

Implementasi Sistem Lampu Lalu Lintas Menggunakan 3 Pushbutton Pada Wokwi Simulator ESP32 dengan PlatformIO di Visual Studio Code

Fransiska Natasya Desyanti

Program Studi Teknologi Informasi, Fakultas Vokasi Universitas Brawijaya

francenats@gmail.com

Abstract

This research aims to implement a traffic light system controlled by three pushbuttons using ESP32 on the Wokwi Simulator with the help of PlatformIO in Visual Studio Code. The system is designed to simulate traffic light behavior based on input from pushbuttons, where each button controls a different light pattern. The methods used include designing an electronic circuit schematic, writing program code, and testing system functionality through simulation. The results show that the system can respond to pushbutton input by controlling the red, yellow, and green lights according to the programmed logic. This simulation proves that ESP32 can be used as an effective microcontroller for controlling simple traffic light systems. This research is expected to serve as a foundation for the development of more complex traffic control systems in the future.

Keywords: *Traffic Light System, ESP32, Wokwi Simulator, PlatformIO, Visual Studio Code, Pushbutton.*

Abstrak

Penelitian ini bertujuan untuk mengimplementasikan sistem lampu lalu lintas yang dikendalikan oleh tiga pushbutton menggunakan ESP32 pada Wokwi Simulator dengan bantuan PlatformIO di Visual Studio Code. Sistem ini dirancang untuk mensimulasikan perilaku lampu lalu lintas berdasarkan input dari pushbutton, di mana setiap tombol mengontrol pola nyala lampu yang berbeda. Metode yang digunakan meliputi perancangan skema rangkaian elektronik, penulisan kode program, dan pengujian fungsionalitas sistem melalui simulasi. Hasil penelitian menunjukkan bahwa sistem mampu merespons input dari pushbutton dengan mengatur nyala lampu merah, kuning, dan hijau sesuai logika yang telah diprogram. Simulasi ini membuktikan bahwa ESP32 dapat digunakan sebagai mikrokontroler yang efektif untuk mengontrol sistem lampu lalu lintas sederhana. Penelitian ini diharapkan dapat menjadi dasar untuk pengembangan sistem kontrol lalu lintas yang lebih kompleks di masa depan.

Kata Kunci: Sistem Lampu Lalu Lintas, ESP32, Wokwi Simulator, PlatformIO, Visual Studio Code, Pushbutton.

1. PENDAHULUAN

1.1. LATAR BELAKANG

Latar belakang penelitian ini berawal dari pentingnya mengembangkan sistem lampu lalu lintas yang efisien, adaptif, dan mudah dikontrol, terutama dalam konteks pembelajaran dan pengembangan prototipe. Sistem lampu lalu lintas konvensional seringkali bersifat statis dan kurang fleksibel dalam menangani situasi lalu lintas yang dinamis. Dengan kemajuan teknologi mikrokontroler seperti ESP32, yang menawarkan kemampuan komputasi tinggi dan konektivitas yang luas, pengembangan sistem lampu lalu lintas yang lebih cerdas dan interaktif menjadi semakin mungkin. Penelitian ini memanfaatkan Wokwi Simulator sebagai platform simulasi yang memungkinkan pengujian dan visualisasi sistem lampu lalu lintas secara virtual sebelum diimplementasikan secara fisik. Dalam simulasi ini, tiga pushbutton digunakan untuk mengontrol perubahan sinyal lampu lalu lintas, memberikan fleksibilitas bagi pengguna untuk mengatur urutan dan durasi sinyal sesuai kebutuhan. PlatformIO di Visual Studio Code dipilih sebagai lingkungan pengembangan terintegrasi (IDE) yang mendukung penulisan kode, debugging, dan pengujian secara efisien. Penggunaan PlatformIO memungkinkan pengelolaan library dan dependencies yang lebih terstruktur, sehingga memudahkan proses pengembangan. Melalui implementasi ini, penelitian ini tidak hanya bertujuan untuk menciptakan prototipe sistem lampu lalu lintas yang fungsional, tetapi juga untuk memberikan pemahaman mendalam tentang integrasi antara hardware dan software dalam sistem embedded. Hasil penelitian ini diharapkan dapat menjadi dasar untuk pengembangan sistem kontrol lalu lintas yang lebih canggih dan dapat diaplikasikan dalam skala yang lebih besar di masa depan.

