ESP32 Simulation Practice & Temperature Humidity Sensor

Shelomita Putri H P
Fakultas Vokasi, Universitas Brawijaya
Email: shelomitaputrii@gmail.com

Abstract

This experiment aims to analyze the acquisition of temperature and humidity data using the DHT22 sensor connected to the ESP32 microcontroller, with development and simulation conducted via Wokwi PlatformIO in Visual Studio Code. The system initializes the DHT22 sensor and reads data every 2 seconds, displaying the measurements on the Serial Monitor. The experimental results indicate that proper GPIO configuration and sensor initialization are critical for obtaining accurate data and handling potential sensor read failures.

Keywords—Internet of Things, ESP32, DHT22, Temperature Sensor, Humidity Sensor, Wokwi PlatformIO, VSCode

1. Introduction

1.1 Background

The development of the Internet of Things (IoT) drives the integration of various sensors for real-time environmental monitoring. The DHT22 sensor is one of the popular sensors used to measure temperature and humidity. With the ESP32's capability for wireless connectivity and efficient data processing, their combination offers an economical and effective solution for environmental monitoring applications. The use of Wokwi PlatformIO in Visual Studio Code provides an integrated environment for development, simulation, and debugging, which accelerates the prototyping process of IoT applications.

1.2 Objectives

- Understand the operation of the ESP32 in reading and processing data from the DHT22 sensor.
- Implement periodic temperature and humidity data acquisition using Wokwi PlatformIO in Visual Studio Code.
- Identify and resolve potential issues, such as sensor read failures and error handling, within an integrated development environment.

2. Methodology

2.1 Tools & Materials

Hardware:

- ESP32 (simulated using Wokwi PlatformIO)
- DHT22 sensor
- Jumper wires

Software:

- Visual Studio Code with the PlatformIO extension
- Wokwi PlatformIO as the simulation environment

2.2 Implementation Steps

Project Creation:

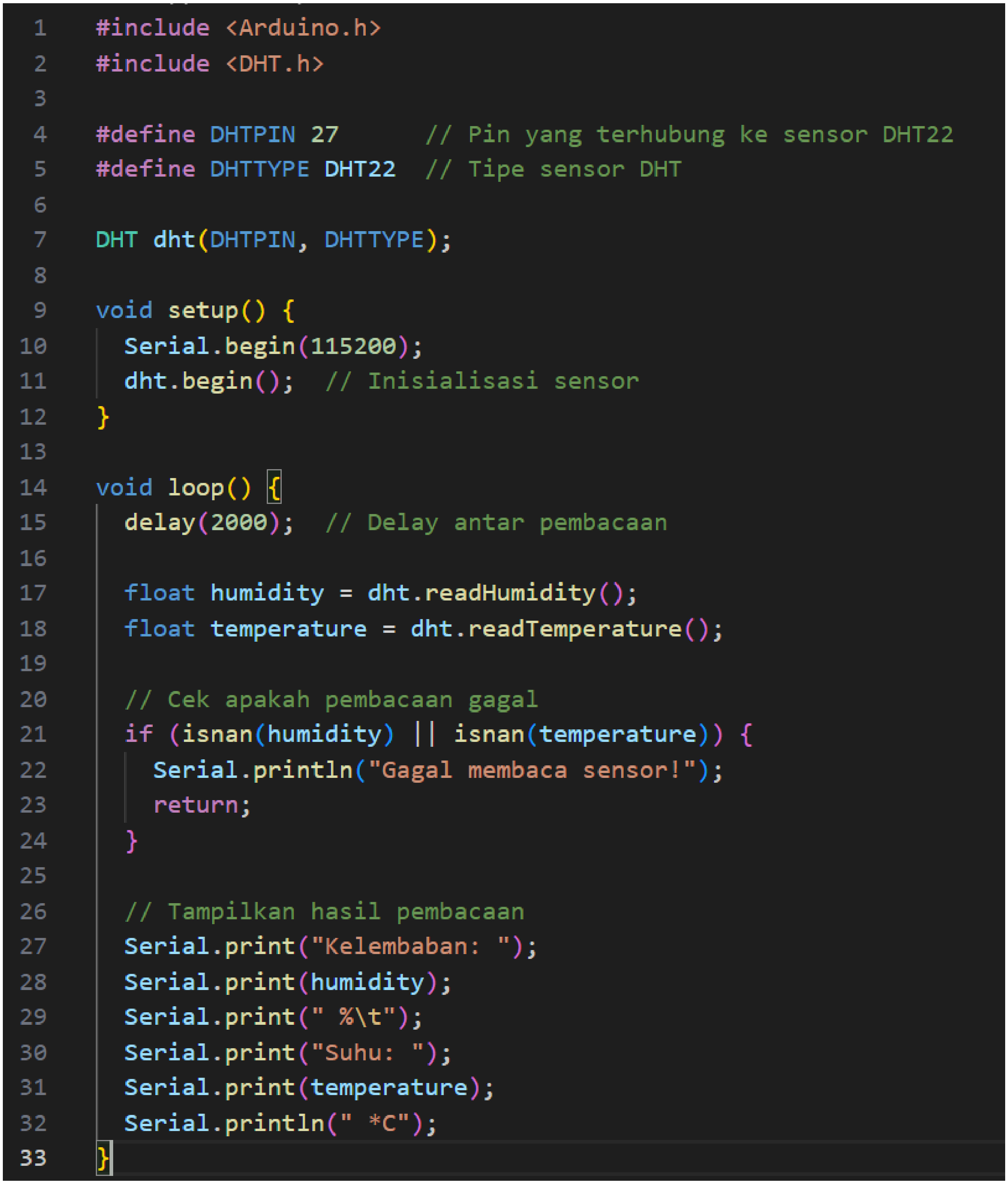
- Open Visual Studio Code and ensure that the PlatformIO extension is installed.
- Create a new PlatformIO project targeting the ESP32 board and integrate the simulation using Wokwi PlatformIO.

