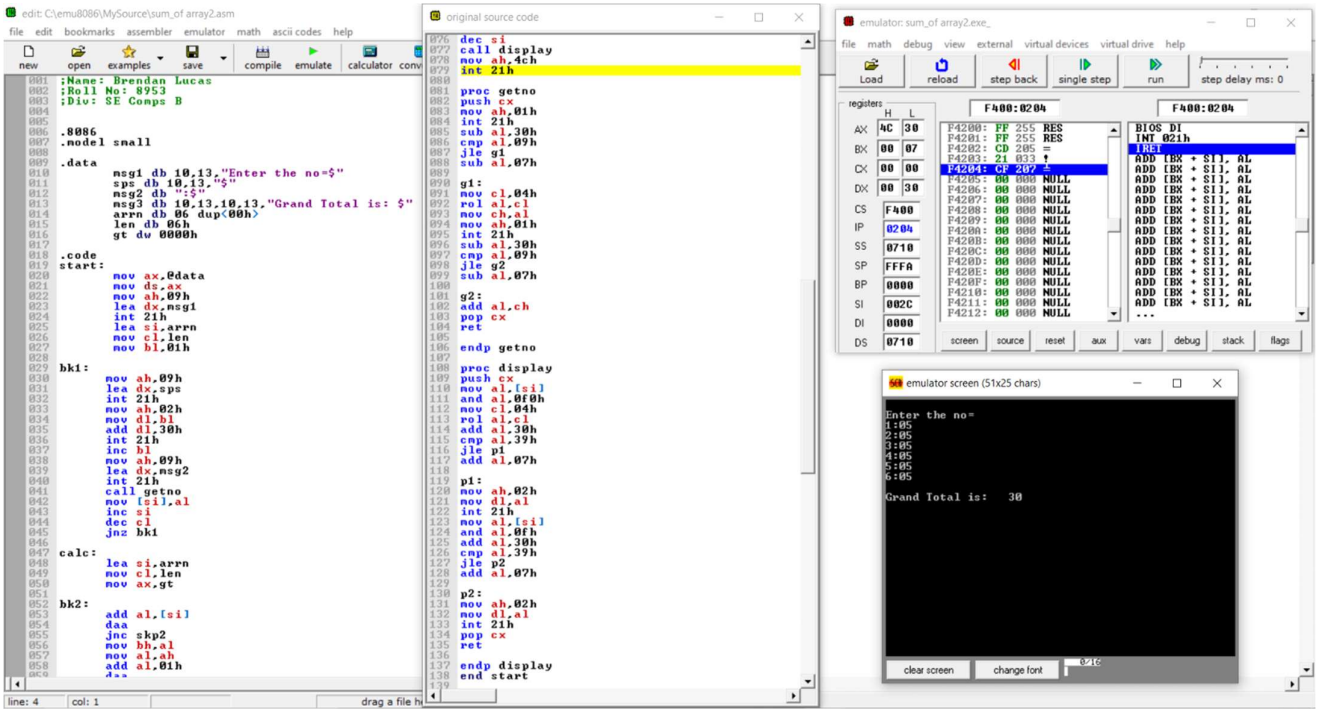
2. a. Find GCD of two numbers



2. b. Find LCM of two numbers



3. Write a Program for addition of numbers taken from keyboard and display result on the screen



```
;Name: Brendan Lucas
;Roll No: 8953
;Div: SE Comps B


.8086
.model small

.data
        msg1 db 10,13,"Enter the no=$"
        sps db 10,13,"$"
        msg2 db ":$"
        msg3 db 10,13,10,13,"Grand Total is: $"
        arrn db 06 dup(00h)
        len db 06h
        gt dw 0000h

.code
start:
        mov ax,@data
        mov ds,ax
        mov ah,09h
        lea dx,msg1
        int 21h
        lea si,arrn
        mov cl,len
        mov bl,01h

bk1:
        mov ah,09h
        lea dx,sps
        int 21h
        mov ah,02h
        mov dl,bl
        add dl,30h
        int 21h
        inc bl
        mov ah,09h
        lea dx,msg2
        int 21h
        call getno
        mov [si],al
        inc si
        dec cl
        jnz bk1

calc:
        lea si,arrn
        mov cl,len
        mov ax,gt

bk2:
        add al,[si]
        daa
        jnc skp2
        mov bh,al
        mov al,ah
        add al,01h
        daa
```

```
dec si
call display
mov ah,4ch
int 21h

proc getno
push cx
mov ah,01h
int 21h
sub al,30h
cmp al,09h
jle g1
sub al,07h

g1:
mov cl,04h
rol al,cl
mov ch,al
mov ah,01h
int 21h
sub al,30h
cmp al,09h
jle g2
sub al,07h

g2:
add al,ch
pop cx
ret

endp getno

proc display
push cx
mov al,[si]
and al,0f0h
mov cl,04h
rol al,cl
add al,30h
cmp al,39h
jle p1
add al,07h

p1:
mov ah,02h
mov dl,al
int 21h
mov al,[si]
and al,0fh
add al,30h
cmp al,39h
jle p2
add al,07h

p2:
mov ah,02h
mov dl,al
int 21h
pop cx
ret

endp display
end start
```

```
Enter the no=
1:05
2:05
3:05
4:05
5:05
6:05

Grand Total is:   30
```