

Experiment 08: Mouse Interfacing

Learning Objective: Student should be able to Develop program to interface mouse driver.

Tools: TASM/MASM

Theory:

Interface mouse using Int 33H

Int 33h MS Mouse Interrupt

Function 0 Reset

Input

AX = 0

Output

AX = mouse status -1 = installed

0 = not installed

BX = number of buttons

Function 1 Show Mouse Cursor

Input AX = 1

Output

NONE

Function 2 Hide Mouse Cursor

Input AX = 2

Output NONE

Function 3 Get Mouse Position & Button

Status

Input AX = 3

Output

BX = Button Status

xxxx xxxx xxxx xMRL

M=middle (if present) R=right L=left

0= not pressed 1 = pressed

CX = Horizontal Mouse Cursor Position

DX = Vertical Mouse Cursor Position

(div positions by 2 for med res
graphics; div by 8 for text mode)

Function 4 Set Mouse Cursor Position

Input AX = 4

CX = new horizontal cursor position

DX = new vertical cursor position

Output NONE

Function 5 Get Button Press Information

Input AX = 5

BX = button of interest (0=L; 1=R; 2=M)

Output

AX = button status (current status of ALL
buttons)

BX = number of buutton presses on
specified button

CX = horizontal position at last press

DX = vertical position at last press

Function 6 Get Button Release

Information

Input AX = 6

BX = button of interest (0=L; 1=R; 2=M)

Output

AX = button status (current status of ALL
buttons)

BX = number of buutton presses on
specified button

CX = horizontal position at last release

DX = vertical position at last release

**Function 7 Set Minimum and Maximum
X Position**

Input AX = 7

CX = new minimum horizontal cursor
position

DX = new maximum horizontal cursor
position

Output NONE

**Function 8 Set Minimum and Maximum
Y Position**

Input AX = 8

CX = new minimum vertical cursor position

DX = new maximum vertical cursor position

Output NONE

Function 9 Define Graphics Cursor

Input

AX = 9

BX = horizontal cursor hot spot (0,0) upper
left

CX = vertical cursor hot spot

ES:DX = address of screen and cursor mask

Output NONE

Application: Use of Int 33 H to interface mouse with system.

Program

```

model small
.stack
.data
msg1 db 10,13,"Mouse driver present:$"
.code
disp macro xx
mov ah,09h
lea dx,xx
int 21h
endm
.startup
mov ax,0000 ;mouse driver check
int 33h
cmp ax,00h
je last
disp msg1

mov ax,0004 ;mouse cursor position
mov cx,0
mov dx,0
int 33h
mov ax,0007 ;set horizontal limit
mov cx,0010
mov dx,055h
int 33h
mov ax,0008 ;set vertical limit
mov cx,0010
mov dx,055h
int 33h

```

```

pixel:
mov ax,0001 ;display mouse cursor
int 33h
mov ax,0003
int 33h
mov ax,0003
int 33h

cmp bx,01 ;left button
je left
jmp right

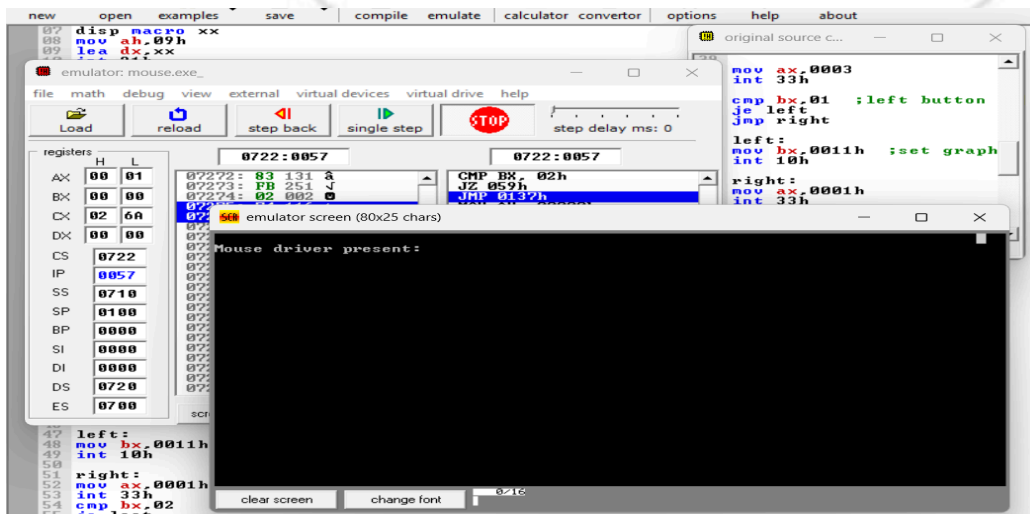
left:
mov bx,0011h ;set graphics mode
int 10h

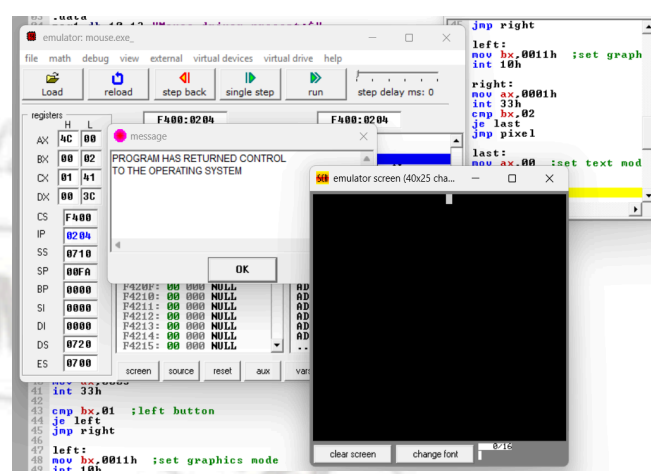
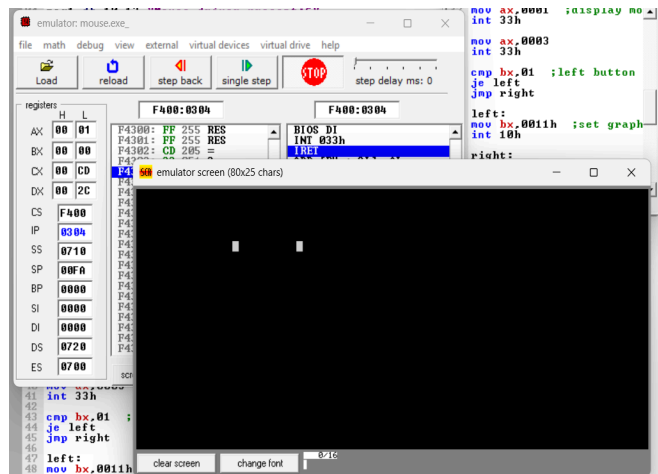
right:
mov ax,0001h
int 33h
cmp bx,02
je last
jmp pixel

last:
mov ax,00 ;set text mode
int 10h

.exit
end

```





Result and Discussion:

Learning Outcomes: The student should have the ability to

LO 9.1 Compare DOS and BIOS interrupts.

LO 9.2 Develop an application for Mouse interfacing using INT 33H.

LO 9.3 Develop an application for keyboard and Printer interfacing using INT 09H and INT 05H respectively.

Course Outcomes: Upon completion of the course students will be able to make use of instructions of 8086 to build assembly and Mixed language programs.

Conclusion:

For Faculty Use

Correction Parameters	Formative Assessment [40%]	Timely completion of Practical [40%]	Attendance / Learning Attitude [20%]	
Marks Obtained				