



# PROYECTO N° 3

## Termómetro de colores

## PROYECTO N° 3: Termómetro de colores

Aprende a construir y programar un termómetro de colores utilizando los componentes del Maker Control Kit y el Maker Kit 2.

Utiliza el sensor de temperatura y el LED RGB para montarlo. Observa cómo el LED cambia de color de acuerdo a la temperatura ambiental, según los colores que muestra el termómetro.

**NIVEL DE DIFICULTAD:** Principiante.

**DURACIÓN DEL EJERCICIO:** 45 min.

## **MATERIALES:**

- 1 LED RGB
- 1 Sensor de temperatura
- 1 Placa controladora Build&Code 4in1
- 1 Cable USB - Micro USB
- Ordenador
- Plantilla del termómetro
- Adhesivo

## **¿Qué es un sensor de temperatura?**

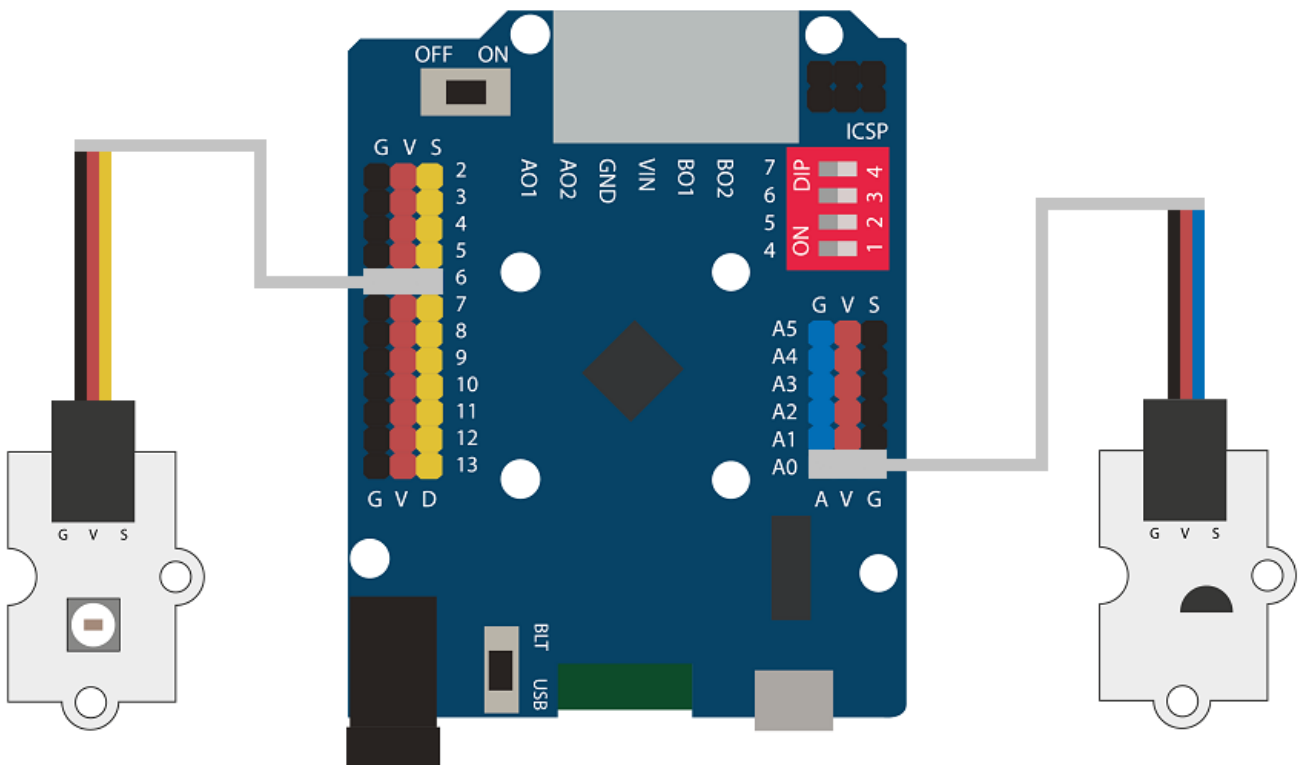
Un sensor de temperatura es un componente electrónico que devuelve un valor de tensión proporcional a la temperatura a la que está sometido.

