

Praktik Pembuatan Rangkaian Pengukuran Suhu, Kelembapan, Dan Intensitas Cahaya

Khayru Rifaa Permana

Fakultas Vokasi , Universitas Brawijaya

Email : Khayrurifaa@student.ub.ac.id

Abstrak :

Perkembangan teknologi Internet of Things (IoT) telah memungkinkan pemantauan kondisi lingkungan secara real-time, terutama dalam mengukur parameter penting seperti suhu dan kelembapan. Penelitian ini bertujuan untuk mengembangkan dan menguji sistem pemantauan berbasis IoT dengan menggunakan sensor DHT22 yang dikendalikan oleh mikrokontroler ESP32. Data dari sensor dikirim ke platform Blynk melalui koneksi WiFi, sehingga dapat dipantau secara langsung melalui aplikasi mobile. Simulasi dilakukan menggunakan Wokwi sebagai emulator perangkat keras, sedangkan pengembangan perangkat lunak dilakukan melalui Visual Studio Code. Hasil eksperimen menunjukkan bahwa sistem ini dapat menampilkan data suhu dan kelembapan dengan pembaruan setiap dua detik, serta dapat diakses dari jarak jauh secara stabil. Keunggulan sistem ini meliputi kemudahan integrasi, aksesibilitas tinggi, serta kemampuan pemantauan real-time. Namun, terdapat tantangan seperti ketergantungan pada koneksi internet dan keterbatasan akurasi sensor DHT22. Untuk pengembangan lebih lanjut, sistem ini dapat ditingkatkan dengan menambahkan sensor intensitas cahaya dan sistem notifikasi otomatis melalui Blynk untuk memberikan peringatan ketika parameter lingkungan melebihi batas yang ditentukan. Dengan inovasi ini, sistem pemantauan dapat diterapkan dalam berbagai bidang seperti pertanian, industri, dan rumah pintar guna mendukung pengelolaan lingkungan yang lebih efisien.

Kata Kunci : *Arduino, ESP32, Wokwi, Blynk, Internet Of Things.*

Abstract:

The development of Internet of Things (IoT) technology has enabled real-time monitoring of environmental conditions, especially in measuring important parameters such as temperature and humidity. This study aims to develop and test an IoT-based monitoring system using a DHT22 sensor controlled by an ESP32 microcontroller. Data from the sensor is sent to the Blynk platform via a WiFi connection, so that it can be monitored directly via a mobile application. Simulations were performed using Wokwi as a hardware emulator, while software development was performed using Visual Studio Code. The experimental results show that this system can display temperature and humidity data with updates every two seconds, and can be accessed remotely in a stable manner. The advantages of this system include ease of integration, high accessibility, and real-time monitoring capabilities. However, there are challenges such as dependence on an internet connection and limited accuracy of the DHT22 sensor. For further development, this system can be improved by adding a light intensity sensor and an automatic notification system via Blynk to provide warnings when environmental parameters exceed specified limits. With this innovation, the monitoring system can be applied in various fields such as agriculture, industry, and smart homes to support more efficient environmental management.

Key Word : *Arduino, ESP32, Wokwi, Blynk, Internet Of Things.*

PENDAHULUAN

Teknologi Internet of Things (IoT) terus berkembang dan mempermudah pemantauan kondisi lingkungan secara real-time, terutama dalam mengukur parameter penting seperti suhu dan kelembapan. Data ini memiliki peran krusial di berbagai sektor, termasuk pertanian, industri, dan sistem rumah pintar. Dalam eksperimen ini, dilakukan simulasi pemantauan suhu dan kelembapan menggunakan sensor DHT22 yang terhubung dengan mikrokontroler ESP32. Sistem ini dikembangkan melalui platform Wokwi sebagai simulator perangkat keras, Blynk sebagai antarmuka pengguna, dan Visual Studio Code sebagai lingkungan pemrograman. Selain itu, isu lingkungan seperti polusi udara akibat meningkatnya jumlah kendaraan bermotor juga berpengaruh terhadap perubahan suhu dan

kelembapan. Oleh karena itu, diperlukan inovasi yang lebih efektif dalam memantau kondisi lingkungan agar dapat menjadi dasar perbaikan di masa depan. Eksperimen ini bertujuan untuk menguji kinerja sensor dalam membaca kondisi lingkungan dan menampilkan data secara real-time melalui platform IoT.