1.2. TUJUAN PRAKTIKUM

Tujuan dari praktikum ini adalah untuk mengimplementasikan dan menguji sistem lampu lalu lintas yang dikendalikan oleh tiga pushbutton menggunakan ESP32 pada Wokwi Simulator dengan bantuan PlatformIO di Visual Studio Code. Praktikum ini bertujuan untuk memahami prinsip kerja sistem kontrol lampu lalu lintas, mulai dari pembuatan rangkaian elektronik, penulisan kode program, hingga simulasi dan pengujian fungsionalitas sistem. Selain itu, praktikum ini juga bertujuan untuk mengembangkan kemampuan dalam menggunakan tools seperti Wokwi Simulator dan PlatformIO, serta meningkatkan pemahaman tentang integrasi antara hardware dan software dalam sistem embedded. Melalui praktikum ini, diharapkan peserta dapat memperoleh pengalaman langsung dalam merancang dan mengimplementasikan sistem kontrol sederhana yang dapat diaplikasikan dalam berbagai konteks pengembangan teknologi berbasis mikrokontroler.

2. METODOLOGI

2.1. ALAT DAN BAHAN

a. Wokwi Simulator

Platform simulasi online yang memungkinkan pengguna untuk merancang dan menguji sirkuit elektronik secara virtual. Wokwi digunakan untuk membuat sirkuit lampu lalu lintas dengan ESP32 sebagai mikrokontroler utama.

b. Visual Studio Code (VSCode)

Editor kode yang digunakan untuk menulis, mengelola, dan mengembangkan program. VSCode dipilih karena fleksibilitas dan dukungannya yang luas terhadap berbagai bahasa pemrograman.

c. PlatformIO

Ekstensi di VSCode yang menyediakan lingkungan pengembangan terintegrasi untuk pemrograman mikrokontroler, termasuk ESP32. PlatformIO memudahkan proses kompilasi, upload, dan debugging kode.

d. Library ESP32

Beberapa library seperti WiFi.h digunakan untuk mengimplementasikan fitur IoT, seperti konektivitas Wi-Fi dan kontrol jarak jauh.

e. ESP32

Mikrokontroler virtual yang berfungsi sebagai otak dari sistem lampu lalu lintas. ESP32 dipilih karena kemampuannya dalam menghubungkan perangkat ke jaringan IoT.

f. LED

Digunakan sebagai simulasi lampu lalu lintas, terdiri dari LED merah, kuning, dan hijau. Setiap LED mewakili lampu pada sistem lalu lintas. Digunakan sebagai simulasi lampu lalu lintas, terdiri dari LED merah, kuning, dan hijau. Setiap LED mewakili lampu pada sistem lalu lintas.

g. Pushbutton

Pushbutton pada Wokwi Simulator berfungsi sebagai input interaktif untuk mengontrol sistem elektronik. Dalam simulasi lampu lalu lintas, pushbutton mengirim sinyal ke ESP32 untuk mengubah status lampu sesuai logika program. Ini memungkinkan pengujian interaktif dan pemahaman tentang integrasi hardware-software dalam lingkungan simulasi yang aman.

h. Resistor

Komponen virtual yang digunakan untuk membatasi arus yang mengalir ke LED, mencegah kerusakan akibat arus berlebih.

i. Kabel Virtual

Digunakan untuk menghubungkan ESP32 dengan LED dan resistor dalam sirkuit virtual.

j. Koneksi Internet

Diperlukan untuk mengakses Wokwi Simulator dan menguji fitur IoT seperti kontrol jarak jauh.

2.2. LANGKAH IMPLEMENTASI

a. Perancangan Skema Rangkaian pada Wokwi Simulator

Langkah pertama adalah merancang skema rangkaian elektronik menggunakan Wokwi Simulator. Pada tahap ini, ESP32 dipilih sebagai mikrokontroler utama karena kemampuannya yang handal dan fleksibel. Tiga pushbutton digunakan sebagai input untuk mengontrol lampu lalu lintas, sementara LED merah, kuning, dan hijau digunakan sebagai output untuk mensimulasikan lampu lalu lintas. Resistor pull-down dipasang pada pushbutton untuk memastikan input stabil, dan resistor current-limiting dipasang pada LED untuk mencegah kerusakan akibat arus berlebih. Koneksi

