

LAB 2: Sensores y Actuadores

PARTE II: Actuadores

La presente evaluación será recibida sólo si se ha cumplido con los checkpoints de todas las preguntas en el **LABORATORIO** con el docente del curso.

La entrega debe constar de:

1. **Introducción** (incluir párrafo introductorio):
 1. **Marco Teórico** (definiciones con referencias)
 2. **Estado del Arte** (trabajos de investigación, artículos, tesis, páginas con referencias).
2. **Metodología** (explicación de componentes e implementos utilizados y diagramas de flujo/bloques).
3. **Desarrollo de la experiencia** (incluye discusión)
4. **Conclusiones** (una por cada experiencia realizada como mínimo)
5. **Referencias** (formato IEEE)

Consideraciones:

- Subir únicamente el PDF del informe a Canvas, usar un link para el repositorio, incluir fotos de paso a paso de cada implementación y un vídeo mostrando el funcionamiento de las experiencias. Sin embargo, para simulaciones, considere la placa Arduino UNO disponible en Tinkercad.
- Considere una placa de Arduino MEGA2560 tal cual existe en el laboratorio.
- Solo un miembro del grupo sube el informe.
- No olvidar colocar los nombres de todos los integrantes.

ÍNDICE

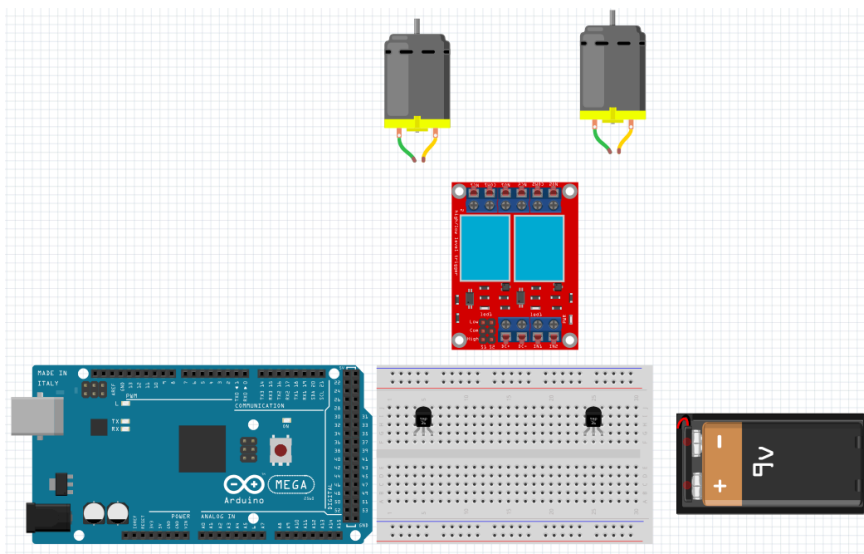
1. [Solo Simulación] Control de ventiladores	2
1.1. Materiales	2
1.2. Requisitos	3
1.3. presentar:	3

PARTE II: Actuadores

1. [Solo Simulación] Control de ventiladores

En aplicaciones domésticas, el monitoreo y respuesta automática a condiciones ambientales son fundamentales para mejorar el confort y la eficiencia energética.

En esta parte de la experiencia se diseñará un sistema de control básico que simula el comportamiento de ventilación automática en una vivienda de dos habitaciones. Para ello, se utilizarán sensores de temperatura en cada habitación.



1.1. Materiales

- 2 Sensor de temperatura
- 2 motores DC
- 2 relés
- Fuente externa de alimentación

1.2. Requisitos

- Leer correctamente la temperatura de cada sensor conectado al Arduino.
- Comparar la temperatura leída con un umbral fijo (por ejemplo, 28 °C).
- Si la temperatura de una habitación supera el umbral, activar el motor correspondiente mediante un transistor.
- Mostrar los valores de temperatura en el monitor serial.

1.3. presentar:

1. Simulación en tinkercad.
2. Diagrama de bloques del sistema.

Checkpoint 12

ANEXO

2. Sensor de temperatura TMP36

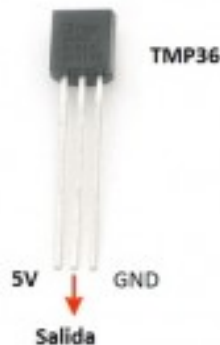


Figure 1 Sensor de temperatura TMP36

El fabricante de los sensores de temperatura **TMP36** y **LM35DZ** especifica que la salida de tensión de estos dispositivos es de **10 mV/°C**. Adicionalmente, se conoce que el microcontrolador **Arduino Uno**, en sus entradas analógicas, proporciona una lectura digital que varía linealmente entre **0 y 1023** en función de un voltaje de entrada entre **0 V y 5 V**. Por lo tanto, el valor en voltios correspondiente a una lectura analógica puede determinarse mediante la siguiente expresión:

$$\text{Voltaje} = \text{LecturaAnalógica} * 5 / 1024$$

Dado que **1 V** representa una variación de **100 °C** (pues $1 \text{ V} / 10 \text{ mV} = 100$), la temperatura en grados Celsius se puede calcular multiplicando el voltaje obtenido por 100.

Sin embargo, en el caso particular del sensor **TMP36**, el fabricante indica que su salida de **0 V** no corresponde a **0 °C**, sino a **-50 °C**, lo que permite realizar mediciones por debajo de cero. En consecuencia, se debe aplicar una corrección de **-50 °C** al resultado final. En contraste, el **LM35DZ** presenta un punto de partida de **0 °C**, por lo que no requiere ninguna corrección adicional, lo que facilita su uso en aplicaciones donde no se requiere medir temperaturas negativas.

En resumen, para obtener la temperatura en grados Celsius a partir de una lectura analógica en Arduino se utilizan las siguientes fórmulas:

- Para el **LM35DZ**:

$$\text{Temperatura (°C)} = (\text{lectura_analógica} \times 5.0 \times 100) / 1024$$

- Para el **TMP36**:

$$\text{Temperatura (°C)} = (\text{lectura_analógica} \times 5.0 \times 100) / 1024 - 50$$

Cabe destacar que existen múltiples sensores de temperatura con distintos rangos de medida, tanto superiores como inferiores, sin embargo, el procedimiento general de calibración permanece válido para la mayoría de ellos, con ajustes según el offset especificado por el fabricante.

Dado que el cálculo de temperatura involucra operaciones de división y puede generar valores no enteros, se recomienda declarar las variables involucradas como tipo **float**, con el fin de preservar la precisión decimal del resultado. Aunque el uso de variables de tipo entero (**int**) es común, su empleo no es adecuado cuando se realizan operaciones aritméticas con posibles resultados decimales.

Finalmente, en el ciclo de medición, se recomienda realizar una lectura por segundo, utilizando un **retardo (delay)** temporal, ya que la temperatura es una magnitud que varía lentamente. Si la temperatura medida supera un umbral predefinido, se deberá activar una alarma; en caso contrario, esta permanecerá desactivada.