

(ASSEMBLY LAB ASSIGNMENT)

(SUBMITTED BY ALI AKBER BSCSF22R036)

(BSCS 4TH SS1)

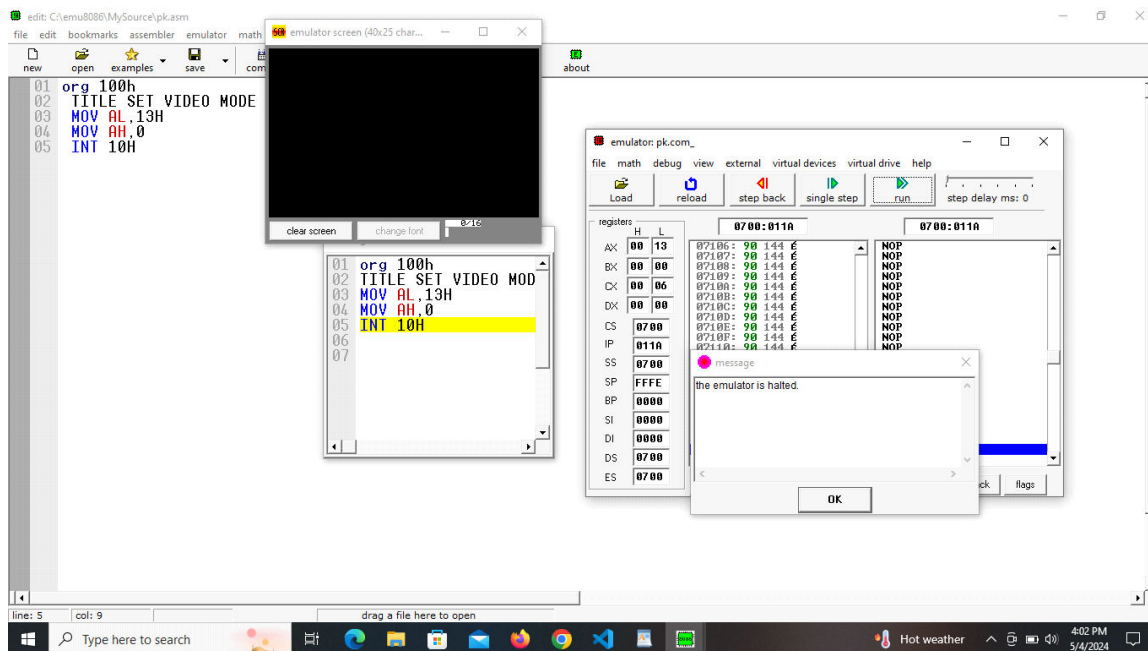
org 100h

TITLE SET VIDEO MODE

MOV AL,13H

MOV AH,0

INT 10H



org 100h

TITLE HIDE BLINKING TEXT CURSOR

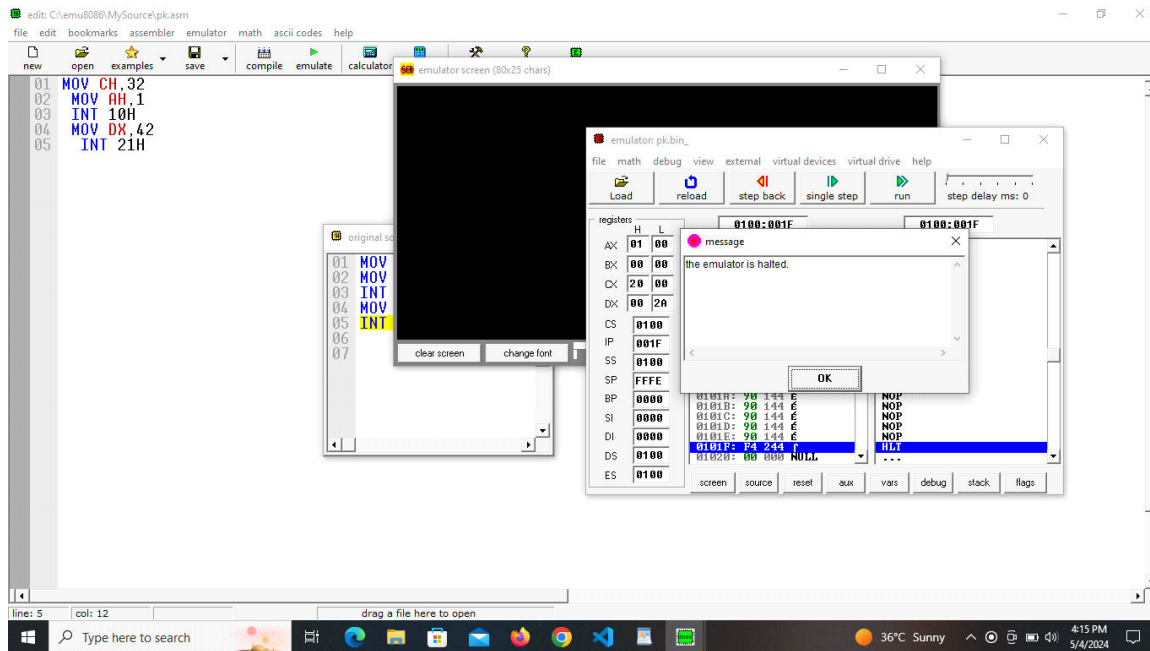
MOV CH,32

MOV AH,1