En este caso, el sensor devuelve una señal analógica, con un rango de temperatura de  $-40^{\circ}\text{C}$  a  $+125^{\circ}\text{C}$  y una diferencia de  $\pm 2^{\circ}\text{C}$ .

## **CONEXIONES:**

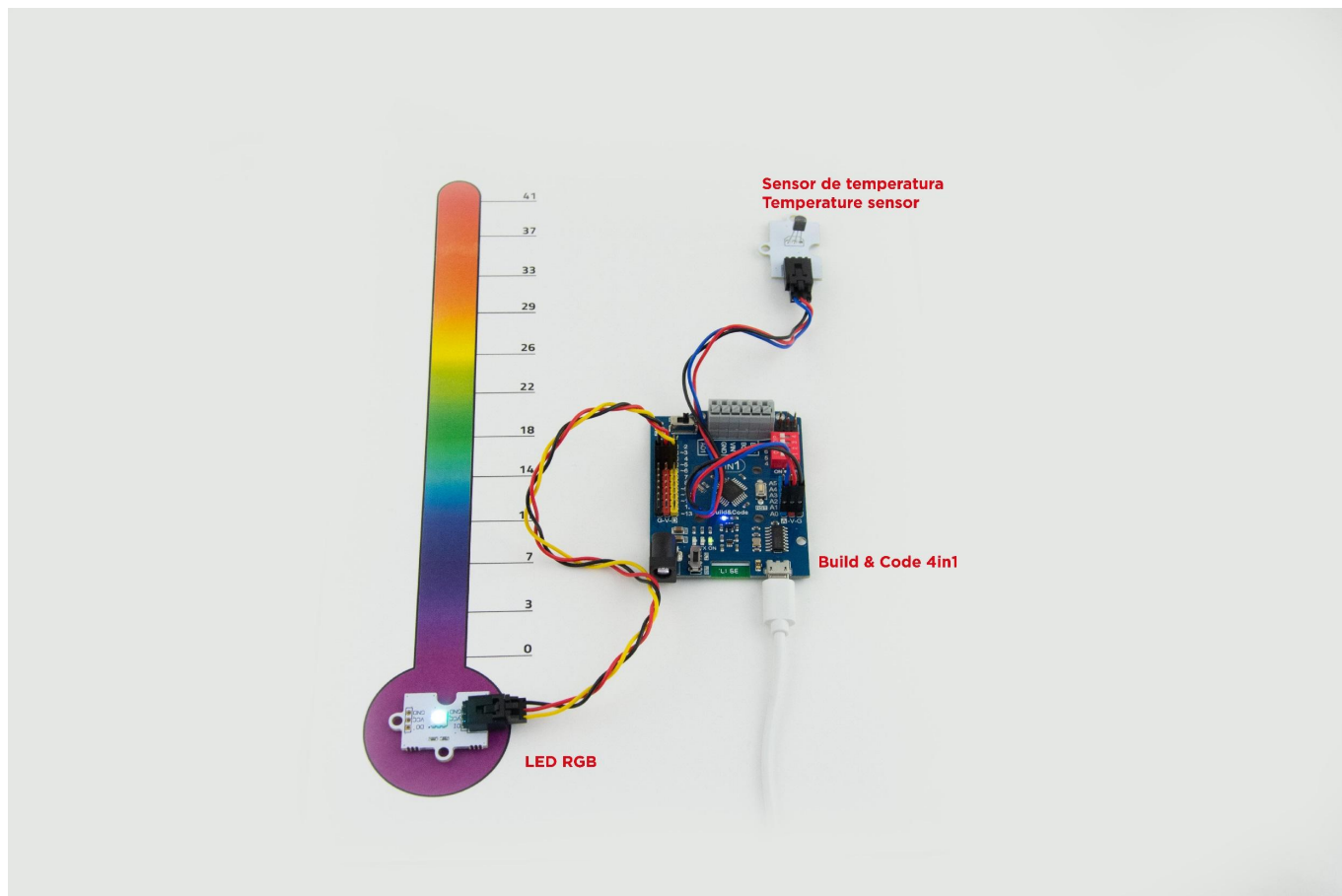
1. Conecta el sensor de temperatura al puerto analógico A0 de la placa controladora Build&Code 4in1.
2. Conecta el LED RGB al puerto digital 6 de la placa controladora Build&Code 4in1.

