# LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

**Praktik Relay, Button & LED**



*Fawwaz Mufid Wardaya*

*Fakultas Vokasi, Universitas Brawijaya*

*Email : mahesfawwaz79@gmail.com*

**Abstract**

Tujuan dari eksperimen ini adalah untuk mengetahui bagaimana relay, tombol, dan LED yang terhubung ke mikrokontroler ESP32 berfungsi. Relay berfungsi sebagai saklar elektronik untuk mengontrol perangkat dengan daya tinggi, seperti lampu atau motor, sementara tombol digunakan sebagai input manual untuk mengaktifkan atau menonaktifkan relay dan LED. Memahami penggunaan pin GPIO pada ESP32 untuk mengontrol output (relay dan LED) dan membaca input (tombol) adalah tujuan utama praktikum ini. Praktik ini juga menunjukkan aplikasi Internet of Things (IoT) dalam sistem otomasi rumah pintar, seperti pengontrol perangkat elektronik atau pencahayaan.

**1. Introduction**

**1.1 Latar belakang**

Perangkat yang terhubung ke Internet of Things (IoT) dapat berkomunikasi dan bekerja secara otomatis tanpa campur tangan manusia. Otomasi rumah pintar adalah salah satu aplikasi IoT yang paling umum di mana sensor dan aktuator dapat digunakan untuk mengontrol perangkat seperti lampu, kipas, atau motor secara remote atau otomatis. Komponen relay sangat penting untuk sistem ini karena memungkinkan mikrokontroler seperti ESP32 mengontrol perangkat dengan kapasitas besar. LED dan tombol digunakan sebagai antarmuka sederhana untuk memberikan input manual dan menampilkan status sistem.

Kami menggunakan ESP32 sebagai mikrokontroler utama dalam praktikum ini untuk mengontrol relay dan LED berdasarkan tombol. Tujuan dari praktik ini adalah untuk mendapatkan pemahaman dasar tentang otomasi perangkat keras dalam konteks Internet of Things (IoT).

**1.2 Tujuan eksperimen**

Tujuan dari tes ini adalah:

1. Mempelajari cara kerja relay, tombol, dan LED serta integrasinya dengan ESP32.
2. Mengontrol perangkat berdaya tinggi menggunakan relay melalui ESP32.
3. Menunjukkan potensi aplikasi IoT dalam otomasi rumah pintar.

**2. Methodology (Metodologi)**

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

* Mikrokontroler : ESP32
* Relay Module
* LED
* Tombol (Button)
* Kabel Jumper
* Software : VSCODE

**2.2 Implementation Steps (Langkah Implementasi)**

1. **Persiapan Perangkat :**

* Hubungkan ESP32 ke komputer menggunakan kabel USB.
* Pin VCC DHT22 → 3.3V ESP32
* Pin GND DHT22 → GND ESP32
* Pin DATA DHT22 → Pin GPIO tertentu misalnya, GPIO 27
* Pin VCC Relay → 3.3V ESP32
* Pin GND Relay → GND ESP32
* Pin IN Relay → Pin GPIO tertentu, misalnya GPIO 27
* Pin Anoda LED (+) → Pin GPIO tertentu, misalnya GPIO 18
* Pin Katoda LED (-) → GND ESP32 melalui resistor 220Ω
* Pin salah satu kaki Tombol → Pin GPIO tertentu, misalnya GPIO 19
* Pin kaki lainnya Tombol → GND ESP32

1. **Code :**

* Buka VSCode dan instal library yang diperlukan (jika ada).
* Tulis kode program untuk:

Membaca input dari tombol.

Mengontrol LED berdasarkan status tombol.

Mengontrol relay berdasarkan status tombol.

1. **Pengujian :**

* Upload ke ESP32.
* Uji sistem dengan menekan tombol dan amati perubahan status LED serta relay.

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

**3.1 Experimental Results (Hasil Eksperimen)**

Setelah kode diupload ke ESP32, sistem berhasil berfungsi sesuai dengan yang diharapkan. Berikut hasil pengujian:

* Ketika tombol ditekan:

LED menyala.