INT 10H

MOV DX,42

INT 21H



org 100h

TITLE STANDARD BLINKING TEXT CURSOR

MOV CH,6

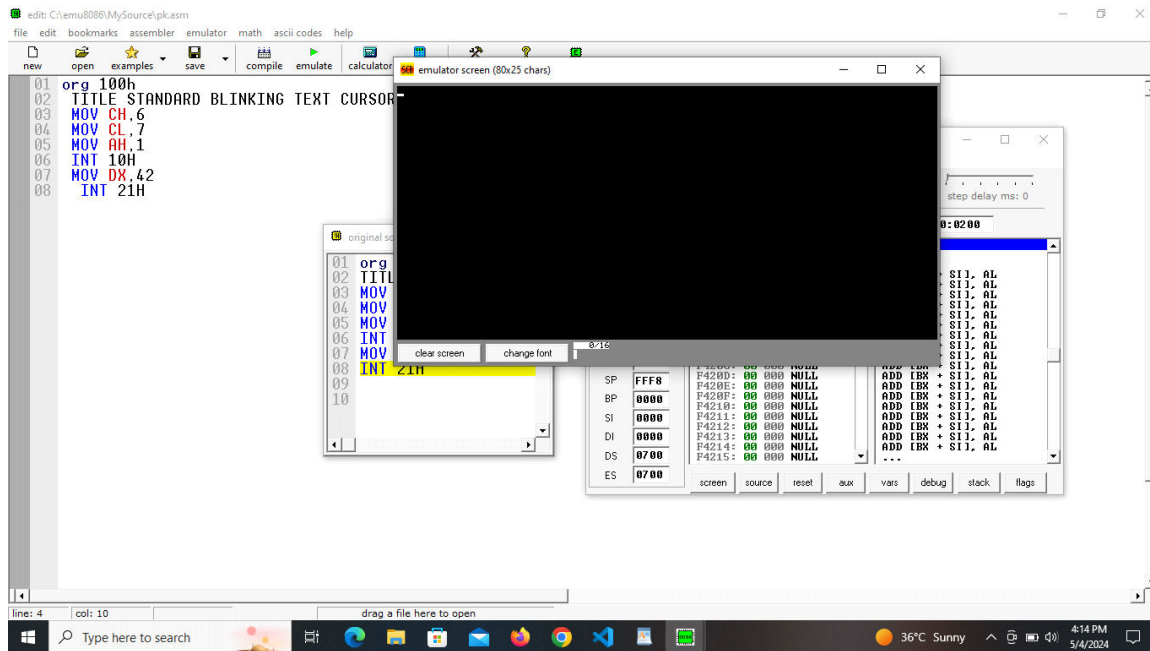
MOV CL,7

MOV AH,1

INT 10H

MOV DX,42

INT 21H



org 100h

TITLE BOX SHAPED BLINKING TEXT CURSOR

MOV CH,0

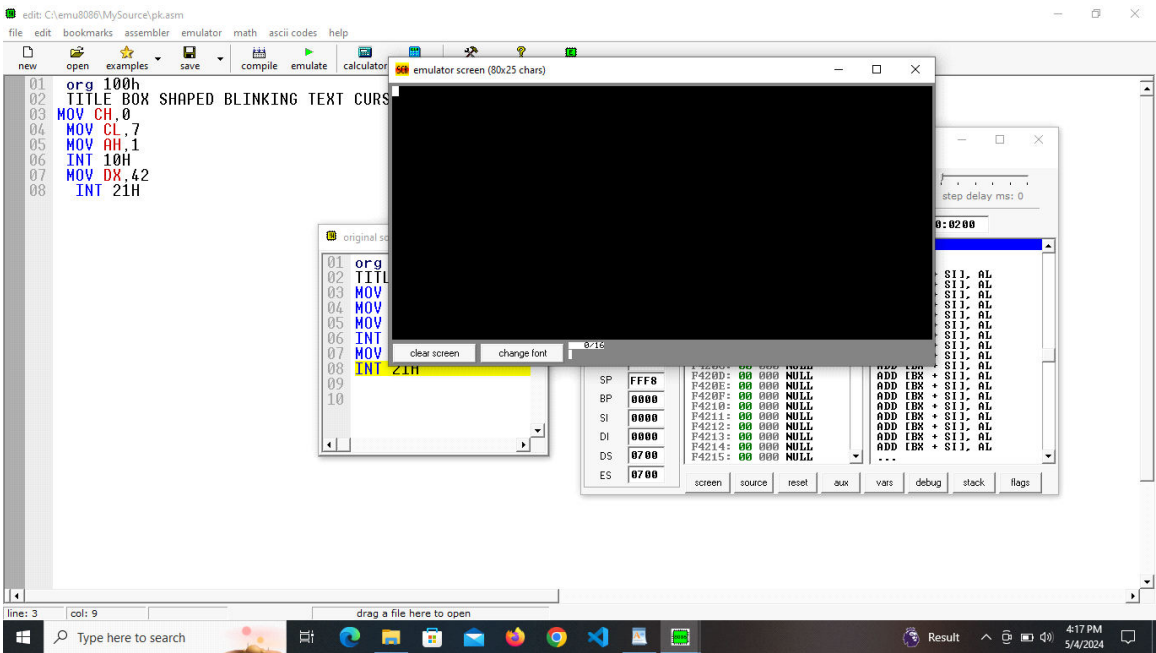
MOV CL,7