Para guiarte, mira los colores de los cables y los colores de los terminales de la placa controladora Build&Code 4in1. Cada cable debe ir conectado a su color:



**CONSTRUCCIÓN DE LA ESTRUCTURA:**

[Descarga la plantilla del termómetro](#) y pega los distintos componentes electrónicos. Usa esta imagen como referencia:



## CÓDIGO DE PROGRAMACIÓN:

El programa consiste en medir la temperatura de registrada por el sensor de temperatura y activar el color que corresponda a ese valor.

Hemos seleccionado una franja de temperatura que va desde 0°C a 41°C, y una selección de 11 colores. Para relacionar cada color con el rango de temperatura que le corresponde haz la siguiente división:

$$41^{\circ}\text{C (Temperatura máxima)} / 11 \text{ (número de colores)} = 3,72^{\circ}\text{C son los grados que deben de pasar entre color y color}$$

Puedes realizar esta actividad utilizando los *software* Arduino y Bitbloq, además de otros *software* de programación por bloques compatibles. A continuación encontrarás el código de programación necesario para cada *software*.

### Código Arduino

1. [Descarga el software Arduino](#) y realiza el proceso de instalación.
2. Añade la librería para el LED RGB al *software* Arduino siguiendo las instrucciones de la guía

[“Cómo añadir una librería para programar tu LED RGB”](#).

