

Alumno: Joaquin Cortabarría

Carrera: TS Telecomunicaciones - Fecha: Octubre 2023

Proyecto Integrador Programación: Evidencia 10 - Capa física.

Se diseñó la capa física del dispositivo, se diagramó en el simulador Wokwi, y se construyó un prototipo para hacer las pruebas físicas.

La idea supuesta es controlar la posición de un deflector de flujo de aire a través del servo, para modificar el caudal de aire frío que incide en el ambiente donde se encuentra el sensor térmico. De manera que la posición del servo (y el deflector) asumirá la siguiente conducta:

Temperatura (°C)	Posición del Servo (°)
> 30.0	90
> 25.0	50
> 22.0	25
Menor o igual a 22.0	0

En el siguiente diagrama se puede observar la utilización de:

- Placa de desarrollo Arduino.
- un servo como actuador
- sensor de temperatura analogico NTC (el que dispone Woki, ya incorpora la resistencia)
- un potenciómetro (para simular estado de batería)

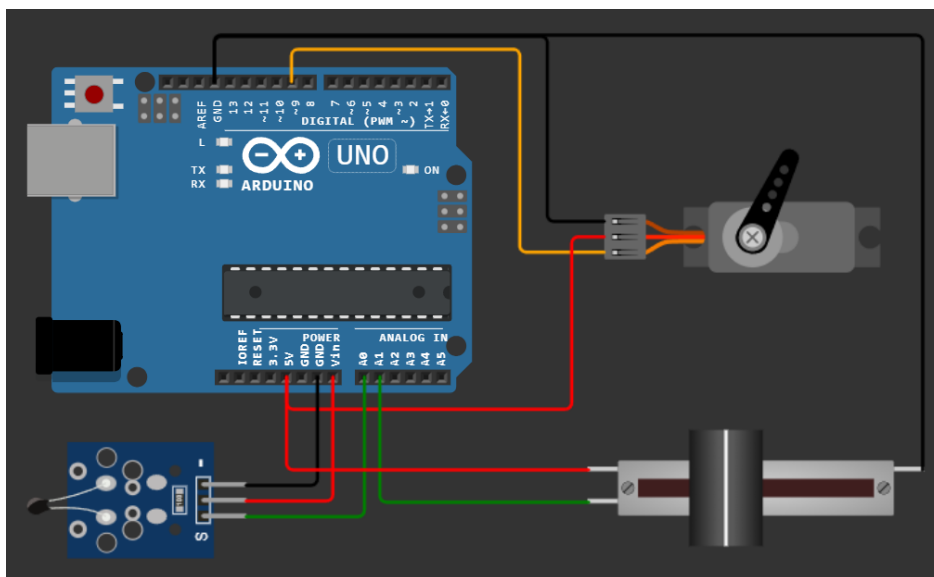


Diagrama en el simulador Wokwi. [acceso al proyecto](#)

Al ejecutar la simulación del proyecto en Wokwi obtenemos la siguiente salida a través del serial.

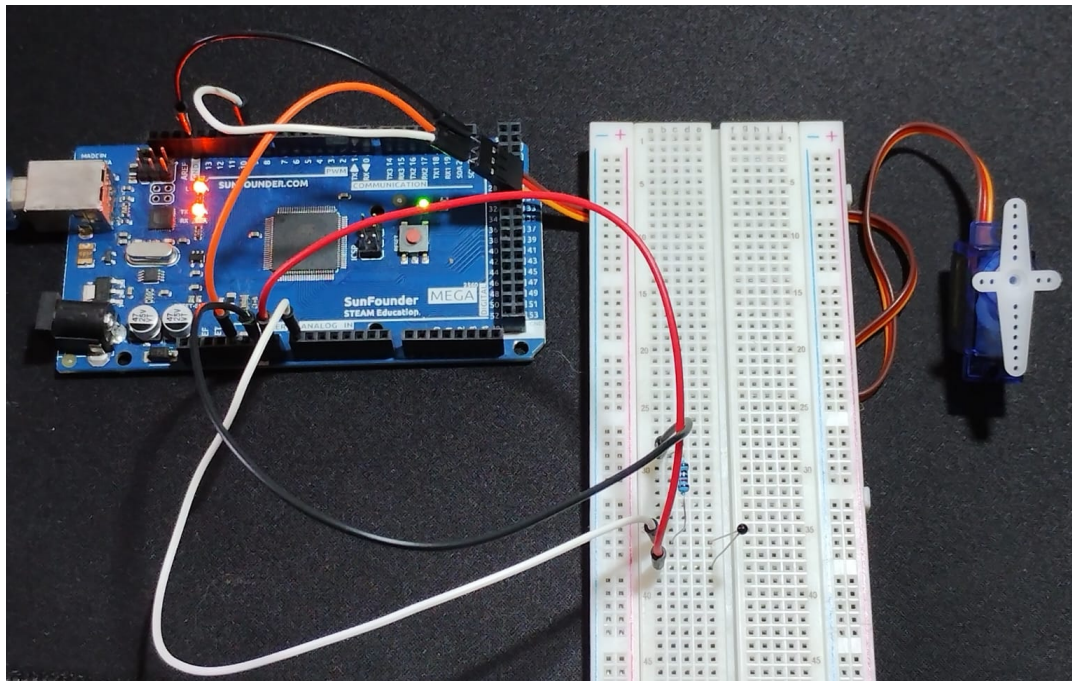
```
Temperatura: 23.99 °C  
Posicion deflector: 25°  
Bateria: 34 %
```

