

REPORTE DE PRÁCTICA

SISTEMA DE CARACTERIZACIÓN DE PRESIÓN ATMÓSFERICA, TEMPERATURA Y ALTURA

BÁEZ LÓPEZ JOSÉ GERMÁN 202236436, COLULA ROQUE BRANDON 202118338, GUZMÁN MONTIEL EDGAR 202223419, LÓPEZ ESPINOZA CRISTÓBAL 202129499, MUÑOZ HERNÁNDEZ SAIRA 202245477, PÉREZ ZAMORA JOSÉ ENRIQUE 202248104,

BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

RESUMEN

En esta práctica, se implementó un sistema de adquisición de datos utilizando el sensor BMP280 para medir presión atmosférica, temperatura y altura. El sistema fue montado en un protoboard y controlado por el microcontrolador ESP32, el cual se programó para mostrar las mediciones en una pantalla OLED. También se diseñó un proceso para almacenar los datos durante 24 horas. Las mediciones se realizaron en un espacio específico, permitiendo analizar los cambios en las variables a lo largo del tiempo. Este proyecto combina conocimientos de electrónica y mecánica de fluidos, proporcionando una herramienta práctica para el monitoreo ambiental.

I. OBJETIVOS

- [1] Relacionar los conocimientos de electrónica con los principios de mecánica de fluidos aplicados en la medición de presión atmosférica, temperatura y altura.
- [2] Caracterizar un espacio específico para realizar mediciones de presión, altura y temperatura utilizando el sistema de adquisición de datos mediante microcontroladores.
- [3] Visualizar las lecturas de presión, altura y temperatura en tiempo real a través de un circuito electrónico funcional.
- [4] Analizar la variación de las tres variables medidas en función del entorno seleccionado, permitiendo identificar patrones o cambios significativos durante el periodo de monitoreo.

II. MARCO TEÓRICO

Presión atmosférica

La presión atmosférica es la fuerza ejercida por el aire sobre la superficie terrestre. Se mide en unidades como hectopascales (hPa) y es fundamental en meteorología para predecir cambios climáticos. La presión varía con la altitud: disminuye conforme se asciende, lo que se usa para calcular la altura.

Temperatura

La temperatura mide el nivel térmico de un entorno, normalmente en grados Celsius ($^{\circ}\text{C}$). Es uno de los parámetros más importantes en la caracterización de ambientes, ya que influye directamente en la presión y la densidad del aire.

La temperatura puede medirse en diferentes unidades. En los contextos científicos se suele utilizar la escala Kelvin, que se basa en un estado teórico conocido como "cero absolutos". En el cero absoluto, 0 K, todas las vibraciones de los átomos y las moléculas han cesado y todas las partículas se encuentran en sus estados de energía más bajos.

Altura

La altitud es la altura sobre el nivel del mar, y puede deducirse de la presión atmosférica utilizando ecuaciones barométricas. El sensor BME280 utiliza las variaciones en la presión para estimar la altura, lo cual es útil para aplicaciones en monitoreo ambiental y estudios de clima local.

Sensor BMP280

El sensor BMP280 es un dispositivo compacto que mide presión atmosférica y temperatura. A través de la interfaz I2C, se comunica con el microcontrolador para obtener y transmitir los valores medidos. Aunque este sensor puede medir humedad, en este caso solo se utilizaron las funciones de medición de presión y temperatura.

Microcontrolador

El ESP32 fue el microcontrolador utilizado para procesar los datos y controlar la pantalla OLED. Este dispositivo es adecuado para aplicaciones de monitoreo y cuenta con una buena capacidad de procesamiento y manejo de múltiples periféricos.

Pantalla OLED

Una pantalla OLED de 0.96 pulgadas fue empleada para mostrar en tiempo real los valores de presión, temperatura y altura. Estas pantallas ofrecen alta resolución y bajo consumo, lo que las hace ideales para proyectos de monitoreo.

III. MATERIALES

- Protoboard
- Sensor BMP280
- ESP 32
- Pantalla OLED

IV. MONTAJE EXPERIMENTAL

Armado del sistema de adquisición

Se realizó el montaje de los componentes en un protoboard, conectando el sensor BMP280, el microcontrolador ESP32 y la pantalla OLED. Las conexiones fueron:

1. Sensor BME280

- VCC: Se conecta al pin de 3.3V del ESP32 para alimentarlo.
- GND: Se conecta al pin de GND del ESP32 para establecer la tierra común.
- SCL (línea de reloj I2C): Se conecta al pin GPIO22 del ESP32.

- SDA (línea de datos I2C): Se conecta al pin GPIO21 del ESP32.

