



Department of Computer Engineering

Digital Hardware Systems

CpeA 3104 - Microprocessors

Laboratory Report

| | | | |
|----------------------------|--------------------------------|-------------------|------------|
| Laboratory Exercise No: | 36 | Date Performed: | 09/23/2020 |
| Laboratory Exercise Title: | Assembly Language Instructions | | |
| Name of Student : | Felisarta, German III | Document Version: | v1.0 |

ACTIVITY #6.1

CODE

```
ORG 100H

MENU db "MENU $"
CHOICEONE db "1-HORIZONTAL STRIPES $"
CHOICETWO db "2-VERTICAL STRIPES $"
CHOICETHREE db "F1-CHECKERED PATTERN $"
QUIT db "Q-QUIT $"
INPUTCHOICE db "ENTER CHOICE $"
PRESSKEY db "PRESS ANY KEY TO CONTINUE $"

MAIN:

    CALL CLEAR_SCREEN
    CALL ENABLEMOUSE

    MOV BH, 1EH
    MOV CH, 0    ; dimensions which to color
    MOV CL, 0
    MOV DH, 24
    MOV DL, 79
    CALL SET_BACKGROUND
```

```
CALL MAINMENU  
CALL CHECKINPUT  
CALL CHECK
```

```
RET
```

```
ENABLEMOUSE:
```

```
MOV AX, 0  
INT 33H
```

```
MOV AX, 1  
INT 33H
```

```
RET
```

```
CHECKINPUT:      ;GET MOUSE STATE.  
MOV AX, 3  
INT 33h          ; CHECK LEFT BUTTON STATE.  
MOV AX, BX  
AND AX, 01B      ; LEFT BUTTON
```

```
JZ CHECK_KEY  
RET
```

```
CHECK_KEY:      ;CHECK IF A KEY WAS PRESSED.
```

```
MOV DH, 10  
MOV DL, 43  
CALL SET_CURSOR_POS ;SET CURSOR POSITION
```

```
MOV ah, 1H  
INT 16H
```

```
JZ CHECKINPUT  ; NO KEY.  
RET
```

```
CLEAR_SCREEN:
```

```
MOV AH, 00H      ;INITIALIZE SCREEN  
MOV AL, 03H  
INT 10H
```

```
RET
```

```
SET_BACKGROUND:
```

```
MOV AH, 06H    ; SET BACKGROUND AND FOREGROUND
MOV AL, 00H
INT 10H
RET
```

DISP_MESS:

```
MOV AH, 09H
INT 21H
RET
```

SET_CURSOR_POS:

```
MOV AH, 02H    ; SET CURSOR POSITION
MOV BH, 00H
INT 10H
RET
```

MAINMENU:

```
MOV DH, 2
MOV DL, 35
CALL SET_CURSOR_POS
```

```
MOV DX, OFFSET MENU
CALL DISP_MESS
```

```
;-----
```

```
MOV DH, 5
MOV DL, 0
CALL SET_CURSOR_POS
```

```
MOV DX, OFFSET CHOICEONE
CALL DISP_MESS
```

```
;-----
```

```
MOV DH, 6
MOV DL, 0
CALL SET_CURSOR_POS
```

```
MOV DX, OFFSET CHOICETWO
CALL DISP_MESS
```

```
;-----
```

```
MOV DH, 7
MOV DL, 0
CALL SET_CURSOR_POS
```

```
MOV DX, OFFSET CHOICETHREE
CALL DISP_MESS
```

```
;-----
```

```
MOV DH, 10
MOV DL, 0
CALL SET_CURSOR_POS
```

```
MOV DX, OFFSET QUIT
CALL DISP_MESS
```

```
;-----
```

```
MOV DH, 13
MOV DL, 30
CALL SET_CURSOR_POS
```

```
MOV DX, OFFSET INPUTCHOICE
CALL DISP_MESS
```

```
RET
```

