

```

//Bibliotecas necessárias
#include <pic.h>
#include <string.h>
#include <stdlib.h>
#include "delay.c"
#include "DS1wire.c"
#include "lcd.c"

__CONFIG(0x3F70);

#define INPUT 1
#define OUTPUT 0
#define TRUE 1
#define FALSE 0

void startup();
char p[7];
volatile unsigned char TM, TL;
unsigned char numROMs;
unsigned char FoundROM[MAX_SENSORS][8]; // Tabela com o numero máximo de ROMs

//Função main
void main(void)
{
    //INTERRUPÇÃO
    INTE = 1; // habilita interrupção externa para o pino 6 (RB0)
    INTEDG = 1; //para subida de pulso
    //FIM INTERRUPÇÃO
    //Variáveis
    unsigned char x=0,y=0;
    unsigned int temp_i;

    //Configuração de portas
    GIE = 0;
    PEIE = 0;
    TRISB = 0b00000001;
    CMCON = 0b00000111;
    TRISA = 0b00000011;
    PORTA = 0x00;

    //Inicialização do Display
    lcd_init();
    startup();

    //Instruções para inicializar rede 1-Wire
    D_Reset();
    FindDevices();

    while(1) //Repetição inf
    {
        //Instruções para mudança de display a ser mostrado
        if (INTF==1) //Interrupção do pino 6 do pic(FLAG)
        {
            INTF=0; //Zera o registrador que é acionado quando o botão é
pressionado
            x++; //Muda o sensor a ser mostrado
            if(x==numROMs) //Número mínimo de ROMs encontradas, mostra o
sensor inicial
            {
                x=0;
            }

            //Instruções para leitura de temperatura
            read_one(x);
            lcd_clear();
            lcd_goto(1);
            lcd_puts("UCPel-Sensor:");

```

```

        long_to_string((x+1), p, 2);
        lcd_puts(p);
        TM = TM << 4;
        y = (TL >> 4) & 0b00001111;
        TM = (TM & 0b11110000) | y;
        TL = TL & 0b00001111;
        lcd_goto(0x40);
        //Avalia se a temperatura é positiva ou negativa
        //TM valor inteiro de temperatura
        //TL valor fracionado de temperatura
        if(TM & 0b10000000)
        {
            lcd_puts("Temp:-");
            TM=(TM^0xFF);
            TL=((TL^0xFF)+1) & 0b00001111;
        }
        else
        {
            lcd_puts("Temp:");
        }
        long_to_string(TM, p, 3); //Define valor inteiro de temperatura a ser
        representado no mínimo por 3 casas
        lcd_puts(p);
        lcd_puts(".");
        temp_i=(int)TL;
        temp_i=temp_i* 625;
        long_to_string_lz(temp_i, p, 3); //Define o número de casas decimais
        para mostrar a parte fracionária de temperatura
        lcd_puts(p);
        lcd_puts(" ");
        lcd_putchar(0b11011111); // Caractere graus
        lcd_puts("C");

        //Teste da temperatura
        if (TM >= 27 || TM <= 24) //Estabelece a temperatura Max e Min para
        acionamento do LED de alarme
        {
            RA2 = 1 ; //Aciona LED.
        }
        else
        {
            RA2 = 0 ;//Desliga LED
        }
    }
}

//Função de inicialização de leitura
void startup()
{
    lcd_clear();
    lcd_puts(" DS18B20 Sensor ");
    lcd_goto(0x40);
    lcd_puts(" Sistemas Aut. ");
    DelayMs(1300);
}

```