

---

# MEDIDOR DE TEMPERATURA CON TERMOPAR TIPO K

Edgar Omar Martínez Vega  
al289664@edu.uaa.mx

**RESUMEN:** En este proyecto se llevo a cabo un medidor de temperatura haciendo uso de un Termopar Tipo K, el cual viene con el módulo MAX6675.

Para mostrar las temperaturas medidas se hizo utilizó una pantalla LCD de 4 renglones por 20 columnas, la cual fue controlada mediante un módulo I2C.

Este Medidor de Temperatura fue realizado en una placa de desarrollo Raspberry Pi Pico, el cual fue programado con Python.

Este Medidor de Temperatura muestra la temperatura en las escalas Celsius, Fahrenheit, Kelvin y Rankine, además muestra el voltaje medido por el termopar.

**PALABRAS CLAVE:** Termopar, Medidor, Escalas de Temperatura, Pantalla LCD.

## 1 OBJETIVO:

El alumno comprobará el funcionamiento del Termopar Tipo K mediante la elaboración de un Medidor de Temperatura empleando las diferentes escalas de temperatura, realizando los circuitos de acondicionamiento necesarios.

## 2 MATERIALES:

- 1 Termopar Tipo K
- 1 Modulo MAX 6675
- 1 Raspberry Pi Pico
- 1 Pantalla LCD 4X20
- 1 Modulo I2C para LCD
- Cables para conexión
- Protoboard

## 3 MARCO TEORICO:

Un termopar o termocupla es un transductor formado por la unión de dos metales distintos que produce una diferencia de potencial muy pequeña (del orden de los milivolts) esta es función de la diferencia de temperatura entre uno de los extremos denominado "Punto Caliente" o "Unión Caliente" o de "Medida" y el otro llamado "Punto Frío" o "Unión Fría" o de "Referencia" (efecto Seebeck).

En instrumentación industrial, los termopares son usados como sensores de temperatura. Son económicos, intercambiables, tienen conectores estándar y son capaces de medir un amplio rango de temperaturas. Su principal limitación está en la exactitud, pues es fácil obtener errores del sistema cuando se trabaja con precisiones inferiores a un grado Celsius. La Figura 1 muestra como se ve un Termopar de Tipo K.

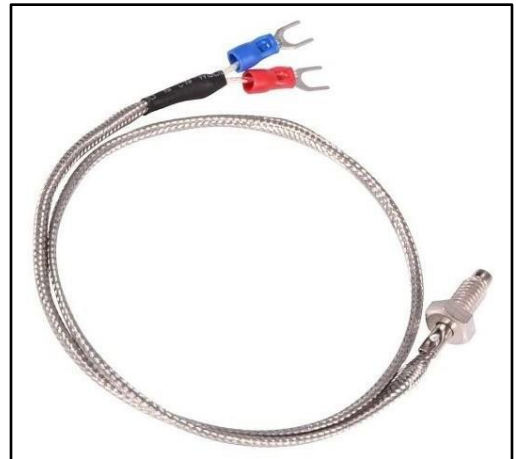


Figura 1.- Termopar Tipo K

El MAX6675 es un módulo interfaz para termopar tipo K que permite acondicionar la señal analógica que entrega un sensor de tipo termopar para que pueda ser leída de forma digital por un microcontrolador.

Dentro del MAX6675 se encuentra la electrónica necesaria para amplificar, compensar y convertir a digital el voltaje generado por el termopar, lo que hace muy sencilla la tarea de conectar un sensor de este tipo a un microcontrolador.

El módulo interfaz para termopar tipo K es una excelente forma de comenzar a medir temperaturas elevadas con cualquier microcontrolador, ya que incluye el MAX6675 en una práctica tarjeta breakout y un termopar tipo K de forma que puedas comenzar a experimentar cuanto antes. La Figura 2 nos muestra el Módulo MAX6675.