MOV AH,1

INT 10H

MOV DX,42

INT 21H



```
org 100h
```

TITLE SET CURSOR POSITION

MOV DH,10

MOV DL,20

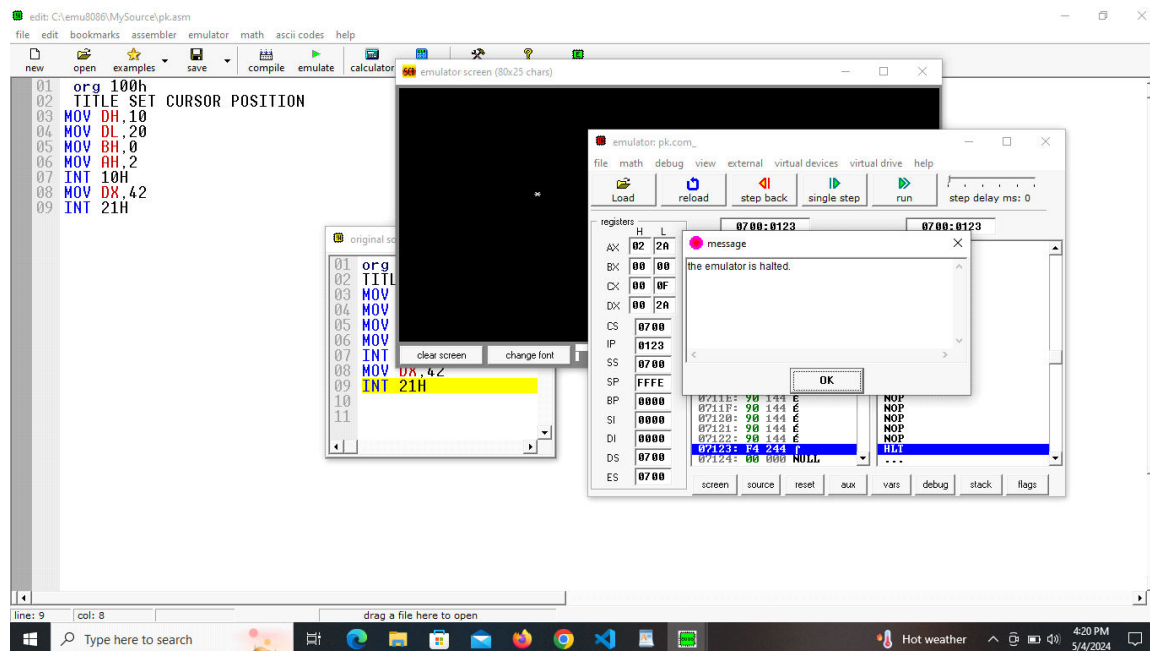
MOV BH,0

MOV AH,2

INT 10H

MOV DX,42

INT 21H



Change color for a single pixel.

mov al, 13h

mov ah, 0

int 10h ; set graphics video mode.

