**Monitoring dan Pengendalian Suhu serta Kelembapan Berbasis ESP32 dengan Protokol MQTT**

**oleh**

*Nuha Rona Zahra*

*Fakultas Vokasi, Universitas Brawijaya*

*Email: nuharonazz@gmail.com*

**Abstrak**

*Tujuan dari proyek ini adalah untuk merancang dan mengimplementasikan sistem pemantauan suhu dan kelembaban berbasis IoT. Dengan menggunakan mikrokontroler ESP32, sensor DHT22, dan protokol komunikasi MQTT, data suhu dan kelembapan dikirimkan secara berkala ke broker MQTT melalui koneksi Wi-Fi. Topik MQTT yang telah ditentukan digunakan untuk mengakses informasi ini. Selain itu, sistem memiliki lampu LED sebagai indikator visual. Lampu ini akan menyala secara otomatis apabila suhu lebih dari 30 derajat celcius atau kelembapan lebih dari 70 persen, menunjukkan kondisi lingkungan yang tidak normal.*

*Kata kunci:* *IoT, ESP32, DHT22, MQTT, LED.*

**Abstract (Bahasa Inggris)**

*The purpose of this project is to design and implement an IoT-based temperature and humidity monitoring system. By utilizing an ESP32 microcontroller, a DHT22 sensor, and the MQTT communication protocol, temperature and humidity data are periodically transmitted to an MQTT broker via a Wi-Fi connection. Predefined MQTT topics are used to access this information. Additionally, the system includes an LED as a visual indicator. The LED automatically lights up when the temperature exceeds 30 degrees Celsius or the humidity exceeds 70 percent, indicating abnormal environmental conditions.*

*Kata kunci:* *IoT, ESP32, DHT22, MQTT, LED.*

**Pendahuluan**

Internet of Things (IoT) telah menjadi teknologi yang berkembang pesat dan diterapkan dalam berbagai bidang, seperti industri, kesehatan, transportasi, dan sistem pemantauan. IoT memungkinkan perangkat elektronik saling terhubung dan bertukar data secara real-time melalui jaringan internet. Salah satunya adalah sistem monitoring suhu dan kelembaban yang berguna di berbagai aplikasi seperti pertanian, industry, dan sistem kontrol ruang.

Pada tugas kali ini, dirancang sebuah sistem pemantauan suhu dan kelembapan yang berbasis IoT menggunakan ESP32, sensor DHT22, dan protocol komunikasi MQTT. Data yang didapat dari sensor akan dikirimkan ke broker MQTT melalui jaringan Wi-Fi sehingga bisa mengakses dari berbagai perangkat yang terhubung.

Selain pengiriman data, sistem ini juga menggunakan indicator LED yang akan menyala dengan otomatis apabila suhu melebihi 30 C atau kelembapan melebihi 70%. Simulasi ini dilakukan di platform Wokwi yang memungkinkan pengujian secara virtual tanpa perangkat keras.

**Metodologi**

Metode yang digunakan dalam praktik ini mencakup beberapa tahapan utama, yaitu perancangan sistem, pembuatan rangkaian simulasi, pemrograman ESP32, dan pengujian sistem.

1. **Pemrograman Sistem**

perangkat yang digunakan adalah mikrokontroler ESP32, sensor DHT22, dan LED merah sebagai indikator. Komunikasi data dilakukan menggunakan protocol MQTT melalui koneksi Wi-Fi.

1. **Pembuatan Rangkaian Simulasi**

Rangkaian yang disusun di wokwi berupa

* + - pin data DHT22 terhubung ke pin GPIO 15
    - Anoda LED merah dihubungkan ke pin GPIO 2
    - Sensor DHT22 diberi suplai dari pin 3v3 ESP32 dan GND

1. **Pemrograman ESP32**
   * + Membaca data suhu dan kelembapan dari sensor DHT22
     + Mengirimkan data ke broker MQTT pada topik tertentu
     + Mengontrol status LED
     + Menerima pesan dari MQTT
2. **Pengujian Sistem**

pengujian dilakukan dengan menjalankan simulasi di wokwi.

**Hasil dan Pembahasan**

Pada praktik ini, berhasil membaca data suhu dan kelembaban secara berkala dan mengirimkannya ke broker MQTT dengan sukses.

LED menyala saat suhu melebihi 30 C atau kelembapan melebihi 70% sesuai dengan kondisi yang telah ditentukan di dalam program.

**Pembahasan**

Berdasarkan hasil proyek tersebut, sistem berhasil menjalankan fungsi-fungsi utamanya dengan baik. ESP32 mampu membaca data sensor DHT22 dan mengirimkan data tersebut ke MQTT.

