

/* =====

Sketch universal para el control de termistores.

Autor: David Losada, basado en trabajos de Rafael Torrales y Antonio Moles.

Version: 0.1

Fecha: 2 de julio de 2012

Descripcion: termometro con termistor NTC y pantalla LCD grafica Nokia 5110
de 84x48 pixeles

===== */

```
#include <Adafruit_GFX.h> //Necesarias para usar la pantalla LCD Nokia
```

```
#include <Adafruit_PCD8544.h>
```

```
// pin 3 - Serial clock out (SCLK)
```

```
// pin 4 - Serial data out (DIN)
```

```
// pin 5 - Data/Command select (D/C)
```

```
// pin 7 - LCD chip select (CS)
```

```
// pin 6 - LCD reset (RST)
```

```
Adafruit_PCD8544 display = Adafruit_PCD8544(7, 6, 5, 4, 3);
```

```
//Control temperatura
```

```
//Para ahorrar cálculos, lo vamos a calcular antes del programa
```

```
const float TPto1 = -6; //Temperatura en °C punto 1
```

```
const float RPto1 = 40000; // Ohms en punto 1
```

```
const float TPto2 = 96; // Temp. punto 2
```

```
const float RPto2 = 970; // Ohms en punto 2
```

```
const float resistor = 10290; //El valor en ohmios de la resistencia del termistor a 25°C
```

```
const float voltage = 4.83; // El voltaje real en el punto 5Vcc de tu placa Arduino
```

```
//Para ahorrar cálculos lo definimos como constante en esta parte del programa
```

```
const float K= 273.15; //Para pasar a grados Kelvin
```

```
const float e = 2.718281828459045; //Constante matemática
```

```
const float B = log(RPto2/RPto1)/(1/(TPto2+K)-(1/(TPto1+K))); //Valor Beta de tu termistor
```

```
const float unodivr = 1/(resistor * pow(e,(-B/298.15))); //Con pow elevamos e al resultado
```

```
float T = 0; //Declaramos la variable Temperatura
```

```
float tmedia; //Variable para calcular media
```

```
int cnt; //temperatura media y contador
```

```
int grados, decimas; //Para ponerle coma al resultado (en español)
```

```
int sensorPin = A0; // Selecciona la entrada para el termistor
int sensorValue = 0; // Aquí almacenamos el valor del sensor
```

```
void setup() {
  // Iniciamos comunicacion serie con pantalla
  Serial.begin(9600);

  // Iniciamos comunicacion con la pantalla
  display.begin();

  // Establecemos el contraste
  display.setContrast(45);

  // Mostramos la pantalla de inicio durante 2 segundos
  display.display();
  delay(2000);
  // borramos la pantalla
  display.clearDisplay();
}
```

```
void loop()
{
  //float B= (RPT02/RPT01);//(1/(TPto2+K)-(1/(TPto1+K)));
  // Parte 1: Leemos el puerto analógico 0 y convertimos el valor en voltios.
  sensorValue = analogRead(sensorPin); //Leemos analógico 0
```

```
  tmedia=0; //Inicializamos variable
  // medimos la temperatura 10 veces y la almacenamos en tmedia
  for ( cnt=0; cnt<10; cnt++)
  {
    sensorValue = analogRead(sensorPin);
    tmedia=tmedia+sensorValue;
    delay(25);
  }
```

```
  // calculamos la media de las medidas
  tmedia=tmedia/cnt;
  //Convertimos a voltios :)
  float v2 = (voltage*float(tmedia))/1024.0f;
```

```
  // Parte 2: Calcular la resistencia con el valor de los voltios mediante la ecuación del divisor de
  voltaje
```

```
//voltage = 4.83
//R2 = 10000
//R1 = Thermistor resistance
//V2= v2
//so V2=(R2*v)/(R1+R2)
//and r1=((r2*v)/v2)-r2 <--final
```

```
float r1a = (voltage*float(resistor))/v2;
float r1 =r1a - resistor;
```

//Parte 3: Calcular la temperatura basandose en la ecuación Steinhart-Hart y la ecuación del valor Beta.

```
// T=B/ln(r1/rinfinite)
```

```
float T = B/log(r1*unodiv);
T=T-273.15; //Convertimos a °C y ya tenemos la temperatura
```

```
// ahora obtenemos la parte entera
grados=(int)T;
```

```
// y las decimas de grado son el resto
decimas=(T-grados)*10;
```

```
// Y por ultimo lo mandamos a la pantalla LCD
display.setTextSize(1);
display.println("Temperatura:");
display.print(" ");
display.setTextSize(2);
display.print(grados);
display.print(",");
display.print(decimas);
display.print(".C");
//display.print(" ");
display.display();
display.clearDisplay();
```

```
}
```