

# LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

## Latihan 2: Praktik Pembuatan Cek Suhu dan Kelembaban Menggunakan DHT22

**Author(s)** (Najwa Firdaus Azkiyah)  
Fakultas Vokasi, Universitas Brawijaya  
Email: najwazkiyah@student.ub.ac.id

### **Abstract** (Abstrak)

Proyek ini bertujuan untuk membuat sistem pengukur suhu dan kelembaban menggunakan sensor DHT22 dengan bantuan platform simulasi Wokwi. Sistem ini melibatkan mikrokontroler ESP32 yang terhubung ke sensor DHT22 untuk membaca data suhu dan kelembaban. Data yang diperoleh dari sensor kemudian diproses dan ditampilkan di terminal serial. Dalam proyek ini, dilakukan pemrograman mikrokontroler menggunakan Arduino IDE dan PlatformIO di VSCode, dengan penekanan pada konfigurasi GPIO dan penggunaan pustaka DHT. Penggunaan Wokwi sebagai alat simulasi memungkinkan pengujian dan pemantauan sistem secara virtual tanpa memerlukan perangkat keras fisik.

Keywords—Internet of Things, wokwi, ESP32, DHT22, arduino

### **1. Introduction** (Pendahuluan)

> (Font: Times New Roman, 10pt, Justified)

#### **1.1 Latar belakang**

Pemantauan suhu dan kelembaban merupakan aspek penting dalam berbagai aplikasi, mulai dari kontrol lingkungan dalam rumah kaca hingga manajemen suhu dan kelembaban di ruangan. Sensor DHT22 adalah alat yang populer untuk mengukur suhu dan kelembaban, karena memiliki presisi yang tinggi dan kemudahan penggunaan. Namun, pengembangan sistem pengukuran suhu dan kelembaban secara fisik dapat memerlukan waktu dan biaya yang tidak sedikit. Di sinilah platform simulasi Wokwi berperan, memungkinkan pengembangan dan pengujian sistem secara virtual tanpa memerlukan perangkat keras fisik.

## **1.2 Tujuan eksperimen**

Proyek ini bertujuan untuk membuat dan menguji sistem pemantauan suhu dan kelembaban menggunakan sensor DHT22 dengan bantuan platform Wokwi. Sistem ini akan menggunakan mikrokontroler ESP32 yang terhubung ke sensor DHT22 untuk membaca data suhu dan kelembaban. Data yang diperoleh dari sensor akan diproses dan ditampilkan di terminal serial. Proyek ini juga bertujuan untuk memberikan pemahaman praktis mengenai pemrograman mikrokontroler menggunakan Arduino IDE dan PlatformIO di VSCode, serta untuk mengatasi masalah teknis yang mungkin muncul selama pengembangan.

## **2. Methodology (Metodologi)**

### **2.1 Tools & Materials (Alat dan Bahan)**

> Mikrokontroler (ESP8266, Arduino, Raspberry Pi, dll.), sensor (DHT22, PIR, dsb.), software (Arduino IDE, MQTT Broker, dsb.)

### **2.2 Implementation Steps (Langkah Implementasi)**

Berikut adalah langkah-langkah implementasi singkat untuk membuat sistem cek suhu dan kelembaban menggunakan sensor DHT22 di platform Wokwi:

#### **1. \*\*Persiapan Alat dan Bahan:\*\***

- Mikrokontroler ESP32
- Sensor DHT22
- Kabel penghubung
- Platform simulasi Wokwi
- PlatformIO di VSCode

#### **2. \*\*Membuat Proyek di Wokwi:\*\***

- Buka situs Wokwi (<https://wokwi.com/>).
- Buat proyek baru dengan memilih mikrokontroler ESP32.
- Tambahkan sensor DHT22 ke dalam proyek.

#### **3. \*\*Koneksi dan Konfigurasi:\*\***

- Sambungkan pin data DHT22 ke salah satu pin GPIO ESP32 (misalnya GPIO4).