3. Abre el programa Arduino y, una vez en él, copia el siguiente programa:

```
/*
R   G   B   TEMP °C
255 0   0   41
244 80  43  37,27
244 102 26  33,54
248 147 31  29,81
243 223 6   26,09
147 198 45  22,36
23  173 66  18,63
40  209 188 14,90
48  59  175 11,18
58  33  117 7,45
105 46  152 3,72
155 36  152 0

Escala (41°C max / 11 colores = 3,72)
*/
int PortTemp = A0; // Sensor de temperatura -> puerto digital A0
float VoltTemp, Temp, ValueTemp; // Voltaje del sensor de
temperatura, Temperatura en °C, Valor de sensor de temperatura
#include <Adafruit_NeoPixel.h> //LIBRERÍA LED RGB
#define PIXEL_PIN 6 //PIN DEL LED RGB
#define PIXEL_COUNT 1 // NÚMERO DE LEDS RGB
Adafruit_NeoPixel strip = Adafruit_NeoPixel(PIXEL_COUNT, PIXEL_PIN,
NEO_GRB + NEO_KHZ800); // CONFIGURACIÓN DEL LED RGB

void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  //CONFIGURACIÓN DEL LED RGB
  strip.begin();
  strip.show();
}

void loop() {
  // Put your main code here, to run repeatedly:
  Temperature(); // Llama a la función para el cálculo de la
temperatura en °C
  Serial.println (Temp); // Mostrar el valor de la temperatura en
°C
  if (Temp>41)colorWipe(strip.Color(255,0,0),0); // Temperatura
superior a 41°C
```

```
    if ((Temp>37)&&(Temp<41))colorWipe(strip.Color(244,80,43),0); //
    Temperatura comprendida entre 41°C y 37°C

    if ((Temp>34)&&(Temp<37))colorWipe(strip.Color(244,102,26),0); //
    Temperatura comprendida entre 37°C y 34°C

    if ((Temp>30)&&(Temp<34))colorWipe(strip.Color(248,147,31),0); //
    Temperatura comprendida entre 34°C y 30°C

    if ((Temp>27)&&(Temp<30))colorWipe(strip.Color(243,223,6),0); //
    Temperatura comprendida entre 30°C y 27°C

    if ((Temp>23)&&(Temp<27))colorWipe(strip.Color(147,198,45),0); //
    Temperatura comprendida entre 27°C y 23°C

    if ((Temp>20)&&(Temp<23))colorWipe(strip.Color(23,173,66),0); //
    Temperatura comprendida entre 23°C y 20°C

    if ((Temp>17)&&(Temp<20))colorWipe(strip.Color(40,209,188),0); //
    Temperatura comprendida entre 20°C y 17°C

    if ((Temp>13)&&(Temp<17))colorWipe(strip.Color(48,59,175),0); //
    Temperatura comprendida entre 17°C y 13°C
    if ((Temp>10)&&(Temp<13))colorWipe(strip.Color(58,33,117),0); //
    Temperatura comprendida entre 13°C y 10°C

    if ((Temp>6)&&(Temp<10))colorWipe(strip.Color(105,46,152),0); //
    Temperatura comprendida entre 10°C y 6°C

    if ((Temp>0)&&(Temp<6))colorWipe(strip.Color(155,36,152),0); //
    Temperatura comprendida entre 6°C y 0°C
}

// FUNCIÓN PARA EL COLOR DEL LED RGB
void colorWipe(uint32_t c, uint8_t wait) {
    for(uint16_t i=0; i<strip.numPixels(); i++) {
        strip.setPixelColor(i, c);
        strip.show();
        delay(wait);
    }
}

void Temperature() // Cálculo de la Temperatura en °C
{
    ValueTemp = analogRead (PortTemp); // ValueTemp = Guarda el valor
    analógico del sensor de temperatura
```

```
VoltTemp = 5.0/1024*ValueTemp; //  
Temp = VoltTemp * 100-50; // Temp es el valor de la temperatura  
en °C  
delay(1000);  
}
```

4. Configura y carga el código, siguiendo las instrucciones indicadas en el [documento de Primeros Pasos de la placa Build&Code 4in1](#).

### **Código para el *software* de programación por bloques compatible**

1. [Descarga el software](#) y realiza el proceso de instalación.
2. Añade la extensión para programar el LED RGB siguiendo las instrucciones de la guía "[Cómo añadir una librería para programar tu LED RGB](#)".
3. Abre el programa y, una vez en él copia el siguiente código:

```

Programa de Arduino
INITIALIZE PIN 6 for 1 LEDS
por siempre
  fijar Temperature a (5.0 / 1024 * leer pin analógico (A) 0 * 100 - 50)
  si Temperature > 41 entonces
    SET LED nr. 0 STRIP to R 255 G 0 B 0
  si Temperature > 37 y Temperature < 41 entonces
    SET LED nr. 0 STRIP to R 244 G 80 B 43
  si Temperature > 34 y Temperature < 37 entonces
    SET LED nr. 0 STRIP to R 244 G 102 B 26
  si Temperature > 30 y Temperature < 34 entonces
    SET LED nr. 0 STRIP to R 248 G 147 B 31
  si Temperature < 30 y Temperature > 27 entonces
    SET LED nr. 0 STRIP to R 243 G 223 B 6
  si Temperature < 27 y Temperature > 23 entonces
    SET LED nr. 0 STRIP to R 147 G 198 B 45
  si Temperature < 23 y Temperature > 20 entonces
    SET LED nr. 0 STRIP to R 23 G 173 B 66
  si Temperature < 20 y Temperature > 17 entonces
    SET LED nr. 0 STRIP to R 40 G 209 B 188
  si Temperature > 13 y Temperature < 17 entonces
    SET LED nr. 0 STRIP to R 48 G 59 B 175
  si Temperature > 10 y Temperature < 13 entonces
    SET LED nr. 0 STRIP to R 58 G 33 B 117
  si Temperature > 6 y Temperature < 10 entonces
    SET LED nr. 0 STRIP to R 105 G 46 B 152
  si Temperature > 0 y Temperature < 6 entonces
    SET LED nr. 0 STRIP to R 155 G 36 B 152
  esperar 1 segundos
  
