**LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)   
PRAKTIK SIMULASI RELAY, BUTTON & LED**

Fakultas Vokasi, Universitas Brawijaya

*Farhan Febrianto**Fakultas Vokasi, Universitas Brawijaya**Email: farhan05@student.ub.ac.id*

## Abstract (Abstrak)

Eksperimen ini bertujuan untuk mensimulasikan sistem kontrol menggunakan Relay Module, Tactile Switch Button, dan LED di platform Wokwi dengan pemrograman C++ melalui Visual Studio Code. Relay Module berfungsi sebagai pengendali utama yang mengoperasikan LED, sedangkan Tactile Switch Button digunakan sebagai perangkat input untuk memberikan perintah aktif atau non-aktif. Pada simulasi ini, program membaca status tombol dan mengirimkan sinyal ke Relay Module untuk mengontrol LED secara real-time. Penggunaan Wokwi memungkinkan pengujian sistem secara virtual tanpa memerlukan perangkat keras fisik, sehingga mempercepat proses debugging dan pengembangan. Hasil simulasi menunjukkan bahwa sistem berhasil menanggapi input tombol dengan mengaktifkan atau mematikan LED sesuai dengan perintah yang diberikan tanpa adanya kesalahan eksekusi. Kesimpulannya, praktikum ini membuktikan efektivitas integrasi Relay Module, Tactile Switch Button, dan LED dalam sistem kontrol digital, yang berpotensi untuk dikembangkan lebih lanjut dengan penambahan fitur seperti integrasi sensor atau konektivitas IoT untuk pengendalian jarak jauh.

**Kata kunci:** Relay Module, Tactile Switch Button, LED, Wokwi, Visual Studio Code, simulasi.

## 1. Pendahuluan

### 1.1 Latar belakang

Pengendalian perangkat elektronik, seperti lampu atau motor, merupakan aspek penting dalam otomasi rumah dan industri. Relay Module berperan sebagai saklar elektromagnetik yang memungkinkan pengendalian arus listrik secara efisien, sedangkan Tactile Switch Button menyediakan input manual untuk mengaktifkan atau menonaktifkan sistem. LED digunakan sebagai indikator status yang memberikan umpan balik visual terhadap keadaan sistem. Dengan adanya platform simulasi seperti Wokwi, pengembangan dan pengujian sistem kontrol ini dapat dilakukan secara virtual, sehingga mengurangi biaya dan risiko yang terkait dengan penggunaan perangkat keras fisik. Hal ini memungkinkan para pengembang untuk dengan cepat melakukan iterasi dan debugging sebelum implementasi nyata.

### 1.2 Tujuan eksperimen

1. Mempelajari cara menghubungkan dan mengkonfigurasi Relay Module, Tactile Switch Button, dan LED dengan mikrokontroler menggunakan pemrograman C++ di Visual Studio Code.
2. Mengimplementasikan logika kontrol untuk membaca input dari tombol dan mengendalikan relay serta LED sesuai perintah yang diberikan.
3. Mensimulasikan sistem kontrol berbasis relay, tombol, dan LED menggunakan platform Wokwi untuk memastikan kehandalan sistem secara virtual sebelum penerapan pada perangkat keras fisik.

## 2. Methodology (Metodologi)

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

Alat dan Perangkat Lunak:

1. **Wokwi Simulator** – Platform berbasis web untuk simulasi mikrokontroler dan rangkaian elektronik.
2. **Visual Studio Code** – Editor kode untuk menulis dan mengunggah program ke ESP32.
3. **Arduino Framework** – Digunakan sebagai dasar pemrograman mikrokontroler ESP32 dalam bahasa C++.

