```
////////////////////////////////////////////////////////////////////////
#include <REGX51.H>
#define lcd P0
#define row P1
#define col P2
sbit rs=P1^5;
sbit rw=P1^6;
sbit en=P1^7;
sbit lcdbusy=P0^7;
sbit ready=P2^4;                        //Green Ready
sbit busy=P2^5;                         //Red Busy
sbit high=P2^7;                         //Green High
sbit low=P2^6;                          //Red Low
sbit check=P3^2;
////////////////////////////////// KEYPAD ///////////////////////////////////
unsigned char keypad[4][3]={ '1', '2',  '3',
                             '4', '5', '6',
                             '7', '8', '9',
                             '*', '0', '#'  };
////////////////////////////// LCD COMMANDS /////////////////////////////////
void lcdcmd(unsigned char);
void lcddata(unsigned char);
void lcdready(void);
void lcd_pre(void);
void lcd_guide(void);
void option(void);
void manually(void);
void osc(void);
void kpad(void);
////////////////////////////////// DELAY ///////////////////////////////////
void delay(unsigned int);
////////////////////////////// VARIABLES ///////////////////////////////////
unsigned char a;                        // Looping
unsigned int b;                         // Looping
unsigned char x;                        // Storing Character
unsigned char p;                        // Storing LCD Current Data #
unsigned char pulses;                   // Storing Pulses
unsigned char d1,d2,d3,dec;
////////////////////////////// MAIN FUNCTION ///////////////////////////////
void on_pulse(void) interrupt 0
{
    pulses++;
    dec=pulses/10;
    d1=pulses%10;
    d2=dec%10;
    d3=dec/10;
    d1=d1|0x30;
    d2=d2|0x30;
    d3=d3|0x30;

    lcdcmd(0x8A);
    lcddata(d3);
    lcddata(d2);
    lcddata(d1);

    while(check==0)
    {
        high=0;
        low=1;
    }
    high=1;
    low=0;
}
////////////////////////////////////////////////////////////
void main(void)
{
    P1=0x00;
    P2=0x00;
```

```
    ////////////////////////////
    ready=0;
    busy=1;
    high=0;
    low=0;
    ////////////////////////////
    pulses=0;
    ////////////////////////////
    IE=0x81;
    lcdcmd(0x38);
    lcdcmd(0x0C);
    lcdcmd(0x01);
    lcdcmd(0x06);
    lcdcmd(0x82);
    lcd_pre();
    lcd_guide();
    option();
    kpad();
    while(1);
}
////////////////////////////////////////////////////////////////////////
void lcd_pre(void)
{
    code unsigned char pre_data[]="-- Welcome --UET FSD MECHA-09Abdullah TahirHabib Sul
tan";
    for (a=0;a<=12;a++)
    {
        lcddata(pre_data[a]);                   // --Welcome --
    }
    delay(400);
    lcdcmd(0xC0);
    for (a=13;a<=28;a++)
    {
        lcddata(pre_data[a]);                   // UET FSD MECHA-09
    }

    delay(1500);                               ////////////////////////////////////
    lcdcmd(0x01);
    lcdcmd(0x81);
    for (a=29;a<=42;a++)
    {
        lcddata(pre_data[a]);                   // Abdullah Tahir
    }
    lcdcmd(0xC2);
    for (a=43;a<=54;a++)
    {
        lcddata(pre_data[a]);                   // M Habib Sultan
    }
    delay(1500);                               ////////////////////////////////////
    return;
}
////////////////////////////////////////////////////////////////////////
void lcd_guide(void)
{
    code unsigned char guide_data[]="-- Guide --Project can be ~used for counti-ng obje
cts,people, eventsetc.";
    lcdcmd(0x01);
    lcdcmd(0x82);
    for(a=0;a<=10;a++)
    {
        lcddata(guide_data[a]);                 // -- Guide --
    }
    lcdcmd(0xC0);
    for(a=11;a<=26;a++)
    {
        lcddata(guide_data[a]);                 // Project can be ~
    }
////////////////////////////////////////////////////////////////////////////////
```

```
    p=1;
    kpad();
////////////////////////////////////////////////////////////////////////////////
    lcdcmd(0x01);
    lcdcmd(0x80);
    for(a=27;a<=42;a++)
    {
        lcddata(guide_data[a]);                 // used for counti-
    }
    lcdcmd(0xC0);
    for(a=43;a<=53;a++)
    {
        lcddata(guide_data[a]);                 // ng objects,
    }
    lcdcmd(0xCF);
    lcddata('~');
////////////////////////////////////////////////////////////////////////////////
    p=1;
    kpad();
////////////////////////////////////////////////////////////////////////////////
    lcdcmd(0x01);
    lcdcmd(0x80);
    for(a=54;a<=67;a++)                     // people, events
    {
        lcddata(guide_data[a]);
    }
    lcdcmd(0xC0);
    for(a=68;a<=71;a++)
    {
        lcddata(guide_data[a]);                 // etc.
