



INSTITUTO POLITECNICO NACIONAL
ESCUELA SUPERIOR DE CÓMPUTO

INSTRUMENTACIÓN Y CONTROL
PRÁCTICA 5

**Sensor de temperatura PT100 con Display en
Arduino**

Equipo: Méndez Hernández Yael

Méndez Pichardo Daniel

Melo Jiménez Jesús

Segura Gutiérrez Obed

Alva Galicia Diego Sebastián

Grupo: 5CM4

Profesor: Alcántara Méndez Alberto Jesús

Índice

Objetivo	3
Código.....	4
Video	7
Conclusiones.....	8
Méndez Pichardo Daniel	8
Melo Jiménez Jesús Uriel.....	8
Segura Gutiérrez Obed	8
Méndez Hernández Yael	8
Alva Galicia Diego Sebastián.....	9
Sello y Firma.....	10

Objetivo

Analizar el tiempo de respuesta, el comportamiento del sensor de temperatura, la velocidad de lectura del circuito acondicionador hacia el Arduino. Diseñar e Implementar una interfaz gráfica, Aplicar sus conocimientos de codificación en lenguaje programación Python y C para poder recibir los datos del circuito de la práctica anterior del sensor a la computadora.

Código

Código para la gráfica en Python

```
import tkinter as tk

from matplotlib.backends.backend_tkagg import FigureCanvasTkAgg

from matplotlib.figure import Figure

import serial

import threading

running = True

ser = None

voltaje_data = [0] * 20

voltaje_data2 = []

try:

    ser = serial.Serial('COM4', 9600)

except serial.SerialException:

    print("Error al abrir el puerto serial.")

    running = False

def update_graph(data_array):

    ax.cla()

    ax.plot(data_array, marker='o', linestyle='-', color='b', label='Temperatura')

    ax.set_title('Temperatura en Tiempo Real')

    ax.set_title(f'Temperatura: {voltaje_data2} grados')

    ax.set_xlabel('Tiempo')

    ax.set_ylabel('Temperuta (C)', color='b')

    ax.set_ylim(0, 40)

    ax.legend(loc='upper left')

    canvas.draw_idle()

def read_serial():

    global voltaje_data

    global voltaje_data2
```

```

while running:

    if ser.in_waiting:

        data = ser.readline().decode('utf-8').rstrip()

        try:

            voltaje = float(data)

            voltaje_data = voltaje_data[1:] + [voltaje]

            voltaje_data2 = voltaje

            update_graph(voltaje_data)

        except ValueError:

            pass

root = tk.Tk()
root.title("Voltaje en Tiempo Real")
fig = Figure(figsize=(10, 8), dpi=100)
ax = fig.add_subplot(111)
ax.set_xlabel('Tiempo')
ax.set_ylabel('Temperatura (C)')
ax.set_title('Gráfica en Tiempo Real de Temperatura')
ax.set_ylim(0, 40)
canvas = FigureCanvasTkAgg(fig, master=root)
canvas.get_tk_widget().pack(fill=tk.BOTH, expand=True)
canvas.draw()

if running:

    threading.Thread(target=read_serial, daemon=True).start()

def on_closing():

    global running

    running = False

    if ser:

        ser.close()

    root.destroy()

root.protocol("WM_DELETE_WINDOW", on_closing)

```

```
root.mainloop()
```

Código para Arduino

```
// Definición de los pines donde están conectados los potenciómetros

const int input = A0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  float valor1 = analogRead(input);

  // Envía los valores a la computadora

  valor1 = (valor1 * 5)/1024;

  valor1 = ((valor1 * 100)/5) - 2;

  Serial.println(valor1);

  delay(1000);
}
```

Video

<https://www.youtube.com/shorts/mUURpExOtws>

Conclusiones

Méndez Pichardo Daniel

En el desarrollo de esta práctica se implementó un programa en Python, el cual nos permitió mostrar la temperatura en grados Celsius y la gráfica de los cambios de temperatura de un sensor PT100, antes acondicionado con un puente de Wheatstone y un AD620 en la práctica anterior. En el desarrollo de esta etapa se utilizó un Arduino el cual nos permito convertir el voltaje de salida (Vs) del amplificador AD620 en un voltaje digital, con la finalidad de integrarlo en un programa en Python para representarlo en una gráfica y de esta manera interpretar los cambios de subida o bajada del sensor.

Melo Jiménez Jesús Uriel

En esta práctica desarrollamos un programa en Python, este programa permite graficar en tiempo real datos de temperatura obtenidos desde un puerto serial, mostrando los resultados en una interfaz gráfica creada con Tkinter y Matplotlib. Fue sencillo de implementar gracias a las herramientas utilizadas, pero el verdadero reto estuvo en la parte del armado del circuito encargado de generar los datos. Al final logramos que el circuito que media la temperatura funcionara y el programa también funcionara correctamente.

Segura Gutiérrez Obed

En esta práctica aprendimos a integrar un sensor PT100 con un programa basado en Arduino y Python para medir y graficar temperaturas. Utilizamos un circuito previamente preparado con un puente de Wheatstone y un amplificador, digitalizando las señales para representarlas en tiempo real. Lo más desafiante fue sincronizar correctamente los datos entre el hardware y el programa, pero una vez solucionado, los resultados fueron satisfactorios. Este ejercicio no solo nos ayudó a comprender mejor el manejo de sensores, sino también a aplicar conceptos prácticos en la programación y análisis de señales.

Méndez Hernández Yael

En esta práctica integramos un sensor PT100 con un circuito previamente acondicionado con un puente de Wheatstone y un amplificador AD620. Usamos un Arduino para digitalizar las señales y un programa en Python para graficar los datos de temperatura en tiempo real. Aunque al principio fue complicado sincronizar el hardware con el software, logramos resolverlo y obtuvimos resultados satisfactorios. Esta experiencia nos ayudó a entender mejor cómo

trabajar con sensores, procesar señales y aplicar programación para el análisis de datos, lo cual fue muy útil para conectar la teoría con la práctica y reforzar nuestros conocimientos .

Alva Galicia Diego Sebastián

El circuito del sensor PT100 ya estaba preparado. La parte laboriosa fue programar la graficadora de temperatura, y el comunicador del puerto serial del Arduino. Otro reto fue estabilizar la salida de datos hacia la computadora, luego de varios intentos fallidos y pruebas de recalibración del circuito por el ruido, logramos que leyera los datos de temperatura a la pantalla del monitor.

Sello y Firma