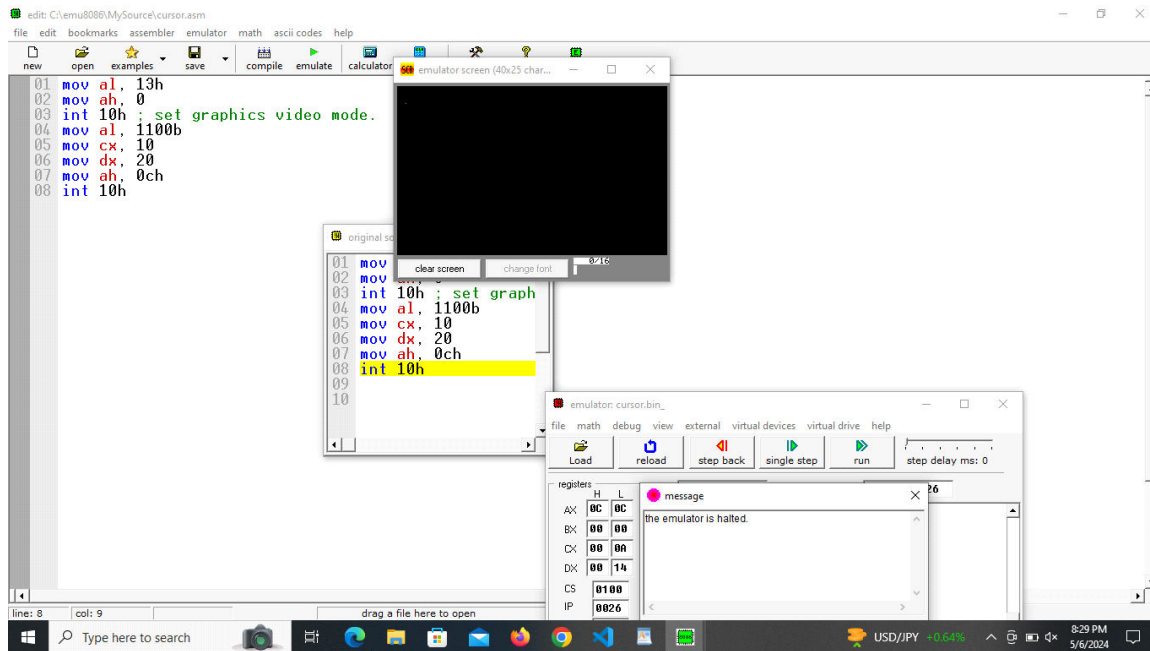
mov al, 1100b

mov cx, 10

mov dx, 20

mov ah, 0ch

int 10h

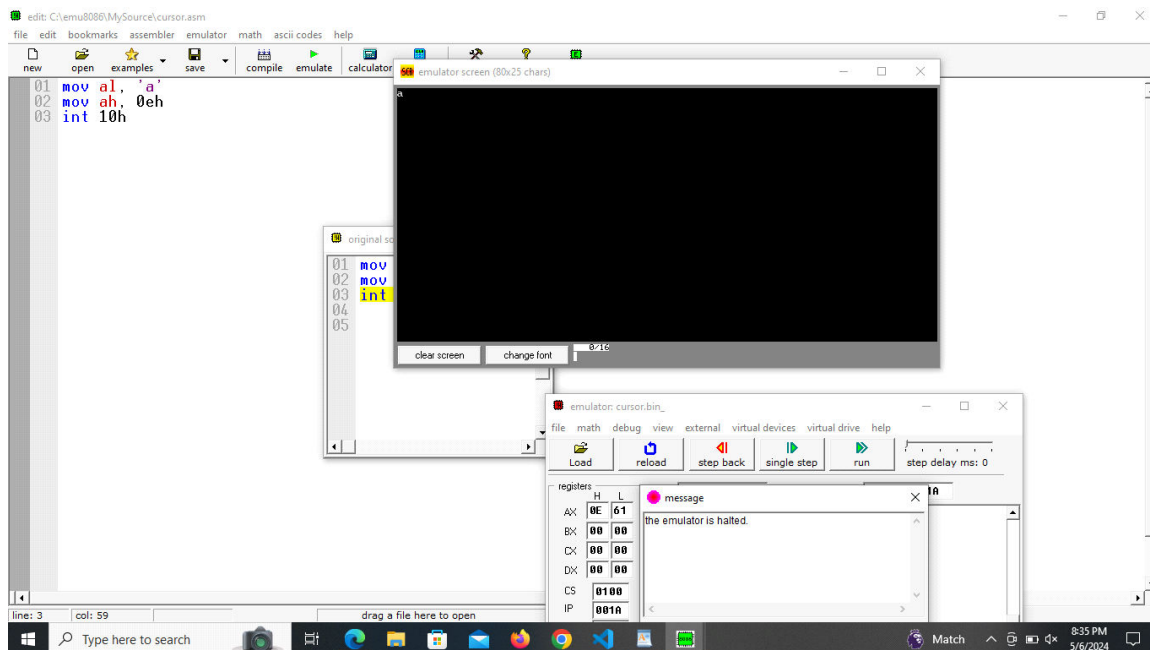


Teletype output.