Physical Connection:

- Add the ESP32 and the DHT22 sensor components into the simulation schematic.
- Connect the DHT22 sensor to the ESP32, ensuring the sensor's data pin is connected to GPIO 27 as specified in the code.

Code Development:

- Copy and upload the following code into the project to initialize the sensor, read temperature and humidity data, and display it on the Serial Monitor.



```
#include <Arduino.h>
#include <DHT.h>

#define DHTPIN 27      // Pin yang terhubung ke sensor DHT22
#define DHTTYPE DHT22  // Tipe sensor DHT

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(115200);
  dht.begin();  // Inisialisasi sensor
}

void loop() {
  delay(2000);  // Delay antar pembacaan

  float humidity = dht.readHumidity();
  float temperature = dht.readTemperature();

  // Cek apakah pembacaan gagal
  if (isnan(humidity) || isnan(temperature)) {
    Serial.println("Gagal membaca sensor!");
    return;
  }

  // Tampilkan hasil pembacaan
  Serial.print("Kelembaban: ");
  Serial.print(humidity);
  Serial.print(" %\t");
  Serial.print("Suhu: ");
  Serial.print(temperature);
  Serial.println(" *C");
}
```

Simulation & Verification:

- Run the simulation using Wokwi PlatformIO in Visual Studio Code.
- Check the output on the Serial Monitor every 2 seconds. In case of a read failure, the system prints “Failed to read sensor!” as an error-handling measure.

3. Results and Discussion

3.1 Experimental Results

- **Data Output:**
Every 2 seconds, the DHT22 sensor reads and displays the humidity (in percent) and temperature (in degrees Celsius) on the Serial Monitor.
- **Error Handling:**
If a read failure occurs (for example, due to simulation issues or sensor initialization errors), the system prints the message "Failed to read sensor!" to alert the user.

