**PRAKTIKUM SIMULASI SISTEM MONITORING SUHU, KELEMBAPAN, DAN KONTROL LED BERBASIS ESP32 DENGAN PLATFORM BLYNK DI WOKWI**

**(IOT)**



Muhammad Fa’iz Ramadhan

Fakultas Vokasi, Universitas Brawijaya

[faizramadhan23@student.ub.ac.id](mailto:faizramadhan23@student.ub.ac.id)

**Abstrak**

Praktikum ini menerapkan sistem IoT menggunakan ESP32 sebagai mikrokontroler utama, yang berfungsi untuk membaca dan menampilkan data suhu dan kelembapan dari sensor DHT22 serta mengontrol LED melalui platform Blynk. Hasil praktikum, yang dilakukan menggunakan Wokwi, menunjukkan bahwa sistem dapat membaca dan menampilkan data suhu dan kelembapan pada aplikasi Blynk serta dapat mengontrol LED secara jarak jauh melalui platform Blynk. Keberhasilan komunikasi antara ESP32 dan platform Blynk bergantung pada penerapan Virtual Pin yang tepat.

**Kata kunci**: IoT, ESP32, DHT22, Blynk, LED Control, Virtual Pin

1. **Pendahuluan**
   1. **Latar belakang**

Perkembangan Internet of Things (IoT) telah membuka peluang baru dalam mengintegrasikan perangkat fisik dengan internet, memungkinkan pengguna untuk memantau dan mengontrol perangkat dari jarak jauh. Praktikum ini berfokus pada implementasi sistem IoT sederhana yang dapat membaca data lingkungan dan mengontrol aktuator melalui jaringan internet.

ESP32 dipilih sebagai mikrokontroler utama karena memiliki modul WiFi dan Bluetooth terintegrasi, menjadikannya pilihan yang ideal untuk aplikasi IoT. Platform Blynk digunakan sebagai antarmuka pengguna karena kemudahannya dalam membangun aplikasi IoT tanpa memerlukan keterampilan pemrograman web atau mobile yang mendalam.

* 1. **Tujuan eksperimen**

Tujuan dilakukannya praktikum ini adalah :

1. Memahami cara mengintegrasikan ESP32 dengan platform Blynk
2. Mengimplementasikan pembacaan sensor DHT22 dan pengiriman data ke cloud
3. Mengontrol LED secara jarak jauh melalui aplikasi mobile
4. Mempelajari konsep Virtual Pin dalam komunikasi IoT
5. Mengatasi masalah yang muncul dalam pengembangan sistem IoT
6. **Metodologi**

Praktikum ini mengadopsi pendekatan pengembangan prototipe iteratif dengan validasi melalui simulasi. Tahapan yang dilakukan meliputi:

* **Perancangan Sistem**: Mendefinisikan kebutuhan fungsional dan non-fungsional sistem monitoring.
* **Pemilihan Komponen**: Seleksi komponen hardware berdasarkan kebutuhan, kompatibilitas, dan ketersediaan.
* **Perancangan Sirkuit**: Merancang koneksi antara ESP32, sensor, dan display.
* **Pengembangan Firmware**: Menulis kode program untuk akuisisi data, pemrosesan, dan visualisasi.
* **Simulasi**: Implementasi sistem dalam simulator Wokwi untuk verifikasi fungsionalitas.
* **Analisis Performa**: Evaluasi kemampuan sistem dalam membaca dan menampilkan data.
  1. **Alat dan Bahan**

1. Mikrokontroler ESP32
2. LED virtual
3. Sensor DHT22
4. Resistor 220 ohm
5. Library WiFi.h untuk koneksi internet
6. Library BlynkSimpleEsp32.h untuk komunikasi dengan Blynk
7. Library DHTesp.h untuk membaca sensor DHT22
   1. **Langkah Implempentasi**
8. Konfigurasi Wokwi: Gunakan template yang sudah disediakan kemudian tambahkan DHT22 & LED, hubungkan DHT22 (data) ke pin 15 dan LED ke pin 26 melalui resistor.
9. Setup VSCode: Salin diagram.json dari Wokwi, buat wokwi.toml.
10. Konfigurasi Blynk yang pertama Buat akun & template
11. Tambahkan datastream: suhu (V4), kelembapan (V3), kontrol LED (V5), status LED (V6).
12. Buat dashboard: gauge (suhu & kelembapan), switch (kontrol LED), LED widget (status LED).
13. Dapatkan Blynk Token dan masukkan ke kode.
14. Implementasi Kode kita Gunakan WiFi.h, BlynkSimpleEsp32.h, dan DHTesp.h.
15. Pastikan fungsi sendSensor() untuk pembacaan suhu & kelembapan.
16. Pastikan fungsi BLYNK\_WRITE() untuk kontrol LED.
17. Mengamati hasil dan mencatat performa sistem selama siklus operasional berlangsung.
18. **Hasil dan Pembahasan**

Hasil simulasi monitoring suhu dan kelembaban Wokwi menunjukkan keberhasilan sistem dalam mengintegrasikan ESP32, sensor DHT22, dan Blynk.