- Temperatura sensada en el termistor NTC
- Posición del deflector, a través del servo
- nivel de batería simulado con el potenciómetro.

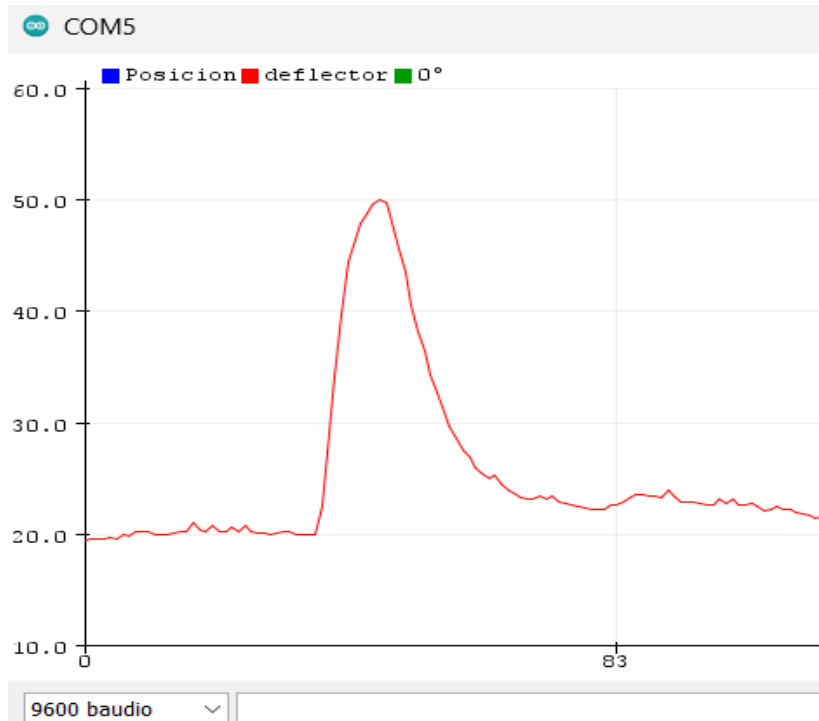
El código puede verse en el repositorio y en el anexo de este documento.