2. Pantalla OLED (0.96")

- VCC: Se conecta también al pin de 3.3V del ESP32 para alimentarla.
- GND: Se conecta al pin de GND del ESP32.
- SCL (línea de reloj I2C): Se conecta al mismo pin GPIO22 del ESP32 (compartido con el BME280).
- SDA (línea de datos I2C): Se conecta al mismo pin GPIO21 del ESP32 (compartido con el BME280).

Este esquema utiliza el protocolo I2C, lo que permite compartir los pines SCL y SDA entre el sensor y la pantalla, haciendo el circuito más eficiente y simple en términos de conexiones.

A continuación, se presentará un diagrama de conexiones y el circuito físico montado en el protoboard.

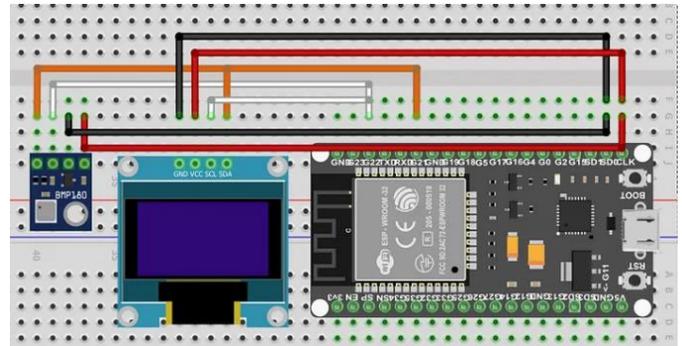


Figura 1: Diagrama de conexiones

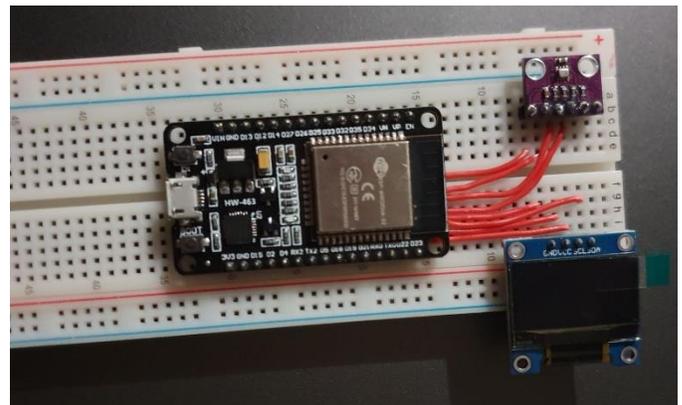


Figura 2: Circuito físico montado

Programación

A continuación, se describe el código, dividiendo en secciones clave para entender su funcionamiento. Este programa está diseñado para leer las mediciones del sensor BMP280 (temperatura, presión y altura) y mostrar estos valores en una pantalla OLED. Además, el sistema incluye un contador que se actualiza y reinicia cada vez que se toma una nueva medición.

1. Importación de librerías y declaración de constantes

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BMP280.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
```

En esta sección se incluyen las bibliotecas necesarias para la comunicación I2C, el manejo del sensor BMP280 y la gestión de la pantalla OLED

2. Definición de Constantes y Objetos

```
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64

#define OLED_RESET -1
Adafruit_SSD1306 display (SCREEN_WIDTH,
SCREEN_HEIGHT, &Wire, OLED_RESET);
Adafruit_BMP280 bmp; // Creación del objeto BMP280
```

Aquí se definen las dimensiones de la pantalla OLED y se inicializan los objetos `display` y `bmp` para la pantalla y el sensor, respectivamente. La correcta configuración de las dimensiones es crucial para asegurar que la visualización sea adecuada.

3. Variables de Tiempo y Mediciones

```
unsigned long ultimaMedicion = 0;
unsigned long intervaloMedicion = 5000; // Cada 5 segundos
unsigned long ultimaActualizacionContador = 0;
int contador = 5; // Contador de tiempo para la siguiente medición
```

```
float temperatura, presion, altura;
```

```
const float PRESION_NIVEL_DEL_MAR = 1013.25;
```

Se declaran variables para gestionar el tiempo entre mediciones y almacenar las lecturas del sensor. El contador se utiliza para controlar el tiempo hasta la siguiente medición. La constante `PRESION_NIVEL_DEL_MAR` es esencial para calcular la altitud a partir de la presión.

4. Configuración inicial (Setup)

```
void setup() {
  Serial.begin(115200); // Iniciar comunicación serial para PLX-DAQ

  // Inicializar la pantalla OLED
  if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) { // Dirección I2C de la OLED (0x3C es común)
    Serial.println(F("No se encontró la pantalla OLED"));
    while (1);
  }
  display.clearDisplay();
  display.display();