```

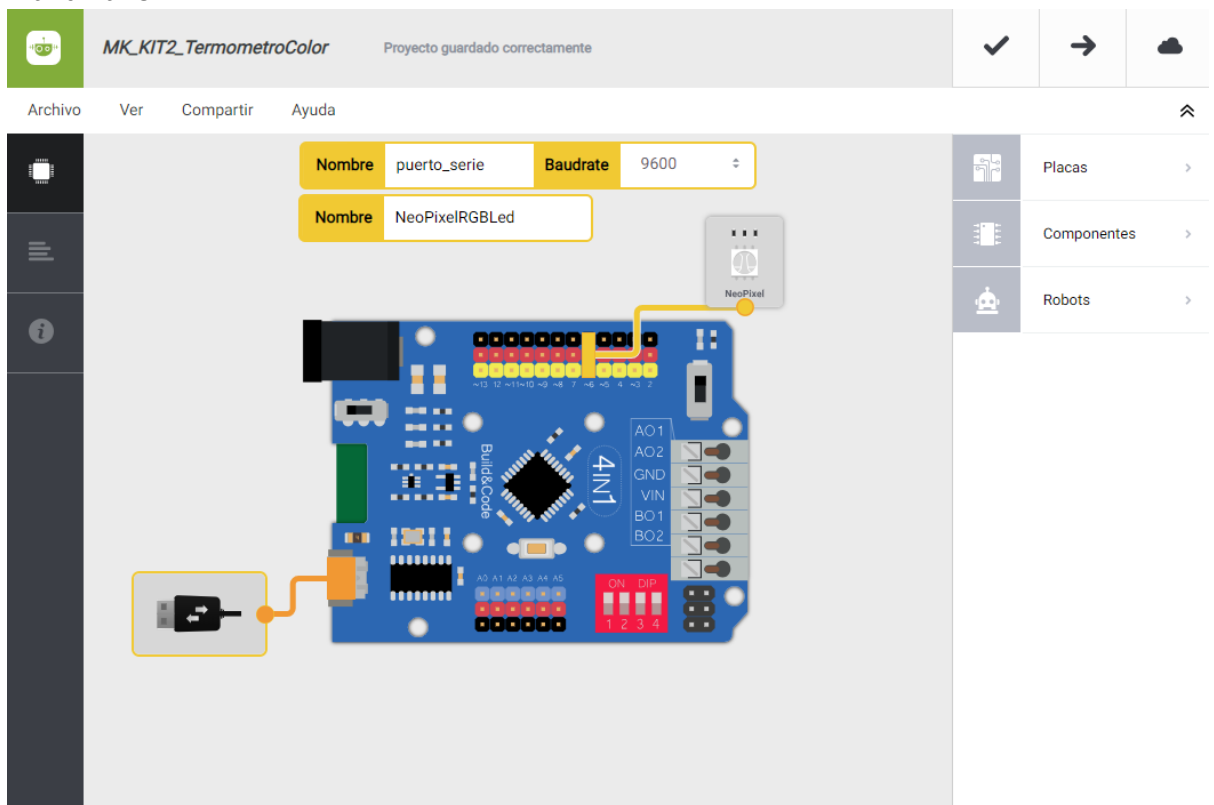


4. Configura y carga el código, siguiendo las instrucciones indicadas en el [documento de Primeros Pasos de la placa Build&Code 4in1](#).

