



```

#include "reg52.h"
#include <math.h>

#define C_SET    1
#define C_RESET 0

sbit SCL = P3^2;
sbit SDA = P3^3;

unsigned char set_position;

unsigned int  XRaw,YRaw;

unsigned int  Xsens, Ysens;
unsigned int  Xoffset, Yoffset;
unsigned int  angle ;

unsigned char code cAtanStep[19]={0x11, 0x12, 0x13, 0x14, 0x15, 0x16, 0x17,
0x18, 0x19,0x1a,0x1b,0x1c,0x1d,0x1e,0x1f,0x20,0x21,0x22,0x23};
int          code          iAtanPoint[19]=
{0x020C,0x0998,0x0d60,0x1057,0x12cd,0x1508,0x1702,0x18ca,0x1a8c,0x1c2c,0x1da6,0x
1f12,0x206e,0x21b8,0x2258,0x244d,0x2576,0x26a8,0x2710};
int          code          iAtanAngle[19]=
{0x001e,0x008a,0x00bd,0x00e3,0x0101,0x011b,0x0131,0x0144,0x0156,0x0166,0x0174,0x

```

```
0181,0x018d,0x0198,0x01a3,0x01ad,0x01b6,0x01bf,0x01c2};
```

```
void SysDelay1ms( unsigned int t)
{
    unsigned int i;
    while(t--)
        {for (i = 0; i < 1250; i++);
         for (i = 0; i < 1000; i++);
         ScanKey();
        }
}
```

```
//=====
```

```
void SysDelay(unsigned int t)
{
    while(t--);
}
```

```
//=====
```

```
void IIC_SDA_HIGH(void)
{
    SDA=1;
}
```

```
//=====
```

```
void IIC_SDA_LOW(void)
{
    SDA=0;
}
```

```
//=====
```

```
void IIC_SCL_HIGH(void)
{
    SCL=1;
}
```

```
//=====
```

```
void IIC_SCL_LOW(void)
```

```

{
    SCL=0;
}

//=====================================================

void IIC_XCLR_LOW(void)
{
    XCLR=0;
}

//=====================================================

void IIC_XCLR_HIGH(void)
{
    XCLR=1;
}

void MCLKOff(void)
{
    TR0 = 0;
    ET0 = 0;
}

//=====================================================

void IIC_Start(void)
{
    IIC_SDA_HIGH();
    SysDelay(DELAY10US);

    IIC_SCL_HIGH();
    SysDelay(DELAY10US);

    IIC_SDA_LOW();
    SysDelay(DELAY10US);

    IIC_SCL_LOW();
    SysDelay(DELAY10US);
}

//=====================================================

void IIC_Stop(void)

```

```

{
    IIC_SCL_LOW();
    SysDelay(DELAY10US);

    IIC_SDA_LOW();
    SysDelay(DELAY10US);

    IIC_SCL_HIGH();
    SysDelay(DELAY10US);

    IIC_SDA_HIGH();
    SysDelay(DELAY10US);
}
//=====

void IIC_ACK(void)
{
    IIC_SDA_LOW();
    SysDelay(DELAY10US);

    IIC_SCL_HIGH();
    SysDelay(DELAY10US);

    IIC_SCL_LOW();
    SysDelay(DELAY10US);
}
//=====

void IIC_NoAck(void)
{
    IIC_SDA_HIGH();
    SysDelay(DELAY10US);

    IIC_SCL_HIGH();
    SysDelay(DELAY10US);

    IIC_SCL_LOW();
    SysDelay(DELAY10US);
}

```

```
//=====
```

```
unsigned char IIC_ReadByte(void)
{
    unsigned char ucValue;
    unsigned char ucIndex;

    IIC_SDA_HIGH();
    SysDelay(DELAY10US);
    for ( ucIndex = 0; ucIndex < 8; ucIndex++ )
    {
        ucValue <<= 1;

        IIC_SCL_LOW();
        SysDelay(DELAY10US);

        IIC_SCL_HIGH();
        SysDelay(DELAY10US);

        if(SDA)
            ucValue |= 1;

        SysDelay(DELAY10US);
        IIC_SCL_LOW();
        SysDelay(DELAY10US);
    }
    return ucValue;
}
```

```
void IIC_WriteByte( unsigned char ucData )
{
    unsigned char i;
    for( i = 0; i < 8; i++ )
    {
        IIC_SCL_LOW();
        SysDelay(DELAY10US);

        if((ucData & 0x80) == 0x80)
        {
            IIC_SDA_HIGH();
            SysDelay(DELAY10US);
        }
    }
}
```

```

        else
        {
            IIC_SDA_LOW();
            SysDelay(DELAY10US);
        }

        IIC_SCL_HIGH();
        SysDelay(DELAY10US);
        ucData <<= 1;
        IIC_SCL_LOW();
    }

    IIC_SDA_HIGH();
    SysDelay(DELAY10US);
    IIC_SCL_LOW();
    SysDelay(DELAY10US);
    IIC_SCL_HIGH();
    SysDelay(DELAY10US);
    IIC_SCL_LOW();
    SysDelay(DELAY10US);
}

```

```

void cal_compass(void)
{
    unsigned int X_raw, Y_raw;
    unsigned long int IAcc;

    if( XRaw > Xoffset )
    {
        X_raw = XRaw - Xoffset;
    }
    else
    {
        X_raw = Xoffset - XRaw ;
    }
    if( YRaw > Yoffset)
    {
        Y_raw = YRaw - Yoffset;
    }
    else
    {