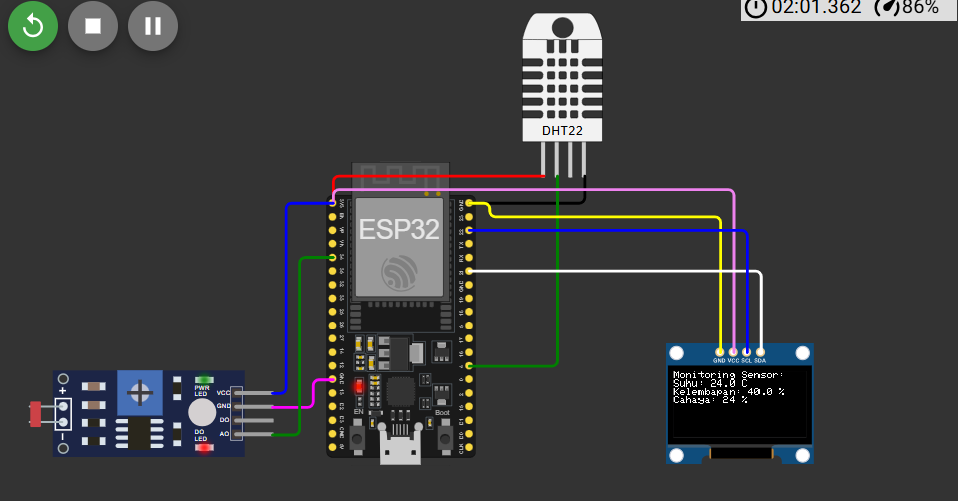
1. Koneksi WiFi Berhasil—ESP32 terhubung ke jaringan WiFi virtual Wokwi.

2. Pembacaan Sensor—DHT22 dapat mendeteksi suhu dan kelembaban dengan baik.  
3. Pengiriman Data ke Blynk—Data telah dikirim dan ditampilkan secara real-time di dashboard.

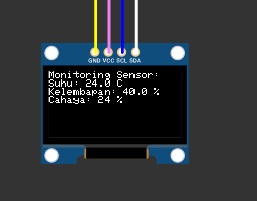
4. Kontrol LED Melalui Blynk—LED dapat dikontrol dari jarak jauh, yang menunjukkan komunikasi dua arah.

* 1. **Hasil Eksperimen**

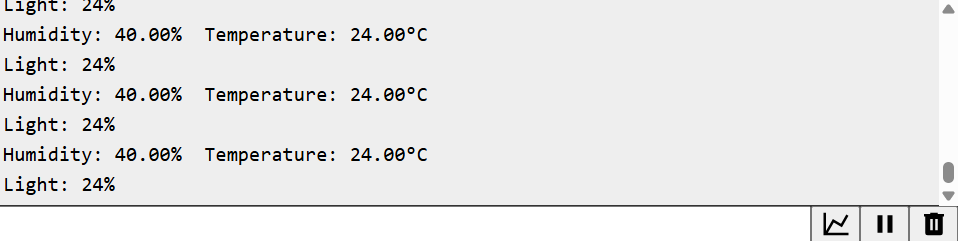
Rancangan komponen



Tampilan pada OLED



Data yang ditampilkan pada terminal



* 1. **Kode Program**

#define BLYNK\_DEVICE\_NAME "Esp32IoT"

#define BLYNK\_PRINT Serial

#define BLYNK\_AUTH\_TOKEN "Ev5T6Q2Ul9NdER7HqQ1Sv-9RYIJp7B-M"

#define BLYNK\_TEMPLATE\_ID "TMPL6G-x2Le-P"

#define BLYNK\_TEMPLATE\_NAME "IOT ESP RANGGA"

#include <Arduino.h>

#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;

//function untuk pengiriman sensor

void sendSensor()

{

TempAndHumidity data = dht.getTempAndHumidity();

//menampilkan temperature pada Serial monitor

Serial.print("% Temperature: ");

Serial.print(data.temperature);

Serial.println("C ");

Serial.print("% Kelembaban: ");

Serial.print(data.humidity);

Serial.println("% ");

Blynk.virtualWrite(V4, data.temperature); //mengirimkan data temperatur ke Virtual pin VO di Blynk Cloud

Blynk.virtualWrite(V3, data.humidity); //mengirimkan data kelemaban ke Virtual pin V1 di Blynk Cloud

}

BLYNK\_WRITE(V5)

{

int nilaiBacaIO =param.asInt();

digitalWrite(LED\_R, nilaiBacaIO);

Blynk.virtualWrite(V6, nilaiBacaIO);

}

void setup()

{

// Debug console

Serial.begin(115200); //serial monitor menggunakan bautrate 9600

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

pinMode(LED\_R, OUTPUT);

Blynk.begin(auth, ssid, pass); //memulai Blynk

timer.setInterval(1000, sendSensor); //Mengaktifkan timer untuk pengiriman data 1000ms

}

void loop()

{

Blynk.run(); //menjalankan blynk

timer.run(); //menjalankan timer