mov al, 'a'

mov ah, 0eh

int 10h



Write string.

```

mov al, 1

mov bh, 0

mov bl, 0011_1011b

mov cx, msg1end - offset msg1 ; calculate message size.

mov dl, 10

mov dh, 7

push cs

pop es

mov bp, offset msg1

mov ah, 13h

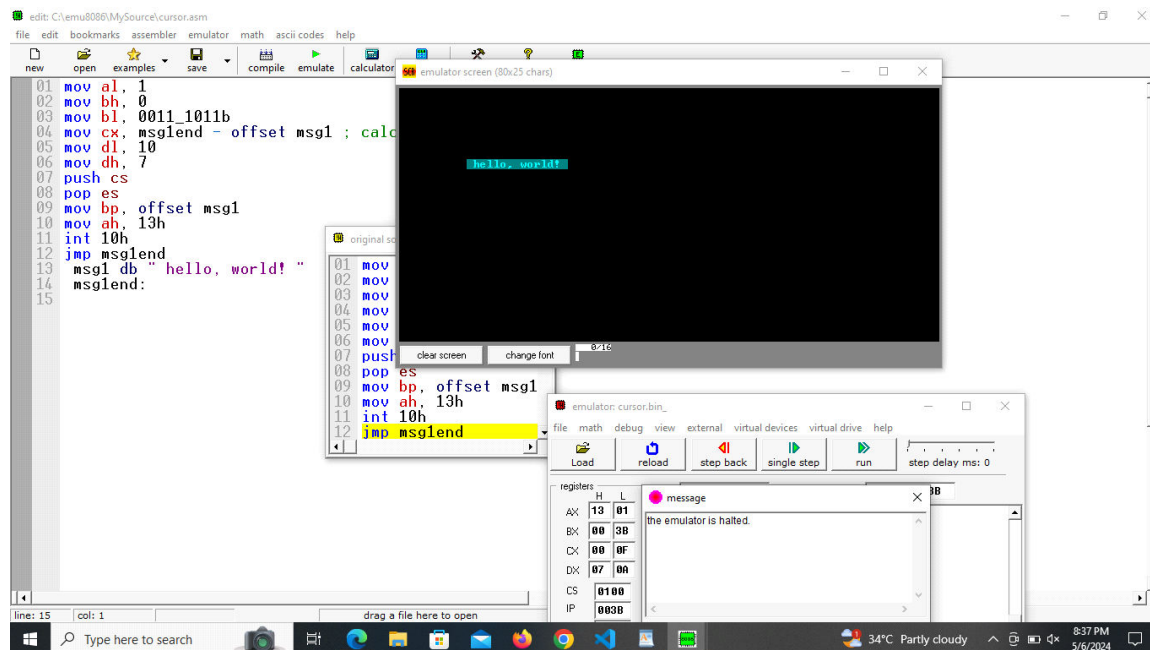
int 10h

jmp msg1end

    msg1 db " hello, world! "

msg1end:

```



Toggle intensity/blinking.