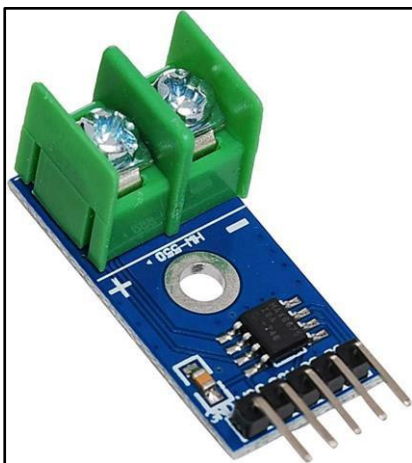


Figura 2.- Módulo MAX6675

## 4 DESARROLLO:

### 4.1 CODIGO DEL MICROCONTROLADOR:

Lo primero que se realizó en este proyecto, fue el desarrollo del código para el microcontrolador, en dicho código se realiza el muestreo de temperatura mediante el termopar y la conversión a las distintas escalas de temperatura, ya que el termopar trabaja en grados Celsius.

Para convertir la temperatura de grados Celsius a grados Fahrenheit, se emplea la Ecuación 1.

$$^{\circ}\text{F} = \left(^{\circ}\text{C} + \frac{9}{5}\right) + 32 \quad (\text{Ec } 1)$$

Para el caso de la conversión de grados Celsius a grados Kelvin, se utilizó la Ecuación 2.

$$K = ^{\circ}\text{C} + 273.5 \quad (\text{Ec } 2)$$

Finalmente, para pasar de Celsius a grados Rankine, utilizamos la Ecuación 3.

$$^{\circ}\text{R} = ^{\circ}\text{C} * \frac{9}{5} + 491.67 \quad (\text{Ec } 3)$$

El código es muy sencillo, esto debido a que al usar el módulo MAX6675 no es necesario el acondicionamiento de señal para el termistor, ya que este módulo se encarga de esa parte. En el código solo es necesario realizar la lectura del termistor mediante el pin SO, almacenamos la lectura obtenida y realizamos las conversiones necesarias para desplegar las distintas escalas de temperatura en la pantalla LCD. En la Figura 3 se muestran las librerías utilizadas, así como la declaración de los pines para la LCD y para el módulo MAX6675.

```
1  #librerias, I2C, Y LCD
2  import utime
3  import machine
4
5  from machine import Pin, I2C
6  from lcd_api import LcdApi
7  from pico_i2c_lcd import I2cLcd
8  from max6675 import MAX6675
9
10 I2C_ADDR = 0x27
11 I2C_NUM_ROWS = 4
12 I2C_NUM_COLS = 20
13
14 #DECLARACION MAX
15 so = Pin(15, Pin.IN)
16 sck = Pin(13, Pin.OUT)
17 cs = Pin(14, Pin.OUT)
18
19 max = MAX6675(sck, cs, so)
```

Figura 3.- Librerías y Pines

La Figura 4 muestra el resto del código, donde se inicializa la LCD y se imprimen las temperaturas medidas en el ciclo while.

```
22 def test_main():
23     i2c = I2C(0, sda=machine.Pin(0), scl=machine.Pin(1), freq=400000)
24     lcd = I2cLcd(i2c, I2C_ADDR, I2C_NUM_ROWS, I2C_NUM_COLS)
25     lcd.putstr("Toma de temperatura")
26     utime.sleep(2)
27     lcd.clear()
28     count = 0
29
30     while True:
31
32         #DEFINICIONES DE LA TEMPERATURAS:
33         temp = max.read()
34         rankine = temp * 1.8 + 491.67
35         kelvin = temp + 273.15
36         fahrenheit = (temp * 9/5) + 32
37         print(temp)
38
39         #Voltaje (5 volt es el voltaje maximo y 1024 la temperatura
40         # maxima segun el datasheet, temp es el dato retornado por el sensor)
41         voltaje = (5/1024)*temp
42
43         #RANKINE
44         lcd.move_to(0, 0)
45         lcd.putstr("R:" + "{:.1f}".format(rankine))
46
47         #CELSIUS
48         lcd.move_to(0, 1)
49         lcd.putstr("C:" + "{:.1f}".format(temp))
50
51         #KELVIN
52         lcd.move_to(0, 2)
53         lcd.putstr("K:" + "{:.1f}".format(kelvin))
54
55         #FAHRENHEIT
56         lcd.move_to(0, 3)
57         lcd.putstr("F:" + "{:.1f}".format(fahrenheit))
58
59         #VOLTAGE
60         lcd.move_to(13, 1)
61         lcd.putstr("{:.3f}".format(voltaje) + str("V"))
62
63     #while True
64     test_main()
```

Figura 4.- Código del Microcontrolador

En la figura 4 se observa el uso de las Ecuaciones 1, 2 y 3 para la conversión entre las distintas escalas de temperatura; también se observa el calculo del voltaje, respecto a la entrada del ADC.