3.2 Discussion

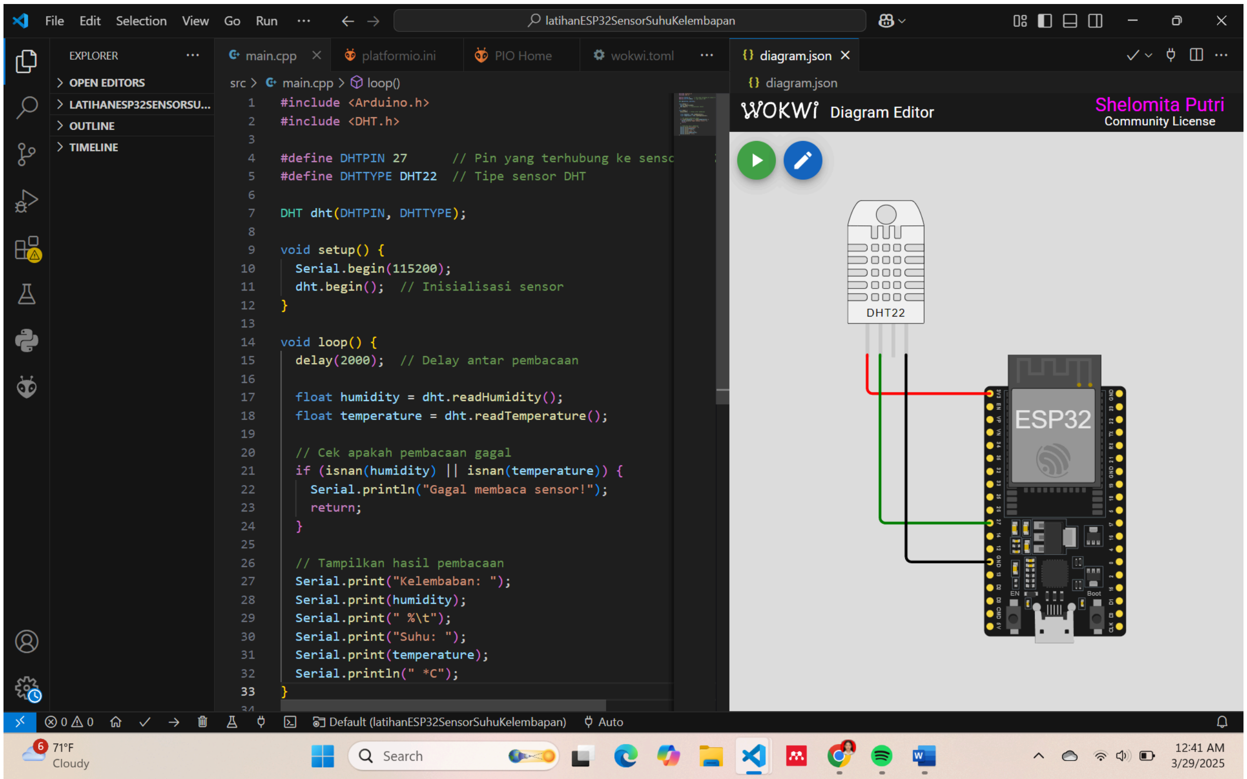
- **Successful Implementation:**
The simulation with Wokwi PlatformIO in VSCode successfully displays environmental data periodically, indicating that both the ESP32 and the DHT22 sensor have been correctly configured.

- **Potential Issues & Solutions:**

    - **Sensor Read Failure:** Ensure that the connection between the sensor and the ESP32 is correct and that the appropriate components are used.

    - **Reading Delay:** A 2000-millisecond delay provides sufficient time between reads; however, this can be adjusted based on real-world application needs.

    - **Data Validation:** Using the `isnan()` function helps detect invalid readings, allowing the system to promptly handle errors.



```cpp
#include <Arduino.h>
#include <DHT.h>

#define DHTPIN 27       // Pin yang terhubung ke sensc
#define DHTTYPE DHT22   // Tipe sensor DHT

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(115200);
  dht.begin();  // Inisialisasi sensor
}

void loop() {
  delay(2000);  // Delay antar pembacaan

  float humidity = dht.readHumidity();
  float temperature = dht.readTemperature();

  // Cek apakah pembacaan gagal
  if (isnan(humidity) || isnan(temperature)) {
    Serial.println("Gagal membaca sensor!");
    return;
  }

  // Tampilkan hasil pembacaan
  Serial.print("Kelembaban: ");
  Serial.print(humidity);
  Serial.print(" %\t");
  Serial.print("Suhu: ");
  Serial.print(temperature);
  Serial.println(" *C");
}
```