CHECK:

```
CMP AL, 31H ;KEYPRESS
JZ HORIZONTAL
```

```
CMP DL, 29H ;MOUSE CLICK
JZ HORIZONTAL
CMP DL, 2AH ;MOUSE CLICK
JZ HORIZONTAL
CMP DL, 2CH ;MOUSE CLICK
JZ HORIZONTAL
CMP DL, 2DH ;MOUSE CLICK
JZ HORIZONTAL
CMP DL, 2EH ;MOUSE CLICK
JZ HORIZONTAL
CMP DL, 2FH ;MOUSE CLICK
JZ HORIZONTAL
```

```
;-----
```

```
CMP AL, 32H ;KEYPRESS  
JZ VERTICAL
```

```
CMP DL, 30H ;MOUSE CLICK  
JZ VERTICAL  
CMP DL, 31H ;MOUSE CLICK  
JZ VERTICAL  
CMP DL, 32H ;MOUSE CLICK  
JZ VERTICAL  
CMP DL, 33H ;MOUSE CLICK  
JZ VERTICAL  
CMP DL, 34H ;MOUSE CLICK  
JZ VERTICAL  
CMP DL, 35H ;MOUSE CLICK  
JZ VERTICAL  
CMP DL, 36H ;MOUSE CLICK  
JZ VERTICAL  
CMP DL, 37H ;MOUSE CLICK  
JZ VERTICAL
```

```
;-----
```

```
CMP AH, 3BH ;KEYPRESS  
JZ CHECKERED
```

```
CMP DL, 38H ;MOUSE CLICK  
JZ CHECKERED  
CMP DL, 39H ;MOUSE CLICK  
JZ CHECKERED  
CMP DL, 3AH ;MOUSE CLICK  
JZ CHECKERED  
CMP DL, 3BH ;MOUSE CLICK  
JZ CHECKERED  
CMP DL, 3CH ;MOUSE CLICK  
JZ CHECKERED  
CMP DL, 3DH ;MOUSE CLICK  
JZ CHECKERED  
CMP DL, 3EH ;MOUSE CLICK  
JZ CHECKERED  
CMP DL, 3FH ;MOUSE CLICK  
JZ CHECKERED
```

```
;-----
```

```
CMP AL, "q" ;KEYPRESS  
JZ END_PROG
```

```
CMP AL, "Q" ;KEYPRESS
```

JZ END_PROG

CMP DL, 50H ;MOUSE CLICK

JZ END_PROG

CMP DL, 51H ;MOUSE CLICK

JZ END_PROG

CMP DL, 52H ;MOUSE CLICK

JZ END_PROG

CMP DL, 53H ;MOUSE CLICK

JZ END_PROG

CMP DL, 54H ;MOUSE CLICK

JZ END_PROG

CMP DL, 55H ;MOUSE CLICK

JZ END_PROG

CMP DL, 56H ;MOUSE CLICK

JZ END_PROG

CMP DL, 57H ;MOUSE CLICK

JZ END_PROG

HORIZONTAL:

MOV BH, 00H

MOV CH, 0 ; dimensions which to color

MOV CL, 0

MOV DH, 6

MOV DL, 79

CALL SET_BACKGROUND

MOV BH, 11011111B

MOV CH, 7 ; dimensions which to color

MOV CL, 0

MOV DH, 12

MOV DL, 79

CALL SET_BACKGROUND

MOV BH, 11101111B

MOV CH, 13 ; dimensions which to color

MOV CL, 0

MOV DH, 17

MOV DL, 79

CALL SET_BACKGROUND

MOV BH, 00011111B

MOV CH, 18 ; dimensions which to color

MOV CL, 0

MOV DH, 24

MOV DL, 79

CALL SET_BACKGROUND

```
MOV DH, 20
MOV DL, 30
CALL SET_CURSOR_POS

MOV DX, OFFSET PRESSKEY
CALL DISP_MESS

MOV AH, 0
INT 16H
MOV AH, 7 ;PRESS ANYKEY
INT 21H

CALL CLEAR_SCREEN
CALL MAIN
```