METODOLOGI

A. Alat dan Bahan

Mikrokontroler ESP32, Library Blynk, Sensor DHT, Kabel jumper, Breadboard, software Visual Studio Code, Website Blynk, PlatformIo, dan platform Wokwi .

B. Langkah Perancangan

1. Pilih ESP32 sebagai mikrokontroler pada situs [Wokwi](#).
2. Tambahkan komponen pendukung s Sensor DHT untuk menyusun rangkaian yang lengkap. Pastikan seluruh komponen tersambung dengan benar ke ESP32.
3. Buat akun pada website lynk .
4. Buat Projek baru, Klik new project.
5. Masukan Project Name lalu pilih device ESP32 pilih connection Type wifi lalu klik create. Setelah create akan mendapatkan Auth Token yang akan digunakan di program
6. Menambahkan widget switch, display LED, dan display Gauge.
7. Atur pin widget sesuai dengan program
8. Unduh dan pasang PlatformIO IDE serta Wokwi Simulator melalui ekstensi pada Visual Studio Code. Selanjutnya, buat proyek baru dengan memilih Board DOIT ESP32 DEVKIT V1 dan Framework Arduino sebagai basis pengembangannya.
9. Kompilasi proyek yang telah dibuat dengan menekan tombol Compile di bagian kiri bawah aplikasi untuk memastikan tidak ada kesalahan dalam kode sebelum menjalankan simulasi.
10. Buat file wokwi.toml dan isi file dengan code dibawah.

```
[wokwi]
version = 1
firmware = 'path-to-your-
firmware.hex'
elf = 'path-to-vour-firmware.elf'
```

11. Copy path firmware.bin dan firmware.elf lalu paste pada file wokwi.toml.

```
[wokwi]
version = 1
firmware = '.pio\build\esp32doit-devkit-v1\firmware.bin'
elf = '.pio\build\esp32doit-devkit-v1\firmware.elf'
```

12. Buat file diagram pada project dan copy script diagram.json di website.

```
{
  "version": 1,
  "author": "Khayru Permana",
  "editor": "wokwi",
  "parts": [
    { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -278.9,
"left": 52.76, "attrs": {} },
    {
      "type": "wokwi-led",
      "id": "led1",
      "top": -306.4,
      "left": -89.47,
      "attrs": { "color": "red" }
    }
  ]
}
```

```

    },
    {
      "type": "wokwi-resistor",
      "id": "r5",
      "top": -274.74,
      "left": -44.52,
      "attrs": { "value": "1000" }
    },
    {
      "type": "wokwi-dht22",
      "id": "dht1",
      "top": -260.42,
      "left": 247.56,
      "attrs": { "temperature": "58.7", "humidity": "77" }
    }
  ],
  "connections": [
    [ "esp:TX0", "$serialMonitor:RX", "", [ ] ],
    [ "esp:RX0", "$serialMonitor:TX", "", [ ] ],
    [ "led1:A", "r5:1", "red", [ "v0" ] ],
    [ "r5:2", "esp:D26", "red", [ "v1.2", "h17.93", "v81.46" ] ],
    [ "dht1:VCC", "esp:VIN", "red", [ "v87.6", "h-228.22", "v-54.65" ] ],
    [ "dht1:GND", "esp:GND.1", "black", [ "v93.06", "h-109.48", "v-76.5" ] ],
    [ "dht1:SDA", "esp:D15", "green", [ "v26.39", "h-81.44", "v-19.67" ] ],
    [ "led1:C", "esp:GND.2", "black", [ "v0" ] ]
  ],
  "dependencies": {}
}

```

13. Ubah nama file diagram menjadi diagram.json agar gambar rangkaian dapat ditampilkan dengan benar.
14. Buka file main.cpp di dalam folder src untuk menulis kode yang mengontrol sistem lampu lalu lintas. Atur agar lampu merah menyala selama 30 detik, lampu kuning selama 5 detik, dan lampu hijau selama 20 detik. Pastikan program berjalan sesuai dengan konfigurasi pada file diagram.json untuk mensimulasikan rangkaian secara optimal.

```

#define BLYNK_PRINT Serial

#define BLYNK_TEMPLATE_ID "TMPL6kP7WYFDt"
#define BLYNK_TEMPLATE_NAME "khayru"
#define BLYNK_AUTH_TOKEN "L08MmR7xoGNxWpnJZ0rHiJ609EzVCvje"

#include <WiFi.h>
#include <BlynkSimpleEsp32.h>
#include <DHTesp.h> // Library untuk DHT

