

Praktik Simulasi Relay, Button & LED

Ardan Pramudya
Fakultas Vokasi, Universitas Brawijaya
Email : ardanpramudya91@gmail.com

Abstract (Abstrak)

Praktikum ini bertujuan untuk merancang dan mensimulasikan sistem kendali relay menggunakan tombol dan LED berbasis ESP32. Relay digunakan untuk mengontrol perangkat listrik, sementara tombol digunakan sebagai input untuk mengaktifkan atau menonaktifkan relay. ESP32 berperan sebagai mikrokontroler utama yang mengolah sinyal dari tombol dan mengendalikan relay serta LED. Hasil eksperimen menunjukkan bahwa ESP32 dapat mengontrol relay dan LED dengan respons yang cepat dan akurat.

1.1 Latar Belakang

Penggunaan relay dalam sistem otomatisasi sangat penting untuk mengendalikan perangkat bertegangan tinggi menggunakan sinyal bertegangan rendah dari mikrokontroler. ESP32 dipilih karena kemampuannya yang luas dalam pemrosesan sinyal dan konektivitas IoT. Dalam praktikum ini, relay dikendalikan menggunakan tombol sebagai input manual, dan LED digunakan sebagai indikator status relay.

1.2 Tujuan Eksperimen

- Merancang dan mengimplementasikan sistem kendali relay menggunakan ESP32, tombol, dan LED.
- Menguji respon sistem terhadap input dari tombol untuk mengaktifkan atau menonaktifkan relay.
- Menampilkan status relay menggunakan LED sebagai indikator.

2. METHODOLOGY

2.1 Tools & Materials (Alat dan Bahan)

- ESP32 sebagai mikrokontroler utama.

- Modul relay sebagai aktuator.
- Push button sebagai input.
- LED sebagai indikator status relay.
- Breadboard dan kabel jumper sebagai media penyambung.
- Arduino IDE sebagai software pemrograman.

2.2 Implementation Steps (Langkah Implementasi)

1. Perancangan Rangkaian Elektronik

- Menghubungkan push button ke ESP32 dengan konfigurasi:
 - Satu kaki push button ke GPIO ESP32
 - Kaki lainnya ke ground (GND)
- Menghubungkan modul relay ke ESP32 dengan konfigurasi:
 - VCC → 5V ESP32
 - GND → GND ESP32
 - IN → GPIO ESP32
- Menghubungkan LED sebagai indikator relay:
 - Anoda (+) ke GPIO ESP32 melalui resistor 220Ω
 - Katoda (-) ke GND

2. Pemrograman di Arduino IDE

- Mengatur komunikasi serial untuk melihat status relay dan LED.
- Mengonfigurasi pin tombol sebagai input dan relay sebagai output.
- Menulis program untuk mendeteksi tombol ditekan dan mengubah status relay serta LED.
- Mengunggah program ke ESP32 dan menguji fungsinya.

3. Pengujian dan Evaluasi

- Menekan tombol dan mengamati apakah relay dan LED berubah status.
- Memastikan relay dapat mengontrol beban eksternal dengan baik.
- Mengevaluasi kecepatan respon sistem terhadap input dari tombol.

3.1 Hasil Eksperimen

- Sistem berhasil mengendalikan relay berdasarkan input dari tombol.
- LED dapat menunjukkan status relay dengan akurat.
- Respon relay terhadap tombol cukup cepat dan stabil.
- Tidak ada delay yang signifikan dalam aktivasi relay.

3.2 Pembahasan

Faktor yang Mempengaruhi Kinerja:

- **Jenis relay:** Relay dengan kualitas baik lebih responsif terhadap sinyal kontrol.
- **Stabilitas sinyal dari ESP32:** Penggunaan pull-up atau pull-down resistor pada tombol dapat meningkatkan kestabilan input.
- **Catu daya yang stabil:** Tegangan yang tidak stabil dapat menyebabkan relay tidak bekerja dengan baik.