  // Inicializar el sensor BMP280
  if (!bmp.begin(0x76)) { // Dirección I2C del sensor BMP280 (puede ser 0x76 o 0x77)
    Serial.println("No se encontró el sensor BMP280");
    while (1);
  }

  // Mostrar mensaje de inicio
  display.setTextSize(1);
  display.setTextColor(SSD1306_WHITE);
  display.setCursor(0, 10);
  display.println("Inicializando...");
  display.display();
  delay(2000); // Esperar 2 segundos

  Serial.println("Sensor BMP280 detectado. Iniciando mediciones...");
}
```

Esta función se encarga de la configuración inicial del programa. Inicia la comunicación serial y verifica la correcta conexión de la pantalla y el sensor. Los mensajes de error son útiles para la depuración. La pausa de 2 segundos permite que el usuario vea el mensaje de inicialización.

5. Ciclo principal (loop)

```
void loop() {
  unsigned long tiempoActual = millis();

  // Actualizar el contador cada segundo
  if (tiempoActual - ultimaActualizacionContador >= 1000) {
    ultimaActualizacionContador = tiempoActual;

    // Actualizar el contador y mostrarlo en la pantalla
    if (contador > 0) {
      contador--;
    }

    display.clearDisplay();

    // Mostrar la temperatura
    display.setTextSize(1);
    display.setCursor(0, 0);
    display.println("Temp:");
    display.setCursor(50, 0);
    display.print(temperatura);
    display.println(" C");

    // Mostrar la presión
    display.setCursor(0, 15);
    display.println("Presion:");
    display.setCursor(50, 15);
    display.print(presion);
    display.println(" KPa");

    // Mostrar la altura
    display.setCursor(0, 30);
    display.println("Altura:");
    display.setCursor(50, 30);
    display.print(altura);
    display.println(" m");

    // Mostrar el contador en la parte inferior
    display.setTextSize(1); // Tamaño normal
    display.setCursor(0, SCREEN_HEIGHT - 10); // Ajustar la
    posición
    display.print("Contador:");
    display.setTextSize(0); // Tamaño más pequeño para el
    contador
    display.setCursor(100, SCREEN_HEIGHT - 10); // Ajustar
    la posición
    display.print(contador);
    display.print("s");
  }
}
```

```
display.display();

// Reiniciar el contador cuando llega a 0
if (contador == 0) {
  // Realizar una nueva medición
  temperatura = bmp.readTemperature();
  presion = bmp.readPressure() / 1000.0F; // Presión en hPa
  altura =
  bmp.readAltitude(PRESION_NIVEL_DEL_MAR); // Altitud
  en metros

  // Enviar las mediciones al Monitor Serie para PLX-DAQ
  Serial.print("DATA,DATE,TIME,");
  Serial.print(temperatura);
  Serial.print(",");
  Serial.print(presion);
  Serial.print(",");
  Serial.println(altura);