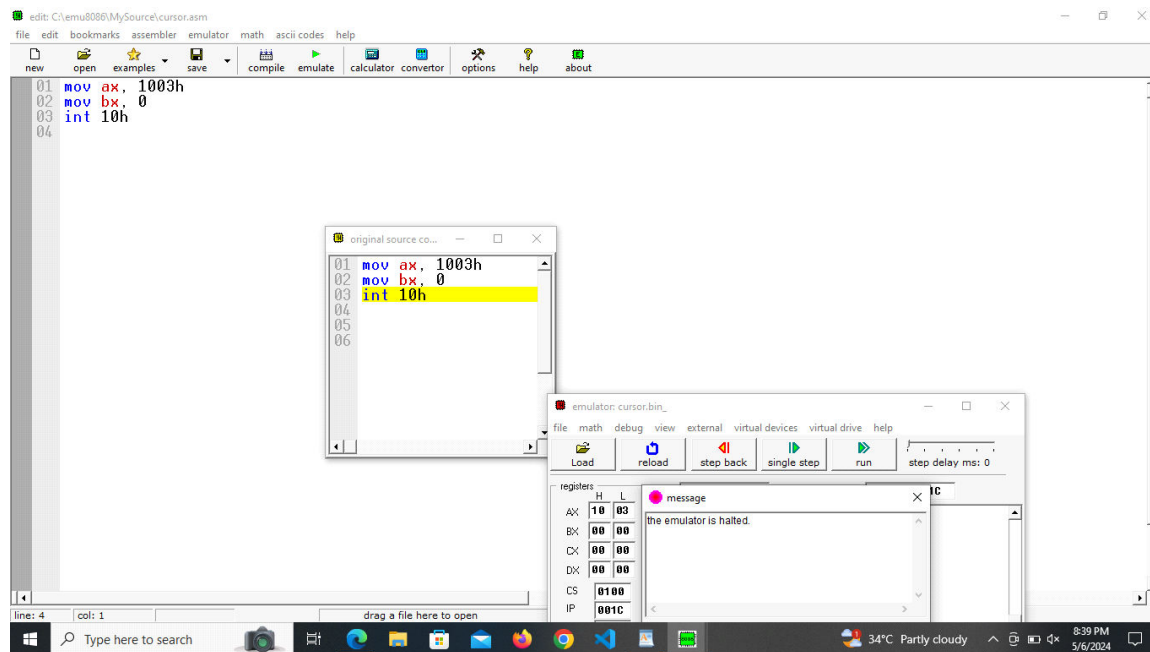
```

mov ax, 1003h

mov bx, 0

```

int 10h

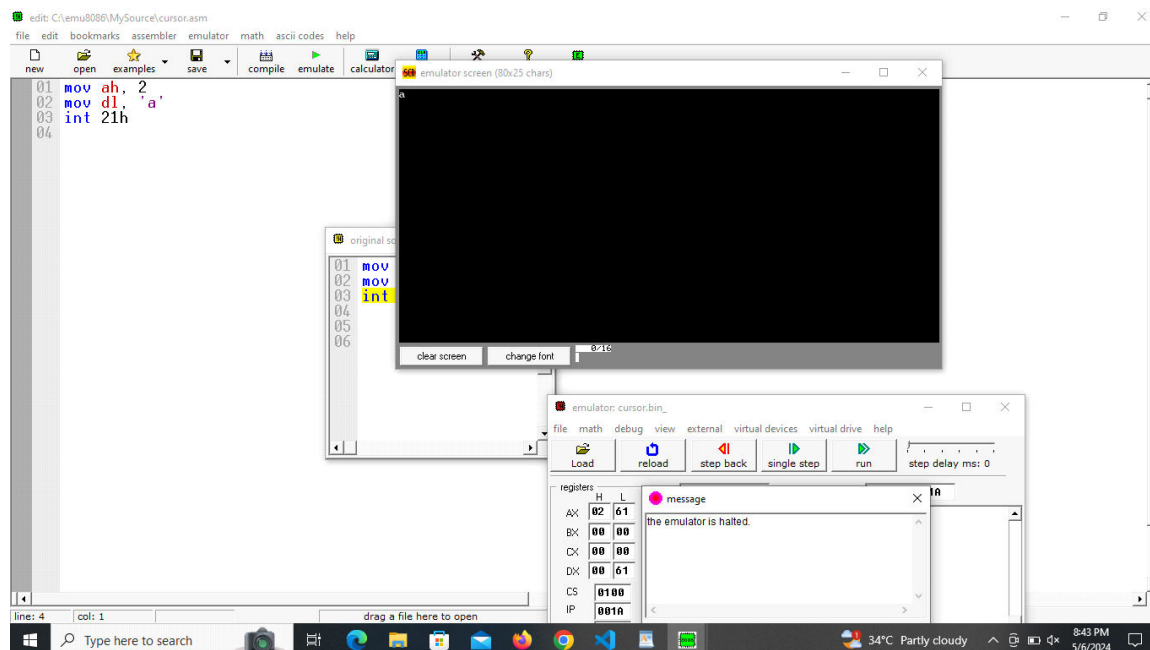


Write character to standard output.

mov ah, 2

mov dl, 'a'