VERTICAL:

```
MOV BH, 00H
MOV CH, 0 ; dimensions which to color
MOV CL, 0
MOV DH, 24
MOV DL, 20
CALL SET_BACKGROUND

MOV BH, 11011111B
MOV CH, 0 ; dimensions which to color
MOV CL, 21
MOV DH, 24
MOV DL, 40
CALL SET_BACKGROUND

MOV BH, 11101111B
MOV CH, 0 ; dimensions which to color
MOV CL, 41
MOV DH, 24
MOV DL, 60
CALL SET_BACKGROUND

MOV BH, 00011111B
MOV CH, 0 ; dimensions which to color
MOV CL, 61
MOV DH, 24
MOV DL, 79
CALL SET_BACKGROUND
```

```
MOV DH, 20
MOV DL, 30
CALL SET_CURSOR_POS

MOV DX, OFFSET PRESSKEY
CALL DISP_MESS

MOV AH, 0
INT 16H
MOV AH, 7 ;PRESS ANYKEY
INT 21H

CALL CLEAR_SCREEN
CALL MAIN
```

CHECKERED:

```
MOV BH, 11011111B
MOV CH, 0 ; dimensions which to color
MOV CL, 0
MOV DH, 6
MOV DL, 79
CALL SET_BACKGROUND
```

```
MOV BH, 00011111B
MOV CH, 7 ; dimensions which to color
MOV CL, 0
MOV DH, 12
MOV DL, 79
CALL SET_BACKGROUND
```

```
MOV BH, 11011111B
MOV CH, 13 ; dimensions which to color
MOV CL, 0
MOV DH, 18
MOV DL, 79
CALL SET_BACKGROUND
```

```
MOV BH, 00011111B
MOV CH, 19 ; dimensions which to color
MOV CL, 0
MOV DH, 24
MOV DL, 79
CALL SET_BACKGROUND
```

```
;-----
```



```
MOV BH, 11101111B
MOV CH, 0 ; dimensions which to color
MOV CL, 0
MOV DH, 24
MOV DL, 10
CALL SET_BACKGROUND
```

```
MOV BH, 00111111B
MOV CH, 0 ; dimensions which to color
MOV CL, 21
MOV DH, 24
MOV DL, 30
CALL SET_BACKGROUND
```

```
MOV BH, 11101111B
MOV CH, 0 ; dimensions which to color
MOV CL, 41
MOV DH, 24
MOV DL, 50
CALL SET_BACKGROUND
```

```
MOV BH, 00111111B
MOV CH, 0 ; dimensions which to color
MOV CL, 61
MOV DH, 24
MOV DL, 69
CALL SET_BACKGROUND
```

```
MOV DH, 20
MOV DL, 30
CALL SET_CURSOR_POS
```