Penggunaan LED sebagai indikator visual efektif untuk memberikan peringatan ke kondisi lingkungan yang tidak normal. Selain itu, sistem ini juga menggunakan komunikasi dua arah melalui MQTT di mana LED dapat di kontrol secara manual dengan mengirimkan pesan ke topik yang sama.

**Lampiran**

#include <WiFi.h>

#include <PubSubClient.h>

#include <DHTesp.h>

const int LED\_RED = 2;

const int DHT\_PIN = 15;

DHTesp dht;

const char\* ssid = "Wokwi-GUEST";

const char\* password = "";

const char\* mqtt\_server = "test.mosquitto.org";

WiFiClient espClient;

PubSubClient client(espClient);

unsigned long lastMsg = 0;

void setup\_wifi() {

delay(10);

Serial.println();

Serial.print("Connecting to ");

Serial.println(ssid);

WiFi.mode(WIFI\_STA);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

randomSeed(micros());

Serial.println("");

Serial.println("WiFi connected");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

}

void callback(char\* topic, byte\* payload, unsigned int length) {

Serial.print("Message arrived [");

Serial.print(topic);

Serial.print("] ");

for (int i = 0; i < length; i++) {

Serial.print((char)payload[i]);

}

Serial.println();

if ((char)payload[0] == '1') {

digitalWrite(LED\_RED, HIGH);

Serial.println("LED dinyalakan lewat MQTT");

} else {

digitalWrite(LED\_RED, LOW);

Serial.println("LED dimatikan lewat MQTT");

}

}

void reconnect() {

while (!client.connected()) {

Serial.print("Attempting MQTT connection...");

String clientId = "ESP32Client-";

clientId += String(random(0xffff), HEX);

if (client.connect(clientId.c\_str())) {

Serial.println("Connected");

client.publish("IOT/Test1/mqtt", "Test IOT");

client.subscribe("IOT/Test1/mqtt");

} else {

Serial.print("failed, rc=");

Serial.print(client.state());

Serial.println(" try again in 5 seconds");

delay(5000);

}

}

}

void setup() {

pinMode(LED\_RED, OUTPUT);

digitalWrite(LED\_RED, LOW); // pastikan mati di awal

Serial.begin(115200);

setup\_wifi();

client.setServer(mqtt\_server, 1883);

client.setCallback(callback);

dht.setup(DHT\_PIN, DHTesp::DHT22);

}

void loop() {

if (!client.connected()) {

reconnect();

}

client.loop();

unsigned long now = millis();

if (now - lastMsg > 2000) {

lastMsg = now;

TempAndHumidity data = dht.getTempAndHumidity();

float temp = data.temperature;

float hum = data.humidity;

Serial.print("Temperature: ");

Serial.println(temp);

Serial.print("Humidity: ");

Serial.println(hum);

String tempStr = String(temp, 2);

client.publish("IOT/Test1/temp", tempStr.c\_str());

String humStr = String(hum, 1);

client.publish("IOT/Test1/hum", humStr.c\_str());

if (isnan(temp) || isnan(hum)) {

Serial.println("Failed to read from DHT sensor!");

return;

}

// 🔥 Ganti ambang jadi lebih rendah agar LED pasti nyala

if (temp > 20.0 || hum > 30.0) {

digitalWrite(LED\_RED, HIGH);

Serial.println("LED ON karena suhu atau kelembaban tinggi (test)");