## Código BitBloq

1. Accede al *software* [Bitbloq](#).
2. Abre el programa Bitbloq y, una vez en él, copia el siguiente código:

- o **Hardware**



- o **Software**

MK\_KIT2\_TermometroColor

Archivo Editar Ver Compartir Ayuda

Bloques Código

Componentes

- Fun Funciones
- Var Variables
- Cód Código
- Mat Matemáticas
- Tex Texto
- Con Control
- Lóg Lógica

— Variables globales y funciones

- Declarar variable Temperature con tipo decimal = 0
- Declarar variable VoltTemp con tipo decimal = 0
- Declarar variable ValueTemp con tipo decimal = 0

— Instrucciones iniciales (Setup)

Arrastra un bloque aquí para empezar tu programa

— Bucle principal (Loop)

Variable ValueTemp = Leer pin analógico A0

Variable VoltTemp = 5.0 / 1024 \* Variable ValueTemp

Variable Temperature = Variable VoltTemp \* 100 / 50

puerto\_serio Enviar Variable Temperature Con salto de línea

Si Variable Temperature > 41 ejecutar:

Encender el LED RGB NeoPixelRGBLed con un valor de rojo de 255, un valor de verde de 0 y un valor de azul de 0

Si Variable Temperature > 37 y Variable Temperature < 41 ejecutar:

Encender el LED RGB NeoPixelRGBLed con un valor de rojo de 244, un valor de verde de 80 y un valor de azul de 43

Si Variable Temperature > 34 y Variable Temperature < 37 ejecutar:

Encender el LED RGB NeoPixelRGBLed con un valor de rojo de 244, un valor de verde de 102 y un valor de azul de 25

Si Variable Temperature > 30 y Variable Temperature < 34 ejecutar:

Encender el LED RGB NeoPixelRGBLed con un valor de rojo de 248, un valor de verde de 147 y un valor de azul de 31

Si Variable Temperature > 27 y Variable Temperature < 30 ejecutar:

Encender el LED RGB NeoPixelRGBLed con un valor de rojo de 243, un valor de verde de 223 y un valor de azul de 6

Si Variable Temperature > 23 y Variable Temperature < 27 ejecutar:

Encender el LED RGB NeoPixelRGBLed con un valor de rojo de 147, un valor de verde de 198 y un valor de azul de 45

Si Variable Temperature > 20 y Variable Temperature < 23 ejecutar:

Encender el LED RGB NeoPixelRGBLed con un valor de rojo de 23, un valor de verde de 173 y un valor de azul de 66

Si Variable Temperature > 17 y Variable Temperature < 20 ejecutar:

Encender el LED RGB NeoPixelRGBLed con un valor de rojo de 40, un valor de verde de 209 y un valor de azul de 188

Si Variable Temperature > 13 y Variable Temperature < 17 ejecutar:

Encender el LED RGB NeoPixelRGBLed con un valor de rojo de 48, un valor de verde de 59 y un valor de azul de 175

Si Variable Temperature > 10 y Variable Temperature < 13 ejecutar:

Encender el LED RGB NeoPixelRGBLed con un valor de rojo de 58, un valor de verde de 33 y un valor de azul de 117

Si Variable Temperature > 6 y Variable Temperature < 10 ejecutar:

Encender el LED RGB NeoPixelRGBLed con un valor de rojo de 105, un valor de verde de 46 y un valor de azul de 152

Si Variable Temperature > 0 y Variable Temperature < 6 ejecutar:

Encender el LED RGB NeoPixelRGBLed con un valor de rojo de 155, un valor de verde de 36 y un valor de azul de 152

Esperar 1000 ms

3. Configura y carga el código, siguiendo las instrucciones indicadas en el [documento de Primeros Pasos de la placa Build&Code 4in1](#).

## **RESULTADO DEL EJERCICIO**

Como resultado del ejercicio, el termómetro comunica la temperatura del ambiente a través de un código de color. ¡Has programado tu propio termómetro de colores!