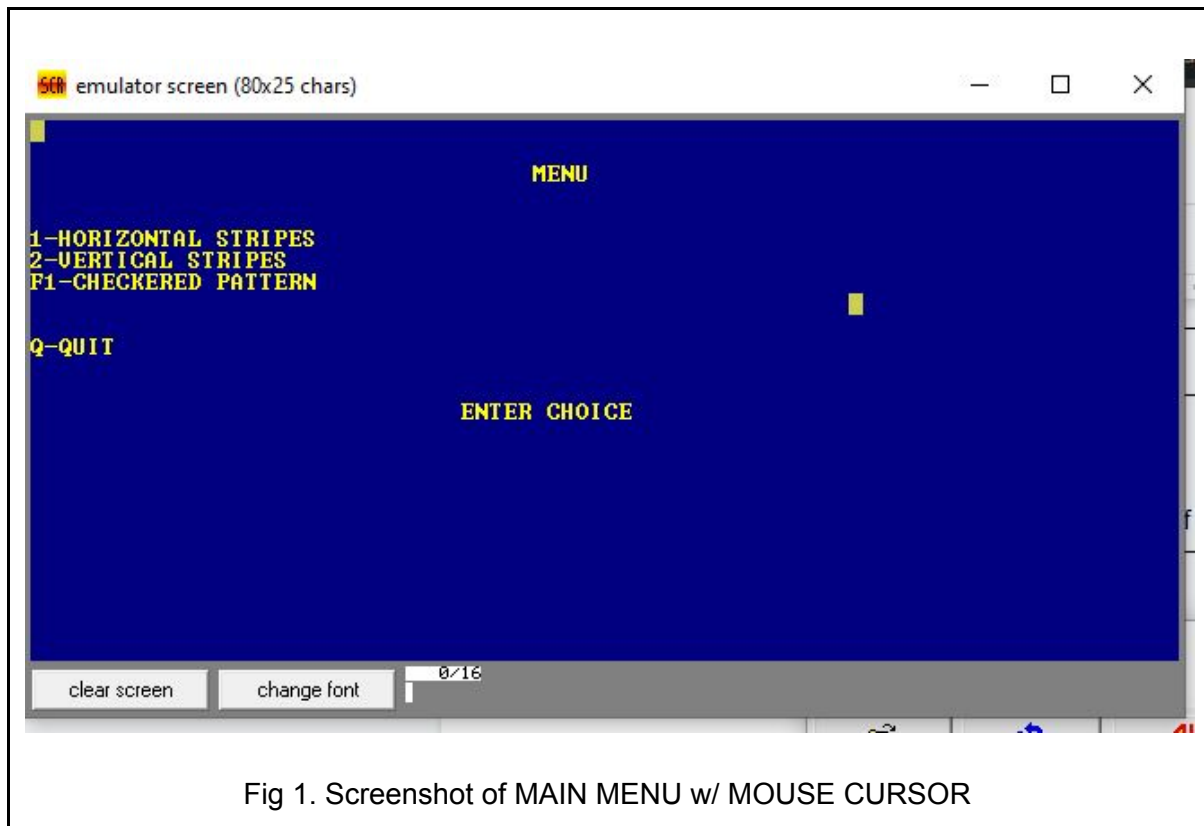
```
MOV DX, OFFSET PRESSKEY
CALL DISP_MESS
```