- antara komponen dan pin ESP32 dirancang sesuai dengan kebutuhan sistem.
- b. **Penulisan Kode Program di PlatformIO (Visual Studio Code)**
Setelah skema rangkaian selesai, langkah selanjutnya adalah menulis kode program menggunakan PlatformIO di Visual Studio Code. Kode program ditulis dalam bahasa C/C++ dan berisi logika kontrol untuk mengatur lampu lalu lintas berdasarkan input dari pushbutton. Misalnya, Pushbutton 1 dapat diprogram untuk mengaktifkan lampu hijau, Pushbutton 2 untuk lampu kuning, dan Pushbutton 3 untuk lampu merah. Durasi nyala lampu dan transisi antar-lampu juga diatur dalam kode program untuk meniru perilaku lampu lalu lintas nyata.
 - c. **Pengujian Sistem pada Wokwi Simulator**
Setelah kode program selesai, langkah berikutnya adalah mengunggah kode ke ESP32 dalam simulasi Wokwi. Pengujian dilakukan untuk memverifikasi fungsionalitas sistem. Pada tahap ini, setiap pushbutton ditekan untuk memastikan lampu menyala sesuai dengan logika yang telah diprogram. Respons sistem terhadap input pushbutton, termasuk perubahan urutan dan durasi lampu, diamati dan dicatat. Jika ditemukan masalah, kode program dan skema rangkaian diperbaiki sesuai kebutuhan.
 - d. **Analisis Hasil Pengujian**
Setelah pengujian selesai, hasilnya dianalisis untuk memastikan sistem bekerja sesuai dengan desain awal. Data yang dikumpulkan meliputi respons sistem terhadap input pushbutton, konsistensi perubahan lampu, dan durasi nyala lampu. Jika terdapat ketidaksesuaian atau kesalahan, langkah-langkah perbaikan dilakukan, seperti memperbaiki kode program atau menyesuaikan skema rangkaian. Analisis ini penting untuk memastikan sistem lampu lalu lintas berfungsi dengan baik dan dapat diandalkan.

3. HASIL DAN PEMBAHASAN

3.1. HASIL EKSPERIMEN

Table 1. Hasil Implementasi Sistem Lampu Lalu Lintas

Deskripsi Input	Perilaku Lampu	Keterangan
Tombol 1 ditekan	Lampu merah berkedip 5 kali.	Setiap kali tombol ditekan, lampu merah berkedip 5 kali dengan interval 500 ms.
Tombol 2 ditekan	Lampu merah dan hijau berkedip bergantian sebanyak 5 kali.	Lampu merah dan hijau menyala bergantian, masing-masing selama 500 ms.

Tombol 3 ditekan	Lampu merah, kuning, dan hijau berkedip bergantian sebanyak 5 kali.	Lampu merah, kuning, dan hijau menyala bergantian, masing-masing selama 500 ms.
Tidak ada tombol ditekan	Semua lampu (merah, kuning, hijau) dalam keadaan mati.	Sistem menunggu input dari tombol untuk mengaktifkan lampu.

Berdasarkan hasil eksperimen, sistem lampu lalu lintas berhasil diimplementasikan dengan baik. Setiap tombol menghasilkan perilaku lampu yang sesuai dengan logika yang telah diprogram. Sistem ini dapat digunakan sebagai simulasi dasar untuk memahami prinsip kerja lampu lalu lintas dan pengendaliannya menggunakan mikrokontroler ESP32.

4. LAMPIRAN

4.1. KODE PROGRAM

```
5.1. #include <Arduino.h>
6.1.
7.1. int tombol1 = 34;
8.1. int tombol2 = 35;
9.1. int tombol3 = 32;
10.1.
11.1.   int redlight = 23;
12.1.   int yellowlight = 22;
13.1.   int greenlight = 21;
14.1.
15.1.   // Fungsi debouncing
16.1.   boolean debounce(int pin) {
17.1.       delay(50); // Delay untuk
           menghindari bouncing
18.1.       return digitalRead(pin) ==
           LOW; // Kembalikan keadaan tombol
           setelah debounce
19.1.   }
20.1.
21.1.   void setup() {
22.1.       Serial.begin(115200);
23.1.       Serial.println("Hello, ESP
           32!");
24.1.
25.1.       // Set pin mode untuk
           tombol
26.1.       pinMode(tombol1,
           INPUT_PULLUP);
27.1.       pinMode(tombol2,
           INPUT_PULLUP);
28.1.       pinMode(tombol3,
           INPUT_PULLUP);
29.1.
30.1.       // Set pin mode untuk LED
31.1.       pinMode(redlight, OUTPUT);
32.1.       pinMode(yellowlight,
           OUTPUT);
33.1.       pinMode(greenlight,
           OUTPUT);
34.1.
35.1.       // Pastikan semua LED mati
           di awal
36.1.       digitalWrite(redlight,
           LOW);
37.1.       digitalWrite(yellowlight,
           LOW);
38.1.       digitalWrite(greenlight,
           LOW);
39.1.   }
```