- Pastikan DHT22 terhubung ke VCC (3.3V atau 5V) dan GND.
- Jika tidak menggunakan modul dengan resistor pull-up bawaan, tambahkan resistor pull-up 10kΩ antara pin data dan VCC.

#### 4. **\*\*Pemrograman:\*\***

- Buka PlatformIO di VSCode.
- Tambahkan pustaka DHT dengan memasukkan `#include "DHT.h"`.
- Tentukan pin data dan tipe sensor:

Kode selengkapnya:

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

#define DHTPIN 25      // 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");
}
```

```
}
```

### 5. \*\*Simulasi dan Pengujian:\*\*

- Unggah kode ke ESP32 di Wokwi.
- Jalankan simulasi dan pantau output di terminal serial.
- Pastikan data suhu dan kelembaban terbaca dengan benar.

## 3. Results and Discussion (Hasil dan Pembahasan)

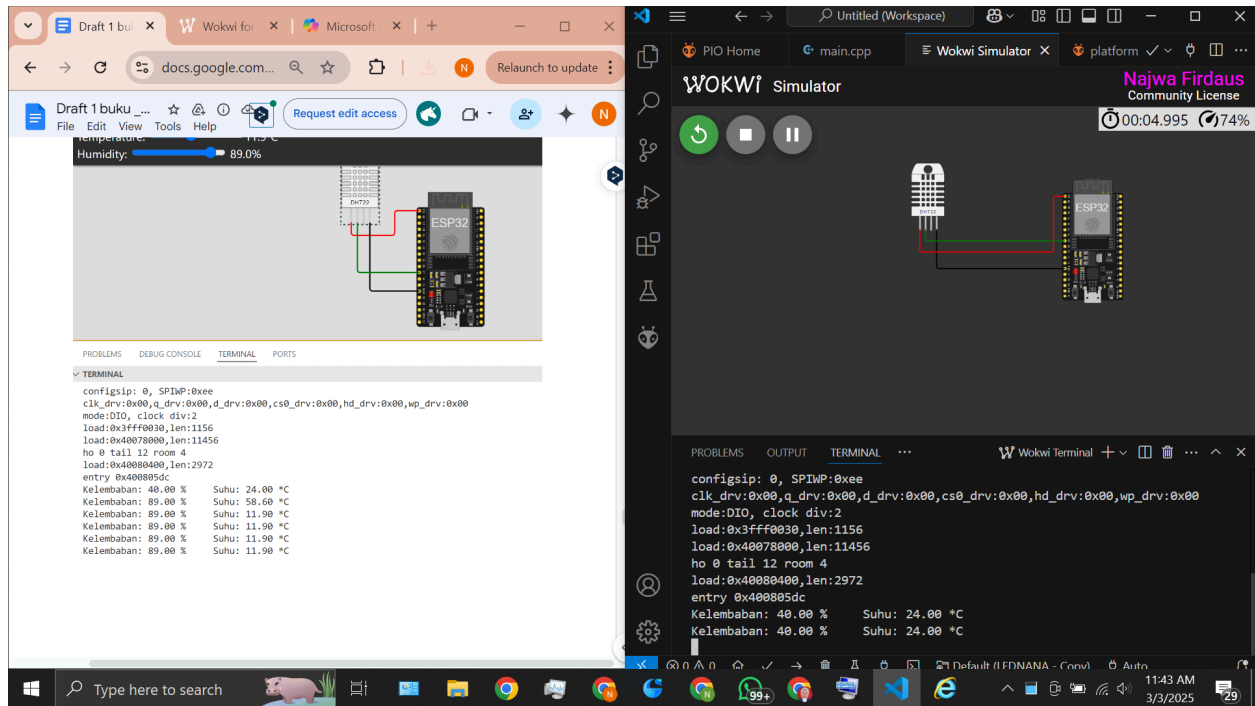
### 3.1 Experimental Results (Hasil Eksperimen)

The screenshot shows the Wokwi web IDE interface. On the left, the sketch.ino file contains the following code:

```
1
2 #include <Arduino.h>
3 #include "DHT.h"
4
5 #define DHTPIN 25
6 #define DHTTYPE DHT22
7
8 DHT dht(DHTPIN, DHTTYPE);
9
10 void setup() {
11   Serial.begin(9600);
12   Serial.println(F("DHTxx test!"));
13
14   dht.begin();
15 }
16
17
18 void loop() {
19   delay(500);
20
21   float h = dht.readHumidity();
22   // Read temperature as Celsius (the default)
23   float t = dht.readTemperature();
24   // Read temperature as Fahrenheit (isFahrenheit = true)
25   float f = dht.readTemperature(true);
26
27 }
```

On the right, the simulation window shows the DHT22 sensor connected to the ESP32. The serial terminal displays the following output:

Humidity	Temperature	Heat index
40.00%	24.00°C 75.20°F	23.50°C 74.30°F
40.00%	24.00°C 75.20°F	23.50°C 74.30°F
40.00%	24.00°C 75.20°F	23.50°C 74.30°F
40.00%	47.80°C 118.04°F	72.53°C 162.56°F
40.00%	47.80°C 118.04°F	72.53°C 162.56°F
40.00%	47.80°C 118.04°F	72.53°C 162.56°F
40.00%	47.80°C 118.04°F	72.53°C 162.56°F



#### 4. Appendix (Lampiran, jika diperlukan)

>sketch.ino / main.cpp

```

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

#define DHTPIN 25
#define DHTTYPE DHT22

DHT dht(DHTPIN, DHTTYPE);

void setup() {

  Serial.begin(9600);
  Serial.println(F("DHTxx test!"));

  dht.begin();
}

void loop() {

```

```

delay(500);

float h = dht.readHumidity();
// Read temperature as Celsius (the default)
float t = dht.readTemperature();
// Read temperature as Fahrenheit (isFahrenheit = true)
float f = dht.readTemperature(true);

// Check if any reads failed and exit early (to try again).
if (isnan(h) || isnan(t) || isnan(f)) {
    Serial.println(F("Failed to read from DHT sensor!"));
    return;
}

// Compute heat index in Fahrenheit (the default)
float hif = dht.computeHeatIndex(f, h);
// Compute heat index in Celsius (isFahreheit = false)
float hic = dht.computeHeatIndex(t, h, false);

Serial.print(F("Humidity: "));
Serial.print(h);

Serial.print(F("%  Temperature: "));
Serial.print(t);

Serial.print(F("°C "));
Serial.print(f);

Serial.print(F("°F  Heat index: "));
Serial.print(hic);

Serial.print(F("°C "));
Serial.print(hif);

Serial.println(F("°F"));
}

```

>diagram.json

```
{
  "version": 1,
  "author": "Najwa Firdaus",
  "editor": "wokwi",
  "parts": [
    { "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 9.6, "left": 158.44, "attrs": {} },
    { "type": "wokwi-dht22", "id": "dht1", "top": -18.9, "left": -101.4, "attrs": {} }
  ],
  "connections": [
    [ "esp:TX", "$serialMonitor:RX", "", [] ],
    [ "esp:RX", "$serialMonitor:TX", "", [] ],
    [ "dht1:GND", "esp:GND.1", "black", [ "v0" ] ],
    [ "dht1:SDA", "esp:25", "green", [ "v0" ] ],
    [ "dht1:VCC", "esp:3V3", "red", [ "v38.4", "h230.4", "v-96" ] ]
  ],
  "dependencies": {}
}
```

>libraries.txt

# Wokwi Library List

# See <https://docs.wokwi.com/guides/libraries>

# Automatically added based on includes:

DHT sensor library