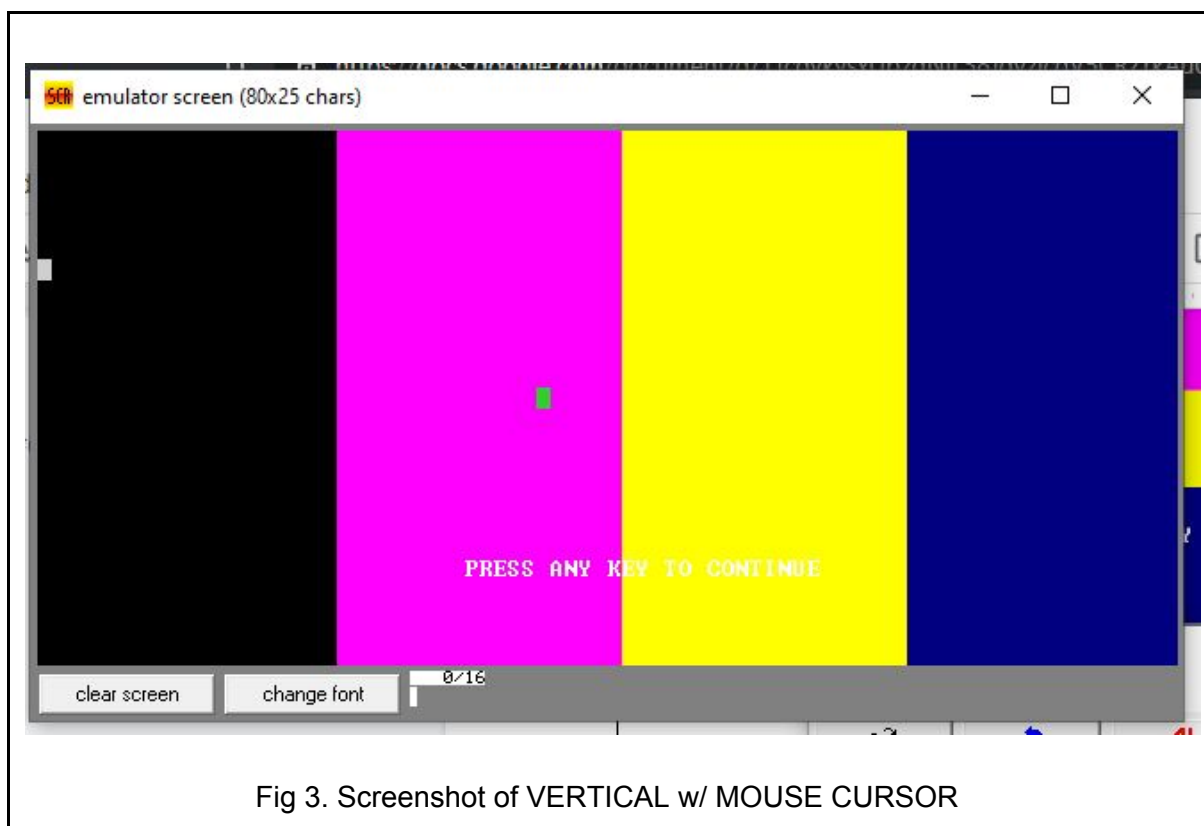
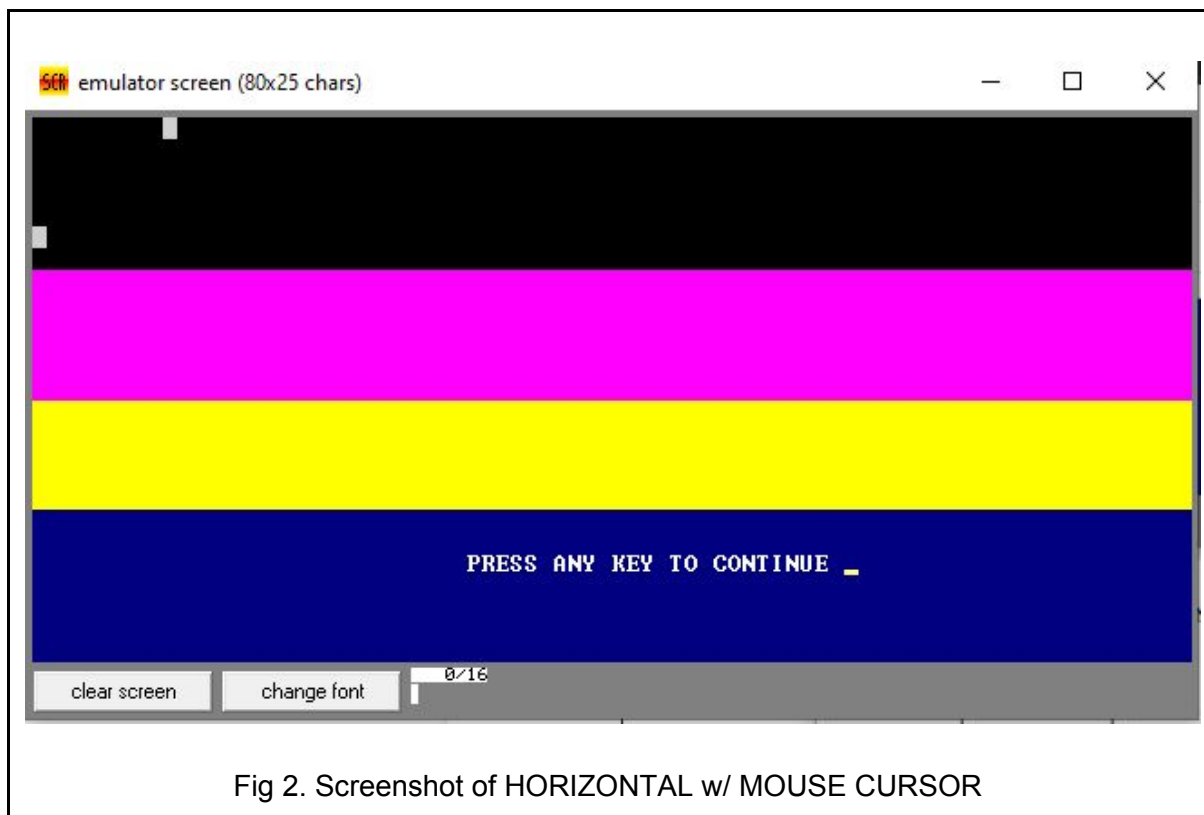
```
MOV AH, 0
INT 16H
MOV AH, 7 ;PRESS ANYKEY
INT 21H
```

```
CALL CLEAR_SCREEN
CALL MAIN
```