4. APPENDIX (LAMPIRAN, JIKA DIPERLUKAN)

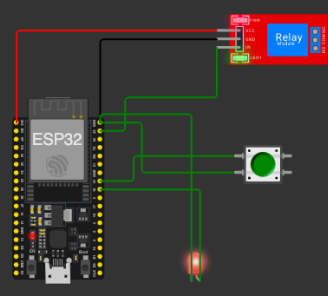
WOKWI SAVE SHARE Docs

sketch.ino • diagram.json • Library Manager

```
1 #include <Arduino.h>
2
3 // Define pin numbers
4 const int ButtonPin = 19; // GPIO19 connected to the pushbutton
5 const int LedPin = 18; // GPIO18 connected to the LED
6 const int RelayPin = 23; // GPIO23 connected to the relay module
7
8 void setup() {
9   // Set pin modes
10  pinMode(ButtonPin, INPUT_PULLUP); // Set the button pin as an input with an internal pull-up resistor
11  pinMode(LedPin, OUTPUT); // Set the LED pin as an output
12  pinMode(RelayPin, OUTPUT); // Set the relay pin as an output
13
14  // Initialize the outputs to be OFF
15  digitalWrite(LedPin, LOW);
16  digitalWrite(RelayPin, LOW);
17 }
18
19 void loop() {
20   // Read the state of the button
21   int buttonState = digitalRead(ButtonPin);
22
23   // Check if the button is pressed
24   // Since the button is wired to pull the pin LOW when pressed, we check for LOW
25   if (buttonState == LOW) {
26     digitalWrite(LedPin, HIGH); // Turn on the LED
27     digitalWrite(RelayPin, HIGH); // Turn on the relay
28   } else {
29     digitalWrite(LedPin, LOW); // Turn off the LED
30     digitalWrite(RelayPin, LOW); // Turn off the relay
31   }
32 }
33
```

Simulation

00:27.165 53%



clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0030, len:1156
load:0x40078000, len:11456
ho 0 tail 12 room 4
load:0x40080400, len:2972
entry 0x400805dc

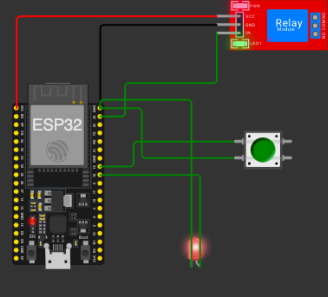
WOKWI SAVE SHARE Docs

sketch.ino • diagram.json • Library Manager

```
1 {
2   "version": 1,
3   "author": "Anonymous maker",
4   "editor": "wokwi",
5   "parts": [
6     { "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 48, "left": -148.76, "attrs": {} },
7     { "type": "wokwi-led", "id": "led1", "top": 217.2, "left": 42.2, "attrs": { "color": "green" } },
8     { "type": "wokwi-relay-module", "id": "relay1", "top": -47.8, "left": 86.4, "attrs": {} },
9     { "type": "wokwi-pushbutton", "id": "btn1", "top": 102.2, "left": 105.6, "attrs": { "color": "green", "xray": "1" } },
10  ],
11  "connections": [
12    [ "esp:TX", "$serialMonitor:RX", "", [] ],
13    [ "esp:RX", "$serialMonitor:TX", "", [] ],
14    [ "relay1:VCC", "esp:3V3", "red", [ "h0" ] ],
15    [ "relay1:GND", "esp:GND.2", "black", [ "h0" ] ],
16    [ "relay1:IN", "esp:23", "green", [ "v57.4", "h-105.6", "v38.4" ] ],
17    [ "btn1:2.L", "esp:GND.2", "green", [ "h-105.6", "v-57.4" ] ],
18    [ "btn1:1.L", "esp:19", "green", [ "h-115.2", "v28.8" ] ],
19    [ "led1:A", "esp:18", "green", [ "v0" ] ],
20    [ "led1:C", "esp:GND.2", "green", [ "v-192", "h-105.2" ] ],
21  ],
22  "dependencies": {}
23 }
```

Simulation

00:35.698 54%



clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0030, len:1156
load:0x40078000, len:11456
ho 0 tail 12 room 4
load:0x40080400, len:2972
entry 0x400805dc

