



Programación 3: A

Nombres y apellidos: Jorge Eduardo Cabrera Talavera

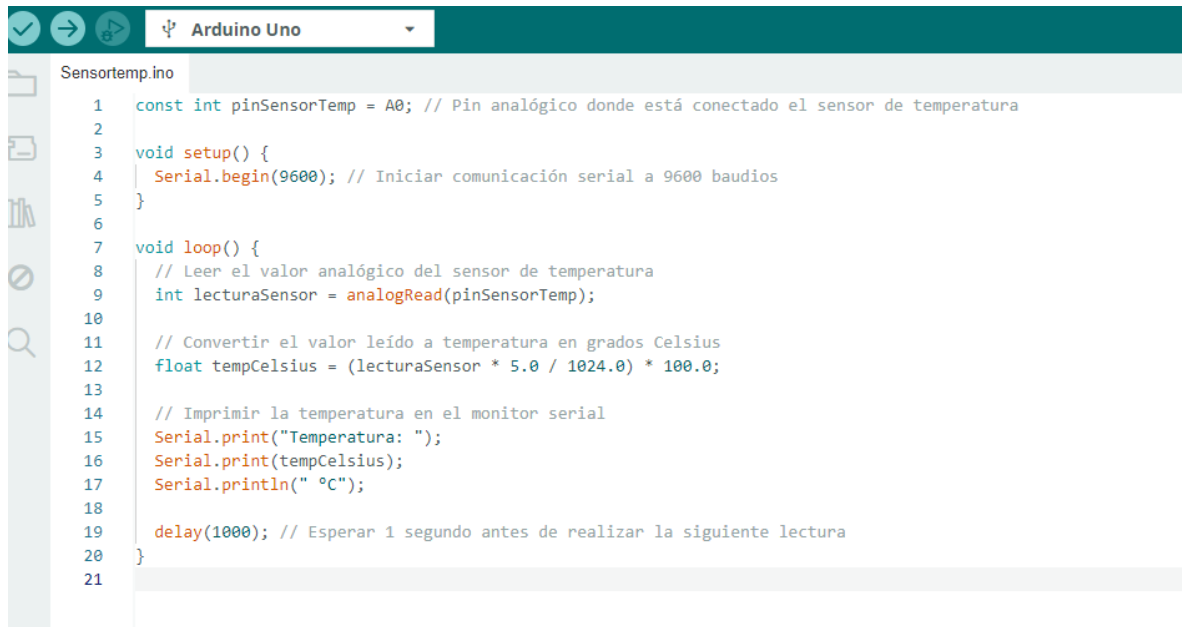
Carnet: 0901-22-628

Introducción

En este laboratorio trabajamos con un sensor de temperatura y un contador, ambos circuitos no tienen mayor dificultad para poder comprender su funcionamiento pero al menos con el sensor de temperatura hay que ser bastante cuidadosos con los cables ya que utiliza varios, hay que verificar que todas las conexiones estén bien realizadas, mas allá de esto el laboratorio es bastante fácil pero a la vez aprendemos bastante sobre nuevos dispositivos que se pueden ir agregando y dándoles funcionalidades con lo que hemos aprendido durante el curso.

Contenido

Código Arduino (Sensor de Temperatura)



```
1  const int pinSensorTemp = A0; // Pin analógico donde está conectado el sensor de temperatura
2
3  void setup() {
4      Serial.begin(9600); // Iniciar comunicación serial a 9600 baudios
5  }
6
7  void loop() {
8      // Leer el valor analógico del sensor de temperatura
9      int lecturaSensor = analogRead(pinSensorTemp);
10
11     // Convertir el valor leído a temperatura en grados Celsius
12     float tempCelsius = (lecturaSensor * 5.0 / 1024.0) * 100.0;
13
14     // Imprimir la temperatura en el monitor serial
15     Serial.print("Temperatura: ");
16     Serial.print(tempCelsius);
17     Serial.println(" °C");
18
19     delay(1000); // Esperar 1 segundo antes de realizar la siguiente lectura
20 }
21
```

Corrida e imagenes

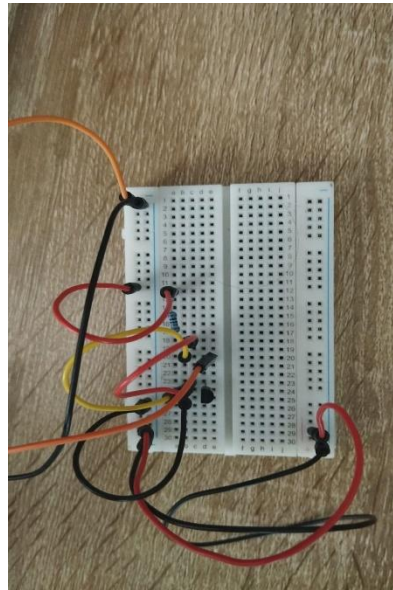


```
15     Serial.print("Temperatura: ");
16     Serial.print(tempCelsius);
17     Serial.println(" °C");
18
19     delay(1000); // Esperar 1 segundo antes de realizar la siguiente lectura
20 }
21
```

Output Serial Monitor x

Message (Enter to send message to 'Arduino Uno' on 'COM3')

```
Temperatura: 41.02 °C
Temperatura: 37.60 °C
Temperatura: 36.62 °C
```