```

```

    Y_raw = Yoffset - YRaw ;
}

if( Xsens > Ysens )
    Y_raw = ( Y_raw * Xsens ) / Ysens;
else
    X_raw = ( X_raw * Ysens ) / Xsens;

if( X_raw >= Y_raw)
{
    IAcc = Y_raw;

    IAcc *= 1000;
    IAcc /= X_raw;

    IAcc /= 1;

    IAcc = arcTan(IAcc);
    angle = IAcc / 10;

else
}

{
    IAcc = X_raw;
    IAcc *= 1000;
    IAcc = IAcc/Y_raw;
    IAcc = arcTan(IAcc);
    angle = 90 - (IAcc/10);
}

if((XRaw < Xoffset) && (YRaw > Yoffset))    angle = 180 - angle;

else if((XRaw < Xoffset) && (YRaw < Yoffset) )    angle = 180 + angle;
else if((XRaw > Xoffset) && (YRaw < Yoffset) )angle = 360 - angle;

if(angle>=360) angle -=360;
else if(angle<0) angle +=360;
}

unsigned long int arcTan(unsigned long int IAcc)
{
    char I;

```

```

IAcc *= 10;

if (IAcc>10000)
    IAcc=10000;

for (l=0x00;l<0x19;l++)
{
    if (IAcc<=iAtanPoint[l])
    {
        if (l== 0)
        {
            IAcc/=cAtanStep[0];
        }
        else
        {
            IAcc/=iAtanPoint[l-1];
            IAcc/=cAtanStep[l-1];
            IAcc+=iAtanAngle[l-1];
        }
        break;
    }
}
return(IAcc);
}

void get_compass_data(void)
{
    unsigned char i;
    unsigned int x_value, y_value, x_max, y_max,x_min, y_min ;
    x_max = y_max = x_value = y_value = 0;
    x_min = y_min = 4095;

    for(i = 0; i<6; i++)
    {
        get_compass_data1();

        x_value += XRaw;
        y_value += YRaw;
        if(XRaw > x_max ) x_max = XRaw;
        if(YRaw > y_max ) y_max = YRaw;
        if(XRaw < x_min ) x_min = XRaw;
        if(YRaw < y_min ) y_min = YRaw;
    }
}

```



```

    x_value -= x_max;
    x_value -= x_min;
    y_value -= y_max;
    y_value -= y_min;

    XRaw = x_value/4;
    YRaw = y_value/4;
}

void get_compass_data1(void)
{
    unsigned char ucValue;
    XCLR = 0;

    compass_set_reset();    // reset
    compass_set_reset();    // set

    IIC_Stop();
    IIC_Start();

    IIC_WriteByte(0x60);
    IIC_WriteByte(00);
    IIC_WriteByte(01);
    IIC_Stop();
    SysDelay1ms(10);        // delay5~~10ms
    IIC_Start();
    IIC_WriteByte(0x60);
    IIC_WriteByte(0x01);
    IIC_Start();

    IIC_WriteByte(0x61);

    ucValue = IIC_ReadByte();
    IIC_ACK();
    XRaw=ucValue;
    ucValue = IIC_ReadByte();
    IIC_ACK();
    XRaw <<= 8;
    XRaw |= ucValue;

    ucValue = IIC_ReadByte();
    IIC_ACK();
    YRaw=ucValue;

```

```

        ucValue = IIC_ReadByte();
        IIC_NoAck();
        IIC_Stop();
        YRaw <<= 8;
        YRaw |= ucValue;

        XRaw &= 0x0fff;
        YRaw &= 0x0fff;
    }

void compass_set(void)
{
    XCLR = 0;

    IIC_Start();
    IIC_WriteByte(0x60);
    IIC_WriteByte(00);
    IIC_WriteByte(02);
    IIC_Stop();
    SysDelay1ms(1);
}

void compass_reset(void)
{
    XCLR = 0;

    IIC_Start();
    IIC_WriteByte(0x60);
    IIC_WriteByte(00);
    IIC_WriteByte(04);
    IIC_Stop();
    SysDelay1ms(1);
}

void compass_set_reset(void)
{
    if(set_position == C_SET)
    {
        compass_set();
        set_position = C_RESET;
    }
    else
    {
        compass_reset();
        set_position = C_SET;
    }
}

```

```

    }
    SysDelay1ms(1);
}

void compass_reset_calibration(void)
{
    unsigned int i;
    unsigned int Xmax = 0,
                Xmin = 4095,
                Ymax = 0,
                Ymin = 4095;

    compass_set_reset();
    compass_set_reset();
    compass_set_reset();

    for(i=0;i<600;i++)
    {
        get_compass_data1();

        if( XRaw > Xmax )
            Xmax = XRaw;
        if( XRaw < Xmin )
            Xmin = XRaw;
        if( YRaw > Ymax )
            Ymax = YRaw;
        if( YRaw < Ymin )
            Ymin = YRaw;
    }
    Xoffset = (Xmax + Xmin) / 2;
    Yoffset = (Ymax + Ymin) / 2;

    Xsens = (Xmax - Xmin) / 2;
    Ysens = (Ymax - Ymin) / 2;
}

void main(void)
{
    SysDelay1ms(100);

    set_position = C_SET;
    compass_reset_calibration();
}

```

```
KeyValue = 0;
while(1)
{
    get_compass_data();
    cal_compass();
}
}
```