int 21h

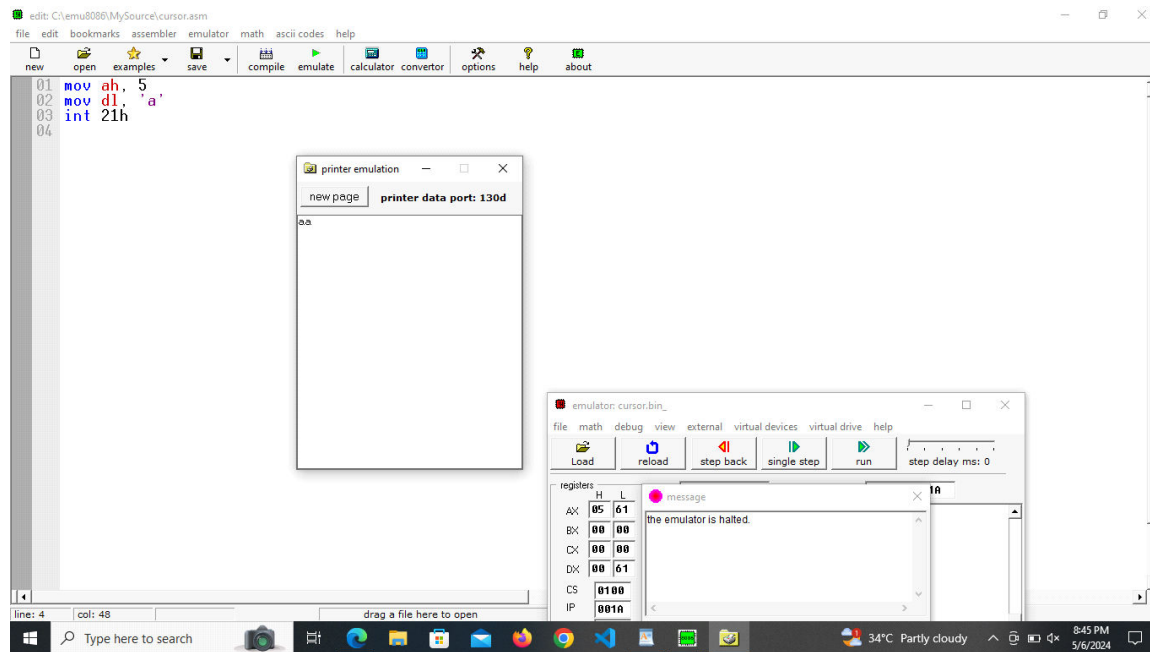


Output character to printer

mov ah, 5

mov dl, 'a'

int 21h



Direct console input or output.

mov ah, 6

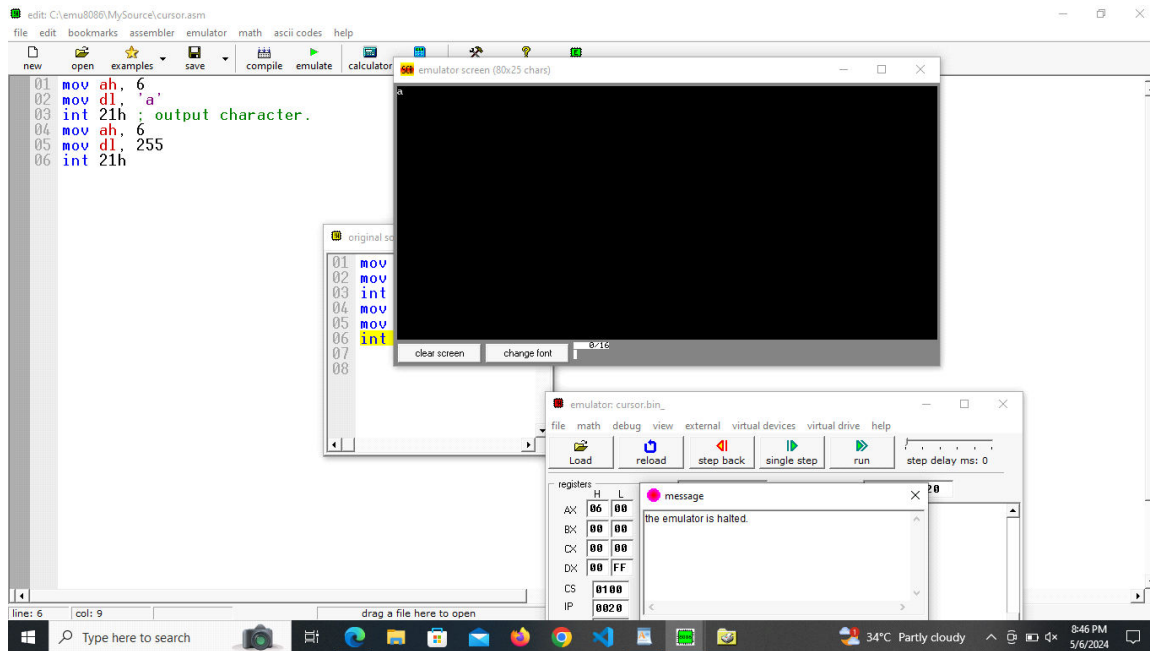
mov dl, 'a'

int 21h ; output character.