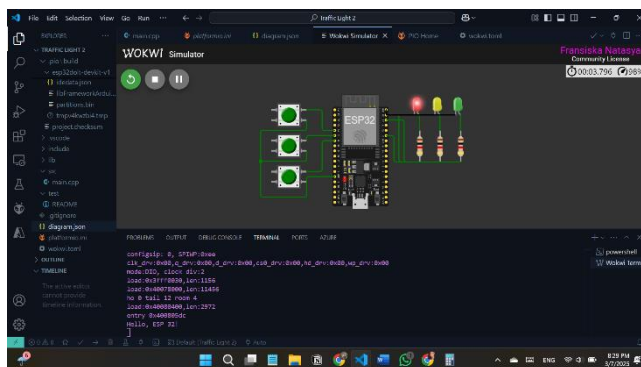
```
40.1.
41.1.   void loop() {
42.1.       // Tombol 1: Lampu merah
           berkedip 5x
43.1.       if (debounce(tombol1)) {
44.1.           for (int i = 0; i < 5;
               i++) {
45.1.               digitalWrite(redlight,
               HIGH);
46.1.               delay(500); // Nyalakan
               selama 500 ms
47.1.               digitalWrite(redlight,
               LOW);
48.1.               delay(500); // Mati
               selama 500 ms
49.1.           }
50.1.       }
51.1.
52.1.       // Tombol 2: Lampu merah
           dan hijau berkedip bergantian
53.1.       if (debounce(tombol2)) {
54.1.           for (int i = 0; i < 5;
               i++) {
55.1.               digitalWrite(redlight,
               HIGH);
56.1.               digitalWrite(greenlight,
               LOW);
57.1.               delay(500); // Merah
               nyala, hijau mati
58.1.               digitalWrite(redlight,
               LOW);
59.1.               digitalWrite(greenlight,
               HIGH);
60.1.               delay(500); // Merah
               mati, hijau nyala
61.1.           }
62.1.           digitalWrite(greenlight,
               LOW); // Pastikan hijau mati
               setelah selesai
63.1.       }
64.1.
65.1.       // Tombol 3: Lampu merah,
           kuning, hijau berkedip bergantian
66.1.       if (debounce(tombol3)) {
67.1.           for (int i = 0; i < 5;
               i++) {
68.1.               digitalWrite(redlight,
               HIGH);
69.1.               digitalWrite(yellowlight,
               LOW);
```

```

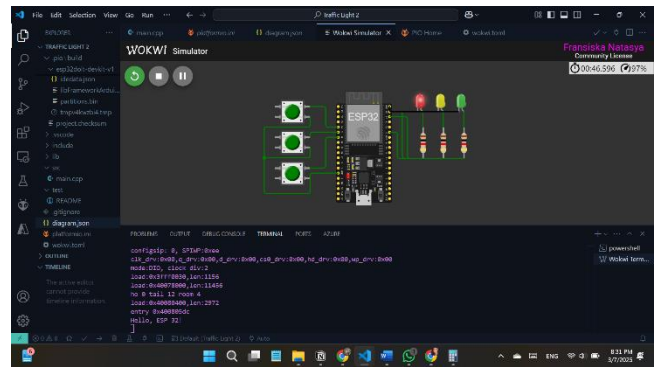
70.1.         digitalWrite(greenlight
, LOW);
71.1.         delay(500); // Merah
nyala, kuning dan hijau mati
72.1.         digitalWrite(redlight,
LOW);
73.1.         digitalWrite(yellowligh
t, HIGH);
74.1.         digitalWrite(greenlight
, LOW);
75.1.         delay(500); // Kuning
nyala, merah dan hijau mati
77.1.         digitalWrite(redlight,
LOW);
78.1.         digitalWrite(yellowligh
t, LOW);
79.1.         digitalWrite(greenlight
, HIGH);
80.1.         delay(500); // Hijau
nyala, merah dan kuning mati
81.1.     }
82.1.         digitalWrite(greenlight,
LOW); // Pastikan hijau mati
setelah selesai
83.1.     }
84.1.
85.1. // Delay kecil untuk
mengurangi bouncing tombol
86.1.     delay(100);
87.1. }