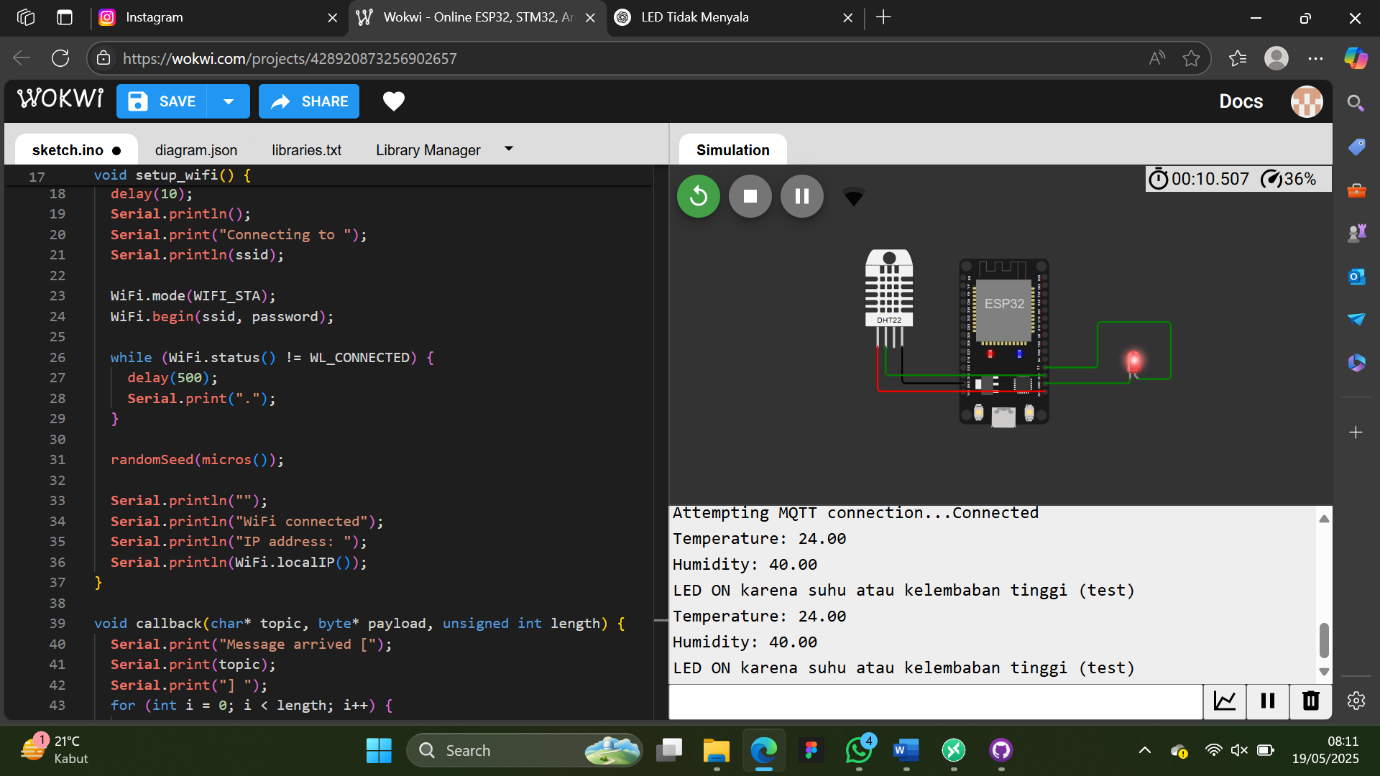
} else {

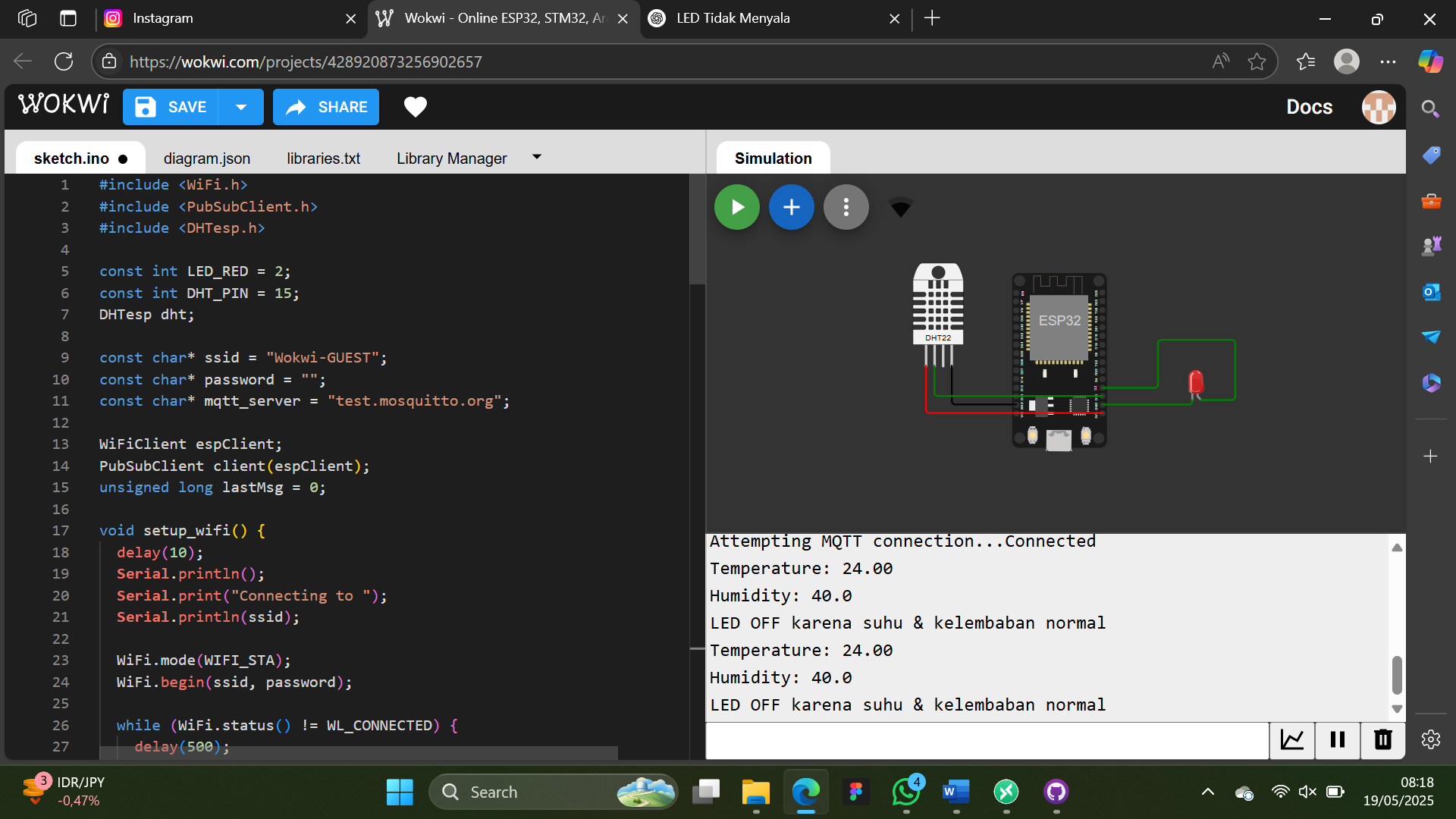
digitalWrite(LED\_RED, LOW);