mov ah, 6

mov dl, 255

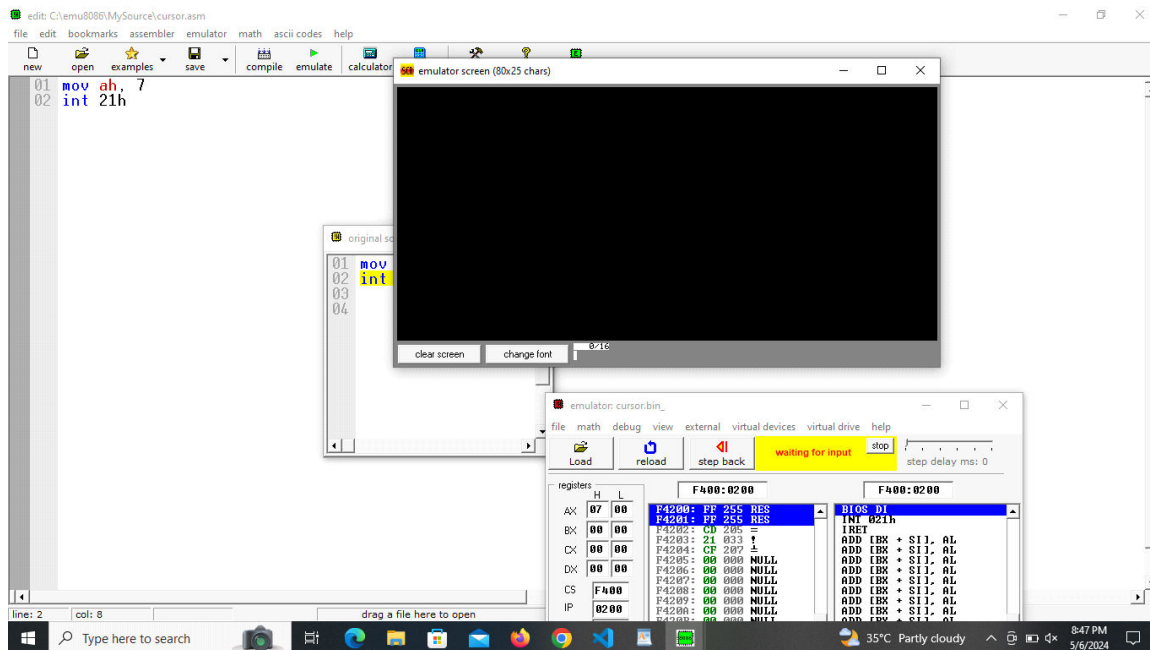
int 21h



Character input without echo to AL.

mov ah, 7

int 21h



Output of a string at DS:DX

org 100h

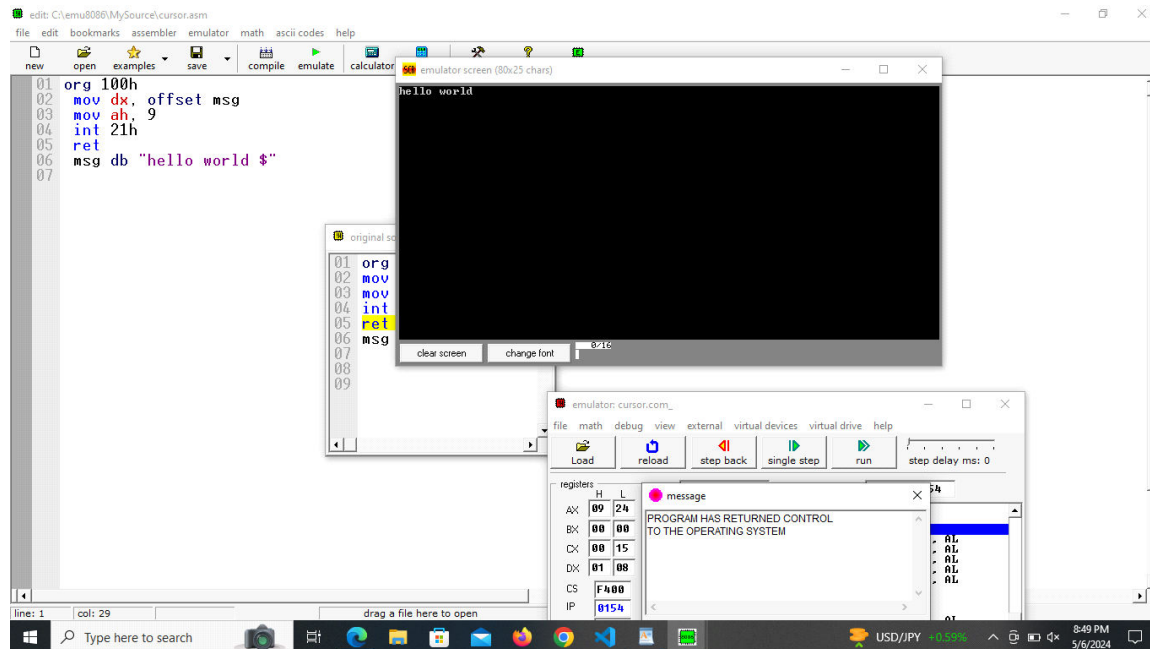
mov dx, offset msg

mov ah, 9

int 21h

ret

msg db "hello world \$"



Input of a string to DS:DX

org 100h

mov dx, offset buffer

mov ah, 0ah

int 21h

jmp print

buffer db 10,?, 10 dup(' ')

print:

xor bx, bx

mov bl, buffer[1]

mov buffer[bx+2], '\$'

mov dx, offset buffer + 2

mov ah, 9

int 21h

ret