    }
    lcdcmd(0xcf);
    lcddata('~');
////////////////////////////////////////////////////////////////////////////////
    p=1;
    kpad();
////////////////////////////////////////////////////////////////////////////////

    return;
}
////////////////////////////////////////////////////////////////////////
void option(void)
{
    code unsigned char option_data[]="1: Manually2: OSC/SEN";
    pulses=0;
    p=2;
    high=0;
    low=0;
    ready=0;
    busy=1;
    lcdcmd(0x01);
    lcdcmd(0x80);
    lcdcmd(0x01);
    lcdcmd(0x80);
    for(a=0;a<=10;a++)
    {
        lcddata(option_data[a]);                    // # of pulses
    }
    lcdcmd(0xC0);
    for(a=11;a<=20;a++)
    {
        lcddata(option_data[a]);                    // 1 for Motor 2 for RESET
    }
////////////////////////////////////////////////////////////////////////////////
    kpad();
////////////////////////////////////////////////////////////////////////////////
    return;
```

```
}
////////////////////////////////////////////////////////////////////////
void osc(void)
{
    code unsigned char osc_data[]="Pulses=1:Manually 2:RST";
    p=4;

    high=0;
    low=0;

    lcdcmd(0x01);
    lcdcmd(0x80);
    for(a=0;a<=6;a++)
    {
        lcddata(osc_data[a]);                   // r/s= r/m=
    }
    lcdcmd(0xC0);
    for(a=7;a<=22;a++)
    {
        lcddata(osc_data[a]);                   // 2:RESET
    }
////////////////////////////////////////////////////////////////////////////////
    ready=1;
    high=1;
    busy=0;
    kpad();
////////////////////////////////////////////////////////////////////////////////
    return;
}
////////////////////////////////////////////////////////////////////////
void manually(void)
{
    code unsigned char osc_data[]="Pulses =1:OSC/SEN 2:RST";
    p=3;

    high=0;
    low=0;

    lcdcmd(0x01);
    lcdcmd(0x80);
    for(a=0;a<=7;a++)
    {
        lcddata(osc_data[a]);                   // r/s= r/m=
    }
    lcdcmd(0xC0);
    for(a=8;a<=22;a++)
    {
        lcddata(osc_data[a]);                   // 2:RESET
    }
////////////////////////////////////////////////////////////////////////////////
    ready=1;
    high=1;
    busy=0;

    kpad();
////////////////////////////////////////////////////////////////////////////////
    return;
}
////////////////////////////////////////////////////////////////////////
void lcdcmd(unsigned char value)
{
    lcdready();
    lcd=value;
    rs=0;
    rw=0;
    en=1;
    delay(50);
    en=0;
```

```
    return;
}
////////////////////////////////////////////////////////////////////////
void lcddata(unsigned char value)
{
    lcdready();
    lcd=value;
    rs=1;
    rw=0;
    en=1;
    delay(50);
    en=0;
    return;
}
////////////////////////////////////////////////////////////////////////
void lcdready(void)
{
    lcdbusy=1;
    rs=0;
    rw=1;
    while(lcdbusy==1)
    {
        en=0;
        delay(50);
        en=1;
    }
    return;
}
////////////////////////////////////////////////////////////////////////
void kpad(void)
{
    unsigned char c,r;
    col=0xff;

    if(p==1)
    {
        high=0;
        low=0;
        ready=0;
        busy=1;
    }

    if(p==2)
    {
        high=0;
        low=0;
        ready=0;
        busy=1;
    }

    if(p==3 || p==4)
    {
        ready=1;
        high=1;
        busy=0;
        low=0;
    }



    while(1)
    {
        do
        {
            row=0x00;
            c=col;
            c&=0x0f;
        }
```

```
        while(c!=0x0f);

        do
        {
            do
            {
                delay(20);
                c=col;
                c&=0x0f;
            }
            while(c==0x0f);

            delay(20);
            c=col;
            c&=0x0f;
        }
        while(c==0x0f);

        while(1)
        {
            row=0xfe;
            c=col;
            c&=0x0f;
            if(c!=0x0f)
            {
                r=0;
                break;
            }

            row=0xfd;
            c=col;
            c&=0x0f;
            if(c!=0x0f)
            {
                r=1;
                break;
            }

            row=0xfb;
            c=col;
            c&=0x0f;
            if(c!=0x0f)
            {
                r=2;
                break;
            }

            row=0xf7;
            c=col;
            c&=0x0f;
            r=3;
            break;

        }

        if(c==0x0e)
        {
            x=keypad[r][0];
        }
        else if(c==0x0d)
        {
            x=keypad[r][1];
        }
        else if(c==0x0b)
        {
            x=keypad[r][2];
        }
        else
```

```
        {
            x=keypad[r][3];
        }
        ////////////////////////////////////////////////////////////////
        if(p==1 && x=='#')
        {
                return;
        }
        if(p==2)
        {
            if(x=='1')
            {
                manually();
                return;
            }
            if(x=='2')
            {
                osc();
                return;
            }
        }
        if(p==3 || p==4)
        {
            if(x=='2')
            {
                option();
                return;
            }
        }
        if(p==3 && x=='1')
        {
            osc();
            return;
        }
        if(p==4 && x=='1')
        {
            manually();
            return;
        }
    }
}
////////////////////////////////////////////////////////////////////////
void delay(unsigned int z)
{
    for(b=1;b<z;b++)
    {
        TMOD=0X01;
        TH0=0XFC;
        TL0=0X66;
        TR0=1;
        while(TF0==0);
        TR0=0;
        TF0=0;
    }
    return;

}
////////////////////////////////////////////////////////////////////////
```