```

```

char auth[] = BLYNK_AUTH_TOKEN; // Auth Token
char ssid[] = "Wokwi-GUEST"; // Nama hotspot yang digunakan
char pass[] = ""; // Password hotspot yang digunakan

const int DHT_PIN = 15;

int value0, value1, value2, value3, value6;

byte LED_R = 26;
byte LED_Y = 27;
byte LED_G = 14;
byte LED_B = 12;

DHTesp dht;
BlynkTimer timer;

void sendSensor() {
    TempAndHumidity data = dht.getTempAndHumidity();

    Serial.print("% Temperature: ");
    Serial.print(data.temperature);
    Serial.println("C ");
    Serial.print("% Kelembaban: ");
    Serial.print(data.humidity);
    Serial.println("% ");

    Blynk.virtualWrite(V0, data.temperature);
    Blynk.virtualWrite(V1, data.humidity);
}

BLYNK_WRITE(V2) {
    int nilaiBacaIO = param.asInt();
    digitalWrite(LED_R, nilaiBacaIO);
    Blynk.virtualWrite(V3, nilaiBacaIO);
}

void setup() {
    Serial.begin(115200);
    dht.setup(DHT_PIN, DHTesp::DHT22);

    pinMode(LED_R, OUTPUT);
    pinMode(LED_Y, OUTPUT);
    pinMode(LED_G, OUTPUT);
    pinMode(LED_B, OUTPUT);
}

```

```

    Blynk.begin(auth, ssid, pass);
    #ifdef WOKWI
        Blynk.connectWiFi(ssid, pass);
        Blynk.config(auth);
    #endif

    timer.setInterval(1000, sendSensor);
}

void loop() {
    Blynk.run();
    timer.run();
}

```

HASIL DAN PEMBAHASAN

Hasil dari eksperimen ini menunjukkan bahwa sistem monitoring suhu dan kelembapan berbasis IoT dengan ESP32, sensor DHT22, dan aplikasi Blynk dapat berfungsi dengan baik dalam menampilkan data secara real-time. Data suhu dan kelembapan yang diperoleh dari sensor berhasil dikirim ke server Blynk Cloud dan ditampilkan melalui antarmuka aplikasi Blynk menggunakan widget Value Display dan Gauge. Sistem ini bekerja dengan memperbarui data setiap dua detik, memungkinkan pengguna untuk memantau kondisi lingkungan dengan mudah melalui perangkat digital. Pengujian yang dilakukan membuktikan bahwa variasi suhu dan kelembapan yang terbaca sesuai dengan kondisi lingkungan, koneksi WiFi dapat berjalan stabil, dan data yang dikirim oleh ESP32 dapat ditampilkan dengan delay yang minimal pada aplikasi.

Dari hasil yang diperoleh, dapat disimpulkan bahwa teknologi IoT dalam pemantauan lingkungan memiliki keunggulan dalam aksesibilitas, kemudahan integrasi, dan kemampuan pemantauan secara real-time. Dengan adanya sistem ini, pengguna dapat mengakses informasi suhu dan kelembapan dari mana saja, selama perangkat terhubung dengan internet. Namun, terdapat beberapa tantangan, seperti ketergantungan pada koneksi WiFi yang dapat menghambat pengiriman data jika terjadi gangguan jaringan. Selain itu, sensor DHT22 memiliki keterbatasan dalam hal akurasi dibandingkan dengan sensor yang lebih canggih seperti SHT31 atau BME280.

Sebagai langkah pengembangan, sistem ini dapat ditingkatkan dengan menambahkan sensor cahaya (LDR) untuk mengukur intensitas cahaya di lingkungan sekitar, serta menggunakan baterai sebagai sumber daya cadangan agar sistem tetap berfungsi saat listrik padam. Selain itu, integrasi notifikasi melalui Blynk juga dapat menjadi fitur tambahan untuk memberikan peringatan jika suhu atau kelembapan berada di luar batas normal. Dengan adanya inovasi lebih lanjut, sistem ini dapat diterapkan dalam berbagai bidang seperti pertanian, industri, maupun sistem rumah pintar guna mendukung pemantauan lingkungan yang lebih efektif dan efisien.

Berikut adalah dokumentasi hasil pengujian praktikum :

