led blinking

```
#include <p18f4550.h> //Include Controller specific .h
#include "vector_relocate.h" //Vector Remapping for USB HID Bootloader
//Declarations
#define lrbit PORTBbits.RB1 //SW0 interfaced to RB1
#define rlbit PORTBbits.RB0 //SW1 interfaced to RB0
#define buzzer PORTCbits.RC2 //Buzzer interfaced to RC2
#define relay PORTDbits.RD7 //Relay interfaced to RC1
//Function Prototypes
void msdelay (unsigned int time);//Function for delay
//Start of Program Code
void main() //Main Program
{
unsigned char i,val=0; //Variable to latch the switch condition
INTCON2bits.RBPU=0; //To Activate the internal pull on PORTB
ADCON1 = 0x0F; //To disable the all analog inputs
TRISBbits.RB0=1; //To configure RB0 as input for sensing SW1
TRISBbits.RB1=1; //To configure RB1 as input for sensing SW0
TRISDbits.TRISD7=0; //To configure RC1 (relay) as output
TRISCbits.TRISC2=0; //To configure RC2 (buzzer) as output
TRISA = 0x00; //To configure PORTD (LED) as output
PORTA = 0x00; //Initial Value for LED
buzzer = 0; //Initial Value for Buzzer
relay = 0; //Initial Value for Relay
while (1) //While loop for repeated operation
{
if (lrbit==0) //To check whether SW0 is pressed
```



```
val = 1; // Latch the status of switch SW0
if (rlbit==0) //To check whether SW1 is pressed
val = 2; // Latch the status of switch SW1
if (val == 1)
{
buzzer = 1;
relay = 1;
// 7led
// 0001 0000
PORTA = 0x20;
msdelay(50);
for(i=0;i<8;i++)
{
PORTA = PORTA >>1; //Shift left by 1 bit
msdelay(50); // Make the MSB bit equal to 1
}
}
if (val == 2)
{
buzzer = 0;
relay = 0;
```

```
PORTA = 0x01;
// 0000 0001
msdelay(50);
for(i=0;i<8;i++)
{
PORTA = PORTA <<1; //Shift left by 1 bit
msdelay(50); // Make the MSB bit equal to 1
}
}
}
}
//End of the Program
//Function Definitions
void msdelay (unsigned int time)//Function for delay
```



```
{
unsigned int i, j;
for (i = 0; i < time; i++)
for (j = 0; j < 710; j++); //Calibrated for a 1 ms delay in MPLAB
}
```