Código Arduino (Contador)

```
Arduino Uno
codigonumeros.ino

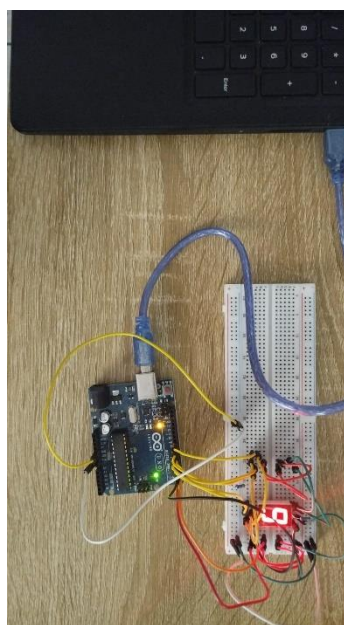
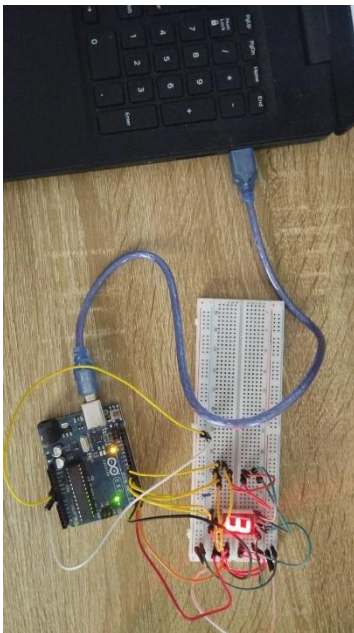
1 // Definir pines para cada segmento del display
2 const int segmentos[] = {2, 3, 4, 5, 6, 7, 8};
3 const int punto = 9; // Pin para el punto decimal
4 const int totalSegmentos = sizeof(segmentos) / sizeof(segmentos[0]);
5
6 // Definir los patrones para cada número
7 const byte numeros[10][8] = {
8   {1, 1, 1, 1, 1, 1, 0, 0}, // 0
9   {0, 1, 1, 0, 0, 0, 0, 0}, // 1
10  {1, 1, 0, 1, 1, 0, 1, 0}, // 2
11  {1, 1, 1, 1, 0, 0, 1, 0}, // 3
12  {0, 1, 1, 0, 0, 1, 1, 0}, // 4
13  {1, 0, 1, 1, 0, 1, 1, 0}, // 5
14  {1, 0, 1, 1, 1, 1, 1, 0}, // 6
15  {1, 1, 1, 0, 0, 0, 0, 0}, // 7
16  {1, 1, 1, 1, 1, 1, 1, 0}, // 8
17  {1, 1, 1, 1, 0, 1, 1, 0} // 9
18 };
19
20 void setup() {
21   configurarPines();
22 }
23
24 void loop() {
25   for (int i = 0; i < 10; i++) {
26     mostrarNumero(i);
27     delay(1000); // Esperar un segundo
28   }
29 }
```

```

codigonumeros.ino
27 |   delay(1000); // Esperar un segundo
28 | }
29 |
30 |
31 | // Función para configurar los pines
32 | void configurarPines() {
33 |   // Configurar los pines de los segmentos como salida
34 |   for (int i = 0; i < totalSegmentos; i++) {
35 |     pinMode(segmentos[i], OUTPUT);
36 |   }
37 |   // Configurar el pin del punto como salida
38 |   pinMode(punto, OUTPUT);
39 | }
40 |
41 | // Función para mostrar un número en el display
42 | void mostrarNumero(int numero) {
43 |   // Apagar todos los segmentos y el punto
44 |   apagarDisplay();
45 |   // Encender los segmentos correspondientes al número
46 |   for (int i = 0; i < totalSegmentos; i++) {
47 |     digitalWrite(segmentos[i], !numeros[numero][i]); // Ánodo común
48 |   }
49 |   digitalWrite(punto, !numeros[numero][totalSegmentos]); // Encender el punto si está activo
50 | }
51 |
52 | // Función para apagar todos los segmentos y el punto
53 | void apagarDisplay() {
54 |   for (int i = 0; i < totalSegmentos; i++) {
55 |     digitalWrite(segmentos[i], HIGH); // Apagar segmentos
56 |   }
57 |   digitalWrite(punto, HIGH); // Apagar el punto
58 | }
59 |
60 |

```

Imagenes



Link de los archivos

<https://github.com/JorgeCabrera19/Laboratorio07>

Conclusión

El resultado de estos ejercicios es bastante satisfactorio ya que los dispositivos utilizados tienen mucha importancia en la vida real y son de bastante utilidad, entonces podemos ver como ya estamos programando con dispositivos que ya tienen una funcionalidad que puede ser bastante importante y que podríamos incluso empezar ya a utilizarlo en proyectos personales para diferentes situaciones.

Existen varios dispositivos más que podemos utilizar con Arduino y programar para tener funcionalidades bastante interesantes como controles remotos, sistemas MIDI, carros electrónicos y demás.

Bibliografía

Alejandro, I. C. (2024). Programación 3. *Universidad Mariano Galvez*. Guatemala: UMG.