Prototipo físico

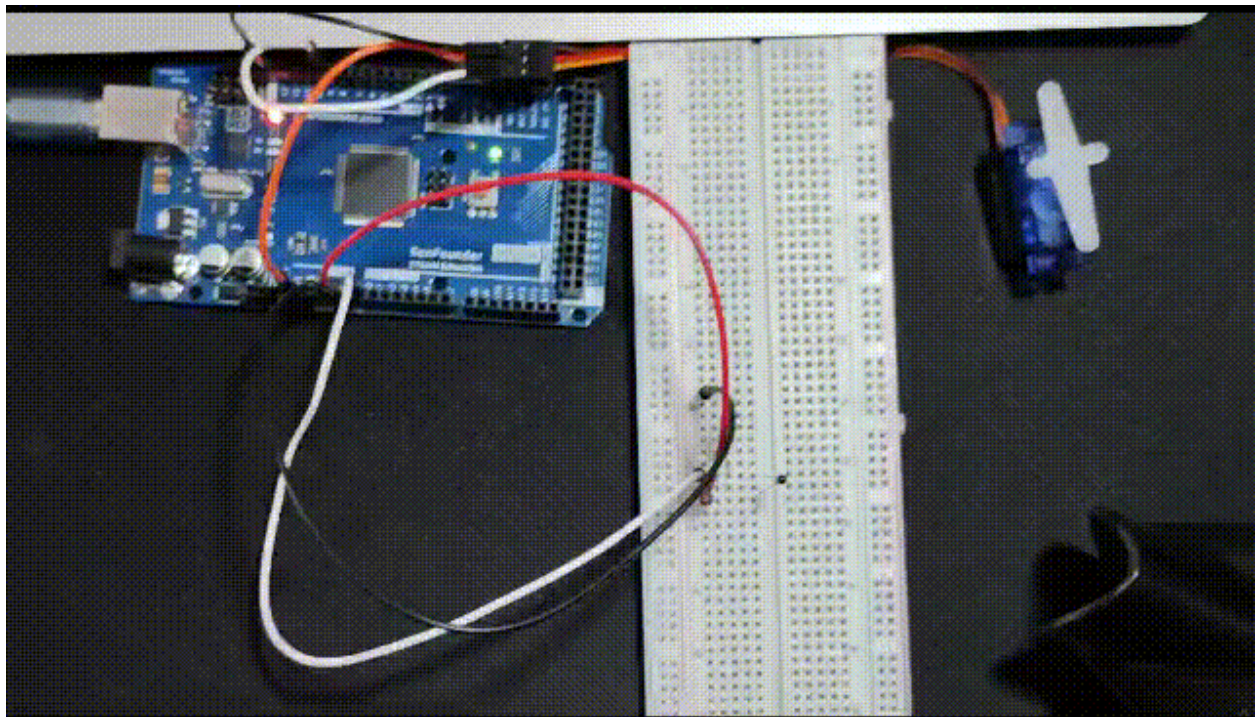
Se construyó el prototipo físico, con el fin de comprobar el funcionamiento correcto y realizar pruebas para la calibración final.



Al afectar el termistor con un flujo de aire caliente generado con un secador de pelo, observamos que el comportamiento del actuador es el esperado ([ver video en formato Gif](#)). Y obtenemos la siguiente respuesta a través del puerto serial.



Captura de video



ANEXOS:

Repositorio:

Los siguientes archivos del proyecto se encuentran en el [repositorio](#)

TermoActuador.ino

Video_calibracion_TermoActuador.gif

Wokwi_termoActuador.zip

Proyecto en Wokwi:

Aquí esta el [Link de acceso](#) del proyecto en el simulador Wokwi.

Codigo:

```
/*
Prototipo de dispositivo que controla la posicion de un deflector con un servo,
segun la temperatura sensada por un termistor.
se agrego un potenciometro para simular el estado de carga de la bateria.
*/

#include <Servo.h> //inxcluimos la libreria servo

Servo myservo; // crear el objeto servo

const float BETA = 3950; // coeficiente del termistor
int term = 0; // pin analogico donde se conecta el termistor
int val; // variable para escribir la salida al servo
int potpin = 1; // pin analogico para conectar el potenciometro simulador de
bateria
int simbate; // variable para escribir la simulacion de bateria desde el
potenciometro

void setup() {
  myservo.attach(9); // salida al servo en el pin 9
  Serial.begin(9600); // incia la comunicacion serial
}

void loop() {

  int analogValue = analogRead(term); //leer el pin del termistor
  float celsius = 1 / (log(1 / (1023. / analogValue - 1)) / BETA + 1.0 /
298.15) - 273.15; //calcula temperatura en celsius
```

```
if (celsius > 30.0) {  
    myservo.write(90); // Posiciona el servo en 90° si la temperatura supera  
los 30°C.  
    val=90;  
} else if (celsius > 25.0) {  
    myservo.write(50); // Posiciona el servo en 50° si la temperatura supera  
los 25°C.  
    val=50;  
} else if (celsius > 22.0) {  
    myservo.write(25); // Posiciona el servo en 25° si la temperatura supera  
los 22°C.  
    val=25;  
} else {  
    myservo.write(0); // Mantén el servo en 0° en otros casos.  
    val=0;  
}  
  
myservo.write(val); // posiciona el servo segun la variable  
  
simbate = analogRead(potpin); // lee el pin del potenciometro  
simbate = map(simbate, 0, 1023, 0, 100); // traduce la lectura del  
potenciometro a porcentaje  
  
// presentacion de resultados por interface serial  
Serial.print("Temperatura: ");  
Serial.print(celsius);  
Serial.println(" °C");  
Serial.print("Posicion deflector: ");  
Serial.print(val);  
Serial.println("");  
Serial.print("Bateria: ");  
Serial.print(simbate);  
Serial.println(" %");  
  
delay(1000); // retraso del loop  
}
```

Link textual del repositorio:

https://github.com/patriciodelsacramento/ABM-DispositivosInteligentesSRL/tree/main/Evidencias_Programacion/10_CapaFisica