Bahan (Simulasi Komponen Elektronik):

1. **ESP32** – Mikrokontroler yang digunakan sebagai pengendali utama dalam simulasi.
2. **Relay Module** – Modul relay yang berfungsi untuk mengendalikan arus listrik dan mengoperasikan perangkat eksternal.
3. **Tactile Switch Button** – Tombol tekan sebagai input manual untuk mengaktifkan atau menonaktifkan sistem.
4. **LED** – Komponen indikator yang menampilkan status operasional sistem.
5. **Kabel Penghubung (dalam simulasi Wokwi)** – Menghubungkan komponen dalam desain rangkaian.

### 2.2 Implementation Steps (Langkah Implementasi)

1. **Perancangan Rangkaian di Wokwi**
2. Membuka platform Wokwi dan memilih ESP32 sebagai mikrokontroler.
3. Menambahkan komponen Relay Module, Tactile Switch Button, dan LED ke dalam proyek simulasi.
4. Menghubungkan Relay Module ke salah satu pin GPIO ESP32 (misalnya GPIO 5) untuk mengendalikan perangkat eksternal.
5. Menghubungkan Tactile Switch Button ke pin input ESP32 (misalnya GPIO 4) dan mengatur rangkaian pull-up atau pull-down sesuai kebutuhan.
6. Menghubungkan LED ke pin output ESP32 (misalnya GPIO 2) serta menyambungkan resistor pembatas arus jika diperlukan.
7. Menghubungkan semua komponen dengan kabel penghubung sesuai skema rangkaian yang dirancang, termasuk koneksi power (3.3V atau 5V) dan ground.
8. **Pemrograman Sistem Monitoring Suhu dan Kelembapan**
9. Membuka Visual Studio Code dan menulis kode dalam bahasa C++ menggunakan Arduino framework.
10. Mendefinisikan pin untuk Relay Module, Tactile Switch Button, dan LED.
11. Mengatur konfigurasi pin dengan menjadikan pin tombol sebagai input dan pin relay serta LED sebagai output dalam fungsi setup().
12. Menulis logika dalam fungsi loop() untuk membaca status tombol, kemudian mengaktifkan atau menonaktifkan Relay Module dan LED sesuai perintah input.
13. Menambahkan mekanisme debounce pada pembacaan tombol untuk menghindari pembacaan sinyal yang tidak stabil.
14. **Pengujian Simulasi**
15. Menjalankan program di Wokwi untuk mengamati respons sistem secara real-time.
16. Menguji apakah penekanan tombol berhasil mengubah status Relay Module dan LED sesuai dengan logika yang diprogram.
17. Memantau output simulasi dan melakukan debugging jika terdapat kesalahan atau respon yang tidak sesuai, serta melakukan perbaikan pada kode atau koneksi rangkaian jika diperlukan.

## 3. Hasil dan Pembahasan

### 3.1 Hasil Eksperimen

* 1. **Rangkaian Berhasil Dijalankan**

1. Rangkaian yang telah dirancang di Wokwi bekerja sesuai dengan desain, di mana Relay Module, Tactile Switch Button, dan LED berhasil terhubung dengan ESP32.
2. ESP32 mampu mengendalikan Relay Module dan LED berdasarkan input dari Tactile Switch Button.
   1. **Pengendalian Relay, Tombol, dan LED Berjalan dengan** **Baik**
   2. Sistem berhasil membaca status Tactile Switch Button secara real-time.
   3. Relay Module berfungsi dengan baik dalam mengaktifkan atau menonaktifkan perangkat eksternal sesuai perintah input.
   4. LED menampilkan status operasional sistem secara visual sesuai dengan logika kontrol yang diterapkan.
   5. **Hasil Simulasi di VsCode**
   6. Program yang dijalankan di Visual Studio Code menampilkan log interaksi input dan output secara jelas pada serial monitor.
   7. Tidak ditemukan error atau kesalahan dalam eksekusi kode.
   8. Serial monitor menampilkan pesan-pesan yang menandakan perubahan status sistem, seperti "Relay activated" atau "LED ON", sebagai indikasi bahwa sistem berjalan dengan baik.

## 4. Lampiran