END_PROG:

END





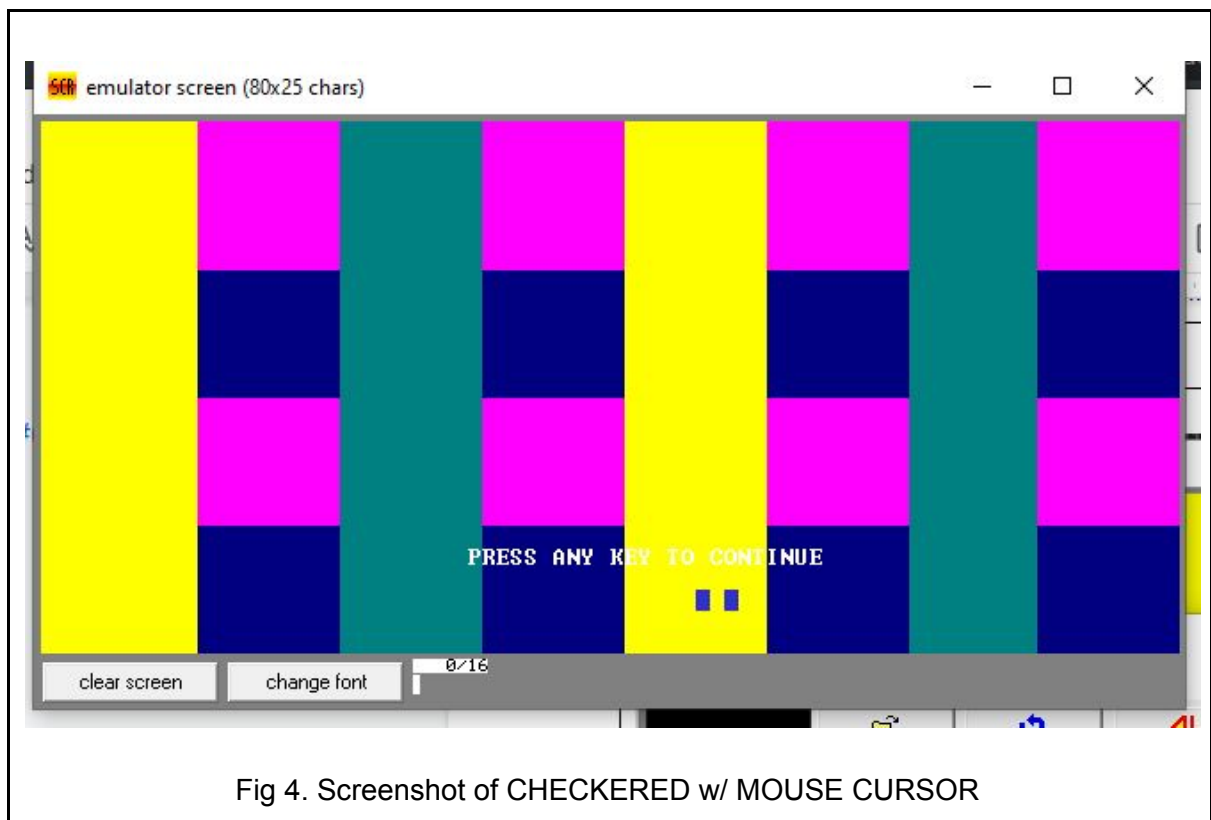


Fig 4. Screenshot of CHECKERED w/ MOUSE CURSOR