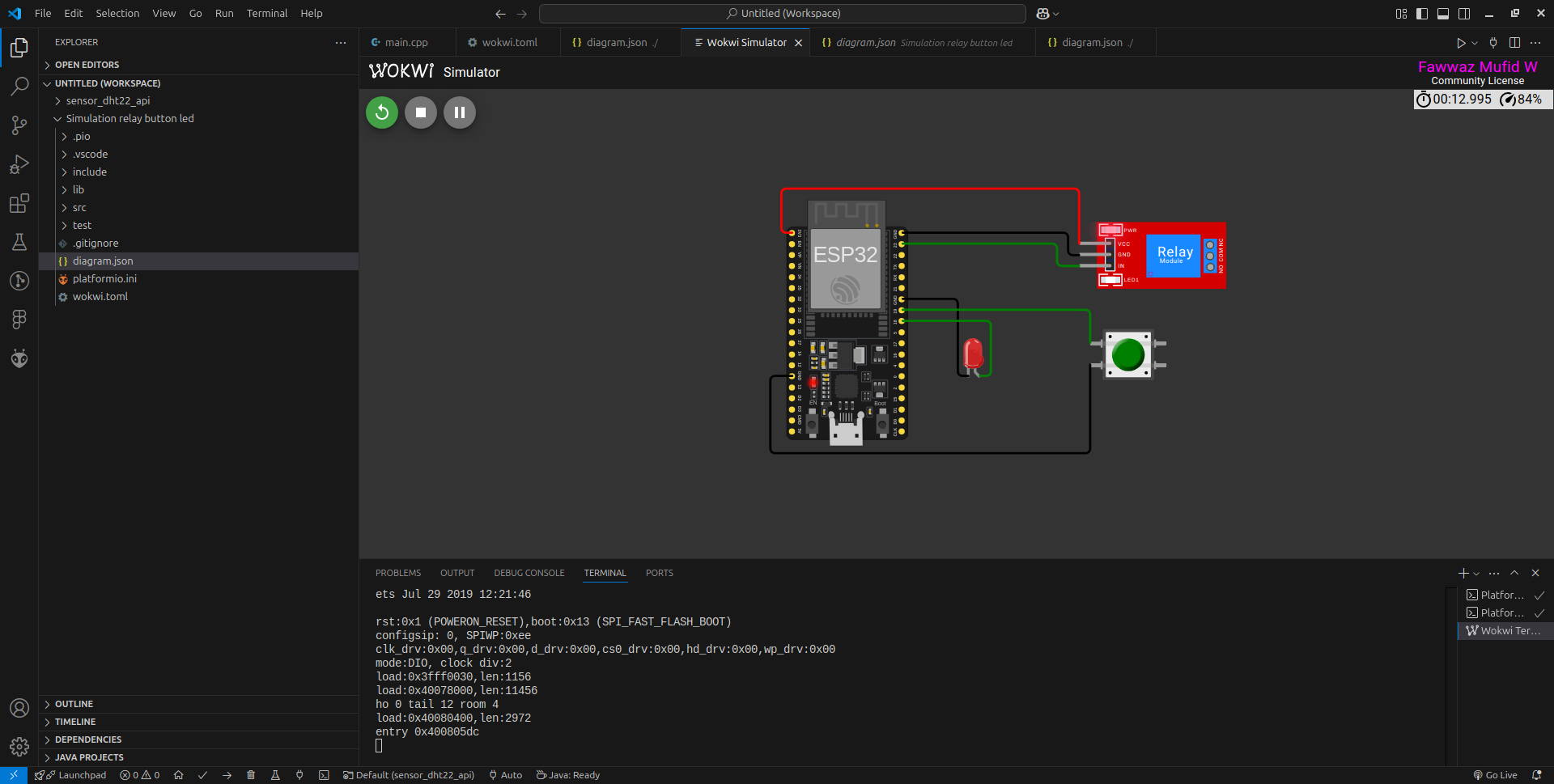
Relay aktif, menghasilkan suara "klik" dan menghidupkan perangkat eksternal (misalnya, lampu).

* Ketika tombol dilepas:

LED mati.

Relay nonaktif, mematikan perangkat eksternal.

Hasil ini menunjukkan bahwa relay, tombol, dan LED dapat bekerja dengan baik saat diintegrasikan dengan ESP32. Sistem ini dapat digunakan sebagai dasar untuk aplikasi IoT yang lebih kompleks, seperti otomasi rumah pintar atau kontrol perangkat jarak jauh.



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

**Kode Program :**

#include <Arduino.h>

// Define pin numbers

const int ButtonPin = 19; // GPIO19 connected to the pushbutton

const int LedPin = 18; // GPIO18 connected to the LED

const int RelayPin = 23; // GPIO23 connected to the relay module

void setup() {

// Set pin modes

pinMode(ButtonPin, INPUT\_PULLUP); // Set the button pin as an input with an internal pull-up resistor

pinMode(LedPin, OUTPUT); // Set the LED pin as an output

pinMode(RelayPin, OUTPUT); // Set the relay pin as an output

// Initialize the outputs to be OFF

digitalWrite(LedPin, LOW);

digitalWrite(RelayPin, LOW);

}

void loop() {

// Read the state of the button

int buttonState = digitalRead(ButtonPin);

// Check if the button is pressed

// Since the button is wired to pull the pin LOW when pressed, we check for LOW

if (buttonState == LOW) {

digitalWrite(LedPin, HIGH); // Turn on the LED

digitalWrite(RelayPin, HIGH); // Turn on the relay

} else {

digitalWrite(LedPin, LOW); // Turn off the LED

digitalWrite(RelayPin, LOW); // Turn off the relay

}

}