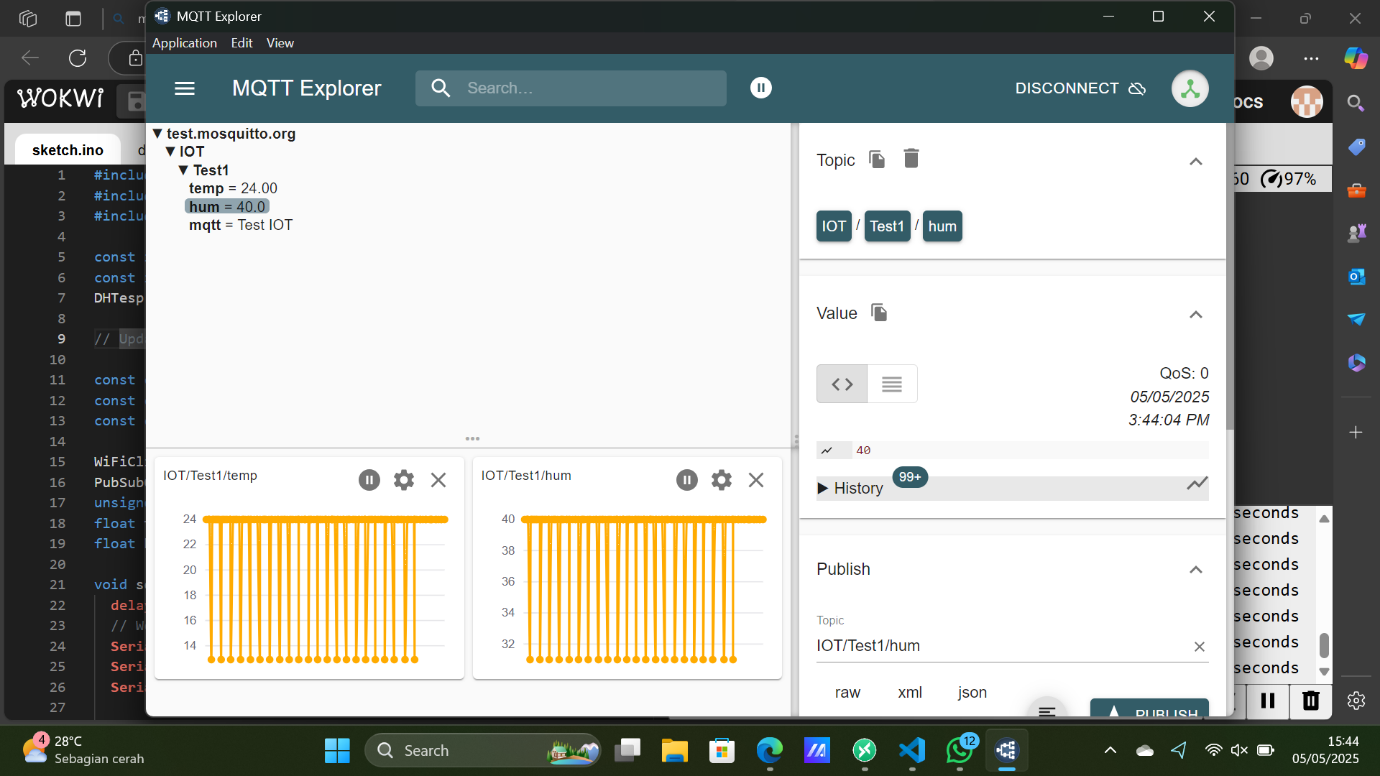
Serial.println("LED OFF karena suhu & kelembaban normal");

}

}

}****



****