Paper 102: Programming & Problem solving through C

Lecture-32:Mouse programming

Mouse Programming: An Introduction

- Mouse is a simple input device which allows you to point and execute interface and is faster than keyboards many times.
- Mouse generally provide random interface while keyboards provide sequential interface.
- Mouse Programming is a topic which every C programmer from beginner to professional needs to have in his toolbox to have a cutting edge.

Mouse device driver

- Initially when a mouse is attached to the computer, the device driver is loaded.
- The device driver senses the signals coming from the port to which the mouse is attached
- On sensing the signals, the driver translates these into related action on the screen
- It is usually available as a program called MOUSE.COM or WITTYMS.COM, which work for variety of mice
- The mouse has a separate cursor, as we move the mouse, the mouse pointer/cursor moves correspondingly

Access mouse settings

- Once the driver is loaded the various functions can be accessed by
  - setting up the AX register with different values
  - and issuing interrupt number 33h

Mouse setting

Interrupt	Service	Purpose
51	0	**Reset mouse and get status** Call with AX = 0 Returns: AX = FFFFh If mouse support is available Ax = 0 If mouse support is not available
51	1	**Show mouse pointer** Call with AX = 1 Returns: Nothing
51	2	**Hide mouse pointer** Call with AX = 2 Returns: Nothing
51	3	**Get mouse position and button status** Call with AX = 3 Returns: BX = mouse button status Value Significance 0 button not pressed 1 left button is pressed 2 right button is pressed 3 center button is pressed CX = x coordinate DX = y coordinate

Mouse setting

51	4	Set mouse pointer position Call with AX = 4 CX = x coordinate DX = y coordinate Returns: Nothing
51	7	Set horizontal limits for pointer Call with AX = 7 CX = minimum x coordinate DX = maximum x coordinate Returns: Nothing
51	8	Set vertical limits for pointer Call with AX = 8 CX = minimum y coordinate DX = maximum y coordinate Returns: Nothing

Example

```
#include <dos.h>
#include <graphics.h>

union REGS in,out;
void main()
{               int gd=DETECT,gm,maxx,maxy,x,y,button;

                initgraph(&gd,&gm,"c:\\tc\\bgi");
                maxx=getmaxx();
                maxy=getmaxy();

                rectangle(0,56,maxx,maxy);
                gotoxy(26,2);
                outtext("Mouse Demonstration program");

                if(initmouse()==0)
                {
                                closegraph();
                                restorecrtmode();
                                outtext("Mouse driver not loaded");
                                exit(1);
                }
                restrictmsptr(1,57,maxx-1,maxy-1);
                showmousept();
                gotoxy(1,2);
                printf("Left Button");
                gotoxy(15,2);
                printf("Right Button");
                gotoxy(55,3);
                printf("Press any key to exit...");
```

Example

```
while(!kbhit())
        {
                getmspos(&button,&x,&y); /*get mouse position*/
                gotoxy(5,3);
                (button==1)?printf("DOWN"):printf("UP  ");
                gotoxy(20,3);
                (button==2)?printf("DOWN"):printf("UP  ");
                gotoxy(65,3);
                printf("x=%d y=%d",x,y);
        }
}

initmouse() /*initialize the mouse */
{
        in.x.ax=0;
        int86(0x33,&in,&out);
        return(out.x.ax);
}

showmousept() /*display the mouse pointer */
{
        in.x.ax=1;
        int86(0x33,&in,&out);
}
```

Example

```
restrictmsptr(int x1,int y1,int x2,int y2) /* restrict movement of mouse */
{
        in.x.ax=7; /* restrict horizontal limit */
        in.x.cx=x1;
        in.x.dx=x2;
        int86(0x33,&in,&out);
        in.x.ax=8; /* restrict vertical limit */
        in.x.cx=y1;
        in.x.dx=y2;
        int86(0x33,&in,&out);
        return;
}

getmspos(int *button,int *x,int *y) /* get mouse status */
{
        in.x.ax=3;
        int86(0x33,&in,&out);
        *button=out.x.bx;
        *x=out.x.cx;
        *y=out.x.dx;
        return;
}
```