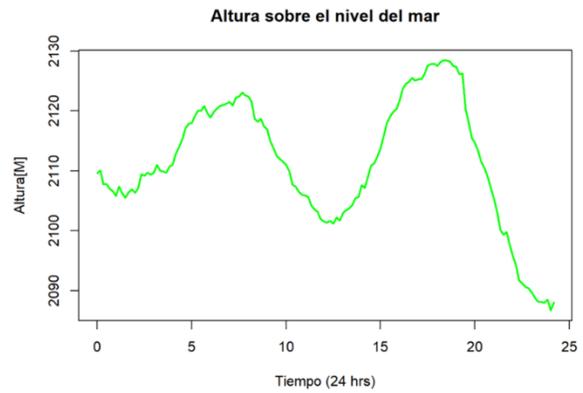
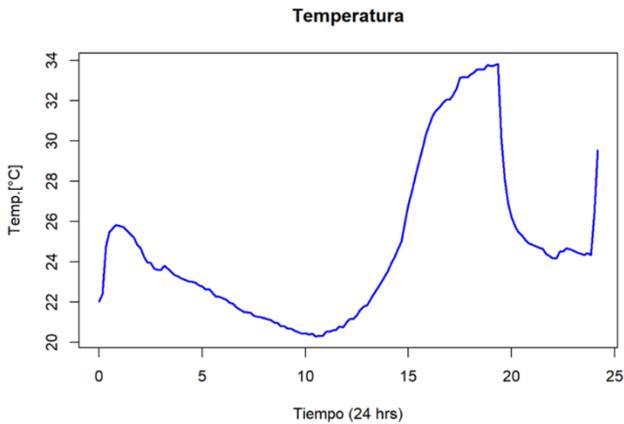
  // Reiniciar el contador
  contador = 600; // Reiniciar el contador a 600
}
}
```

V. RESULTADOS

Una vez armado el circuito con el sensor BMP280 y el ESP, y asegurado su correcto funcionamiento, se procedió a un monitoreo durante 24 horas. El objetivo fue recolectar una muestra lo suficientemente amplia para obtener parámetros representativos de temperatura, presión atmosférica y altitud. Las mediciones se realizaron cada 10 minutos, lo que permitió registrar variaciones detalladas en estos tres parámetros clave.

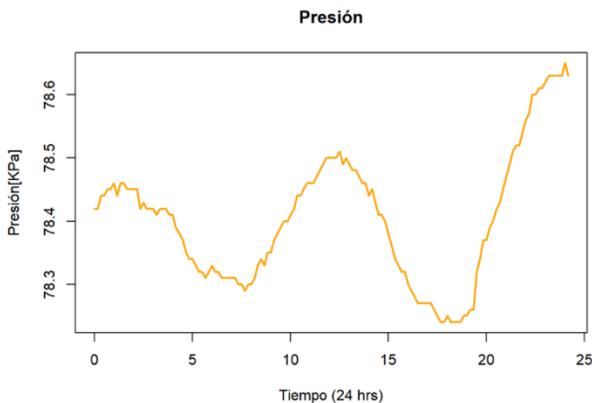
1. Temperatura

La temperatura varió entre 20°C y 32°C, mostrando un comportamiento fluctuante sin seguir necesariamente un ciclo diurno. Se observaron aumentos y caídas abruptas en la temperatura que podrían estar relacionados con cambios ambientales o eventos externos.



2. Presión

La presión osciló entre 78.3 kPa y 78.6 kPa, con fluctuaciones regulares que no necesariamente coinciden con un ciclo horario típico. Estas variaciones podrían estar influenciadas por condiciones meteorológicas locales o cambios en la masa de aire.



3. Altitud

La altitud fluctuó entre 2100 y 2130 metros, reflejando los cambios en la presión atmosférica. Aunque se observan patrones de subida y bajada, estos no parecen estar vinculados directamente a la hora del día, sino más bien a las variaciones de presión.

VI. ANÁLISIS DE RESULTADOS

Como podemos observar en las gráficas notamos cambios muy poco significativos en la presión ya que se debería de mantener constante, por otro lado, notamos que la altura se ve afectada de manera un poco más significativa que atribuimos a una variación en la precisión del sensor puesto que a lo largo del experimento y la medición la altura no tiene esta variación. Para finalizar podemos analizar la gráfica de temperatura que resulta tener un cambio más significativo que en nuestras mediciones asociamos a la caída de temperatura que hemos estado registrando a lo largo de estos días tanto por las lluvias, por el aire y por la noche que resulta en una variación más significativa.

VII. CONCLUSIONES

En este proyecto, se implementó exitosamente un sistema de adquisición de datos utilizando el sensor BMP280 y el microcontrolador ESP32 para caracterizar las variables de presión atmosférica, temperatura y altitud. Las mediciones realizadas durante 24 horas permitieron observar fluctuaciones representativas de estas variables en un entorno controlado, proporcionando información valiosa sobre los cambios ambientales.

Los datos de temperatura mostraron variaciones significativas, mientras que las fluctuaciones en la presión y la altitud se mantuvieron dentro de un rango predecible, aunque sin seguir un ciclo horario específico. Esto demuestra la sensibilidad del sistema para detectar cambios tanto naturales como influenciados por factores externos.

El sistema construido probó ser efectivo para este tipo de monitoreo ambiental, ofreciendo una herramienta práctica para la recolección de datos precisos y confiables. A partir de este experimento, se validan las capacidades del sensor BMP280 y del ESP32 como un conjunto robusto para futuras aplicaciones en monitoreo y análisis ambiental.

VIII. USO DE LA IA

Durante el desarrollo de este proyecto, utilizamos herramientas de inteligencia artificial (IA) para optimizar y corregir ciertas partes del código, lo que facilitó la detección de errores y la mejora del rendimiento del programa. Además, la IA fue un recurso clave para aclarar conceptos complejos de mecánica de fluidos, como la relación entre presión atmosférica y altitud, así como para obtener recomendaciones sobre la interpretación de datos y la presentación de resultados.

IX. REFERENCIAS BIBLIOGRÁFICAS

Agung, A. (2022, 7 enero). Project#6 ESP32 : Communication Protocol - Albertus Agung - Medium. Medium. <https://albrts-as.medium.com/project-6-esp32-communication-protocol-5640180e1b3d>

Bitwise Ar. (2018, 8 abril). *Arduino desde cero en Español - Capítulo 36 - BMP280 Presión atmosférica, Temperatura, Altitud* [Vídeo]. YouTube. https://www.youtube.com/watch?v=LtZ_b2WVMrU