## 4.2 DIAGRAMA DE CONEXION:

En la Figura 5 se observa el diagrama de conexiones para nuestro medidor de temperatura, en este diagrama se observa a que pines de la Raspberry se conectaron nuestros componentes, en este caso el módulo MAX6675 y la pantalla LCD.

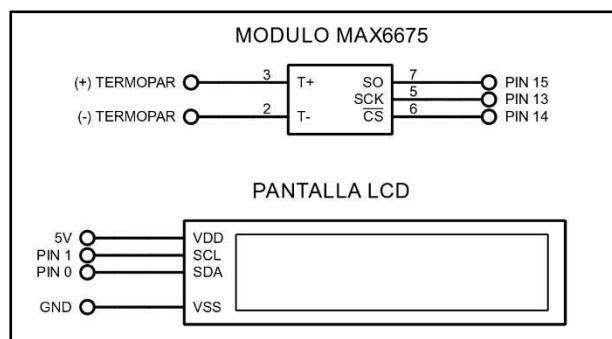


Figura 5.- Diagrama de Conexión

## 4.3 RESULTADOS:

Finalmente, la Figura 6 muestra el circuito del medidor de temperatura armado de manera física en el protoboard; en la figura se observa la temperatura medida en las distintas escalas mencionadas anteriormente.

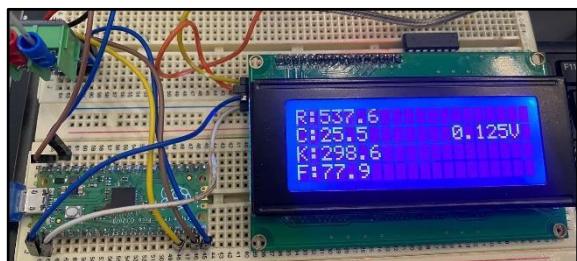
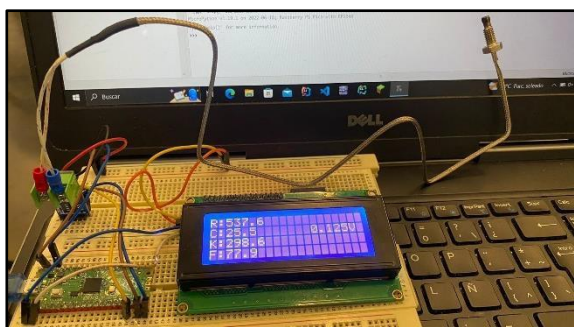


Figura 6.- Medidor de Temperatura

En la Figura 7, se observa el circuito mencionado anteriormente y el Termopar conectado a dicho circuito.



## 5 CONCLUSION:

Realizar este proyecto nos permitió complementar lo aprendido sobre los termopares en las sesiones teóricas de clase.

El uso del termopar tipo K y el módulo MAX6675 permitió obtener mediciones de temperatura precisas y confiables en un amplio rango de temperaturas. Utilizar La Raspberry Pi Pico fue una gran elección, ya que es una plataforma versátil y de alto rendimiento para controlar y procesar los datos.

La utilización del módulo I2C para la comunicación con la pantalla LCD simplificó la conexión y minimizó la complejidad del cableado, lo que contribuyó a una mejor organización y una implementación más eficiente del sistema.

El resultado final del proyecto fue un medidor de temperatura funcional y de bajo costo, con una interfaz clara y legible en la pantalla LCD. La precisión de las mediciones, combinada con la facilidad de uso y la flexibilidad de la Raspberry Pi Pico, lo convierte en una solución efectiva para aplicaciones que requieren monitoreo de temperatura en tiempo real.

## 6 REFERENCIAS:

- [1] Geek Factory. (2023, 11 mayo). MAX6675 módulo interfaz para termopar tipo K - Geek Factory. <https://www.geekfactory.mx/tienda/modulos/max6675-modulo-interfaz-para-termopar-tipo-k/>

