#include <xc.h>

#include <stdio.h>

#include "lcd.h"

#include "security.h"

//LCD Functions Developed by electroSome

#define \_XTAL\_FREQ 8000000

#define RS RD2

#define EN RD3

#define D4 RD4

#define D5 RD5

#define D6 RD6

#define D7 RD7

void Lcd\_Port(char a)

{

if(a & 1)

RD4 = 1;

else

RD4 = 0;

if(a & 2)

RD5 = 1;

else

RD5 = 0;

if(a & 4)

RD6 = 1;

else

RD6 = 0;

if(a & 8)

RD7 = 1;

else

RD7 = 0;

}

void Lcd\_Cmd(char a)

{

RS = 0; // => RS = 0

Lcd\_Port(a);

EN = 1; // => E = 1

\_\_delay\_ms(4);

EN = 0; // => E = 0

}

Lcd\_Clear()

{

Lcd\_Cmd(0);

Lcd\_Cmd(1);

}

void Lcd\_Set\_Cursor(char a, char b)

{

char temp,z,y;

if(a == 1)

{

temp = 0x80 + b - 1;

z = temp>>4;

y = temp & 0x0F;

Lcd\_Cmd(z);

Lcd\_Cmd(y);

}

else if(a == 2)

{

temp = 0xC0 + b - 1;

z = temp>>4;

y = temp & 0x0F;

Lcd\_Cmd(z);

Lcd\_Cmd(y);

}

}

void Lcd\_Init()

{

Lcd\_Port(0x00);

\_\_delay\_ms(20);

Lcd\_Cmd(0x03);

\_\_delay\_ms(5);

Lcd\_Cmd(0x03);

\_\_delay\_ms(11);

Lcd\_Cmd(0x03);

/////////////////////////////////////////////////////

Lcd\_Cmd(0x02);

Lcd\_Cmd(0x02);

Lcd\_Cmd(0x08);

Lcd\_Cmd(0x00);

Lcd\_Cmd(0x0C);

Lcd\_Cmd(0x00);

Lcd\_Cmd(0x06);

}

void Lcd\_Write\_Char(char a)

{

char temp,y;

temp = a&0x0F;

y = a&0xF0;

RS = 1; // => RS = 1

Lcd\_Port(y>>4); //Data transfer

EN = 1;

\_\_delay\_us(40);

EN = 0;

Lcd\_Port(temp);

EN = 1;

\_\_delay\_us(40);

EN = 0;

}

void Lcd\_Write\_String(char \*a)

{

int i;

for(i=0;a[i]!='\0';i++)

Lcd\_Write\_Char(a[i]);

}

void Lcd\_Shift\_Right()

{

Lcd\_Cmd(0x01);

Lcd\_Cmd(0x0C);

}

void Lcd\_Shift\_Left()

{

Lcd\_Cmd(0x01);

Lcd\_Cmd(0x08);

}

void ADC\_Init()

{

ADCON0 = 0x81;

ADCON1 = 0x00;

}

unsigned int ADC\_Read(unsigned char x)

{

unsigned char channel =x;

ADCON0 &= 0xC5; //Clearing channel selection bits

ADCON0 |= channel<<3; //Setting channel selection bits

\_\_delay\_ms(2); //Acquisition time to charge hold capacitor

GO\_nDONE = 1; //Initializes A/D conversion

while(GO\_nDONE); //Waiting for conversion to complete

return (ADRESH); //Return result

}

void main(void)

{

OSCCON=0x76; //Configure to use 8MHz internal oscillator.

float a,b,at,bt,check=0,state=0;

char s[20];

TRISD=0x01;

TRISB0=0;

ADC\_Init();

Lcd\_Init();

Lcd\_Clear();

bt=35;

while(1)

{

a=ADC\_Read(0);

b=ADC\_Read(1)\*1.85;

char s[20];

Lcd\_Set\_Cursor(2,1);

if(RD0==0)

{

Lcd\_Write\_String(s);

sprintf(s, " GAS LEAK");

check=1;

}

else if(b>bt)

{

Lcd\_Write\_String(s);

sprintf(s, " HIGH TEMPERATURE");

check=1;

}

else if(a>100 && a<185)

{

Lcd\_Write\_String(s);

sprintf(s, " LOW LIGHT");

check=1;

}

else

{

check=0;

Lcd\_Write\_String(s);

sprintf(s, " Welcome");

}

Lcd\_Set\_Cursor(1,1);

Lcd\_Write\_String(s);

sprintf(s, " Temp=%.02fC",b );

if(check==1)

{

int i=0;

for(i=0;i<1000;i++)

{

RB0=1;

\_\_delay\_us(200);

RB0=0;

\_\_delay\_us(100);

}

}

Lcd\_Clear();

}

}