```

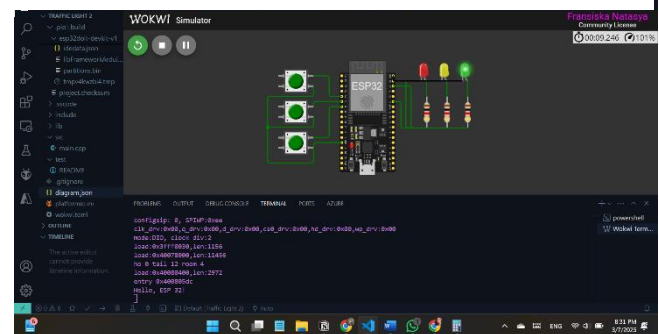
4.2. SIMULASI LAMPU LALU LINTAS



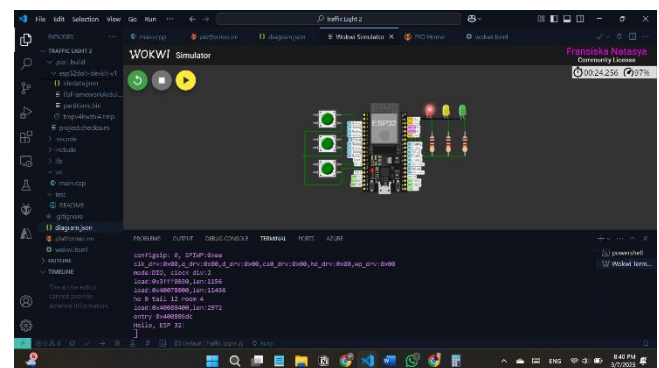
Gambar 1. Tombol 1 ditekan, lampu merah menyala 5 kali



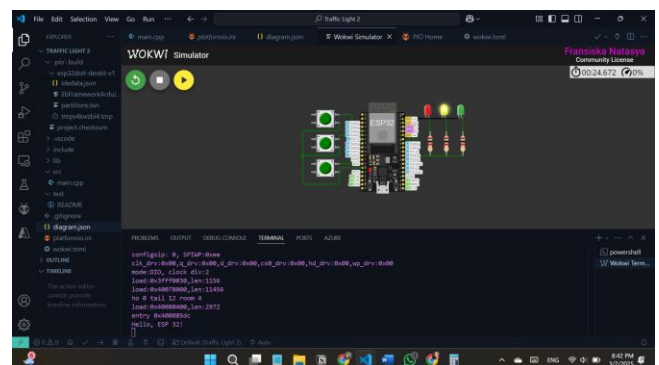
Gambar 2. Tombol 2 ditekan, lampu merah dan lampu hijau menyala 5 kali secara bergantian, lampu merah menyala



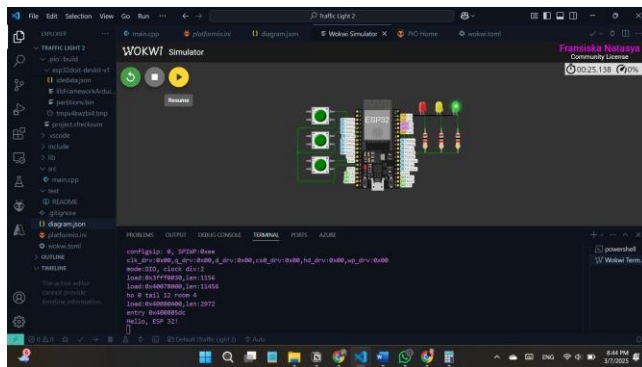
Gambar 3. Tombol 2 ditekan, lampu merah dan lampu hijau menyala 5 kali secara bergantian, lampu hijau menyala



Gambar 4. Tombol 3 ditekan, lampu merah, lampu kuning dan lampu hijau menyala secara bergantian, lampu merah menyala



Gambar 5. Tombol 3 ditekan, lampu merah, lampu kuning dan lampu hijau menyala secara bergantian, lampu kuning menyala



Gambar 6. Tombol 3 ditekan, lampu merah, lampu kuning dan lampu hijau menyala secara bergantian, lampu hijau menyala