

# LAPORAN PROYEK IMPLEMENTASI CARA KERJA TRAFFIC LIGHT ATAU LAMPU LALU LINTAS PADA PIO

**Rumiris Butarbutar**

Program Studi Teknologi Informasi, Fakultas Vokasi, Universitas Brawijaya

Email: ririssbtr09@student.ub.ac.id

**Abstrak** – Lampu lalu lintas merupakan sistem pengendalian lalu lintas yang berperan penting dalam mengatur pergerakan kendaraan dan mengurangi risiko kecelakaan. Perkembangan teknologi memungkinkan sistem ini diimplementasikan menggunakan mikrokontroler dan platform pemrograman. Penelitian ini mengimplementasikan simulasi sistem lampu lalu lintas berbasis mikrokontroler ESP32 menggunakan Platform Input/Output (PIO) pada lingkungan Wokwi. Sistem dirancang dengan tiga LED yang merepresentasikan lampu merah, kuning, dan hijau, dikendalikan berdasarkan siklus waktu terprogram: hijau selama 20 detik, kuning 5 detik, dan merah 30 detik. Proses pengujian dilakukan dengan memastikan bahwa setiap lampu beroperasi sesuai dengan logika kontrol yang ditetapkan. Hasil eksperimen menunjukkan bahwa sistem berjalan secara stabil dan akurat dalam menjalankan siklus pergantian lampu lalu lintas. Kesimpulan dari penelitian ini menunjukkan bahwa ESP32 dapat digunakan secara efektif untuk mengontrol sistem lalu lintas berbasis PIO, yang dapat dikembangkan lebih lanjut dengan integrasi sensor untuk pengaturan lalu lintas yang lebih adaptif.

**Kata Kunci:** Sistem Lampu Lalu Lintas, ESP32, PIO, Simulasi, Wokwi

*Abstract -- Traffic lights are a traffic control system that plays an important role in regulating vehicle movement and reducing the risk of accidents. Technological developments allow this system to be implemented using a microcontroller and programming platform. This study implements a simulation of a traffic light system based on an ESP32 microcontroller using the Input/Output (PIO) Platform in the Wokwi environment. The system is designed with three LEDs representing red, yellow, and green lights, controlled based on a programmed time cycle: green for 20 seconds, yellow for 5 seconds, and red for 30 seconds. The testing process is carried out by ensuring that each light operates according to the specified control logic. The experimental results show that the system runs stably and accurately in running the traffic light change cycle. The conclusion of this study shows that ESP32 can be used effectively to control a PIO-based traffic system, which can be further developed with sensor integration for more adaptive traffic control.*

**Keywords:** Traffic Light System, ESP32, PIO, Simulation, Wokwi

## 1.2 Tujuan

### 1. Pendahuluan

#### 1.1 Latar Belakang

Lampu lalu lintas merupakan sistem pengaturan lalu lintas yang sangat penting dalam kehidupan sehari-hari. Dengan adanya lampu lalu lintas, arus kendaraan dapat dikendalikan dengan lebih efektif untuk mengurangi kemacetan dan kecelakaan. Proyek ini bertujuan untuk membuat sistem simulasi lampu lalu lintas menggunakan PIO (Platform Input/Output) dengan mikrokontroler.

- Mendesain dan mengimplementasikan sistem traffic light menggunakan PIO.
- Memahami prinsip kerja lampu lalu lintas dalam pengaturan lalu lintas.
- Mengembangkan keterampilan dalam pemrograman mikrokontroler untuk kontrol sinyal lalu lintas.

## 2. Metodologi

## 2.1 Alat dan Bahan

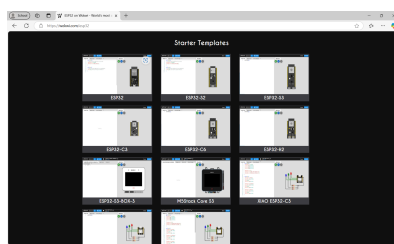
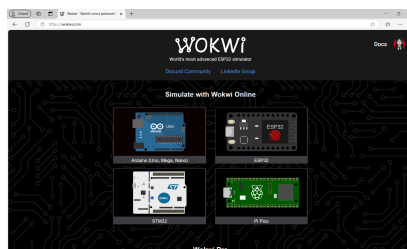
- Simulasi menggunakan website Wokwi
- Mikrokontroler (ESP 32)
- LED (Merah, Kuning, Hijau)
- Software pemrograman

## 2.2 Langkah Implementasi

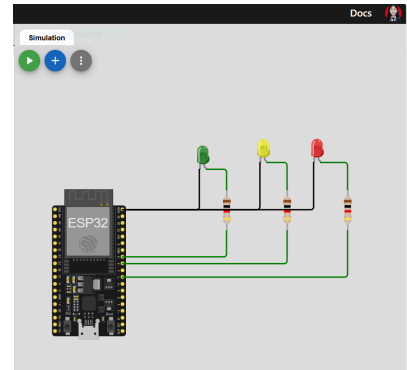
Sistem terdiri dari tiga lampu LED yang dikendalikan oleh mikrokontroler menggunakan PIO untuk mengontrol nyala dan mati lampu berdasarkan waktu yang telah ditentukan, yang mana lampu hijau menyala selama 20 detik, lampu kuning menyala selama 5 detik, lampu merah menyala selama 30 detik, dan Proses berulang secara terus-menerus.

### Implementasi dan Konfigurasi

1. Menyiapkan akun dan login pada website Wokwi (<https://wokwi.com>)



2. Menambahkan LED dan Resistor lalu menghubungkan pada mikrokontroler sesuai dengan pin yang telah ditentukan.



3. Menulis kode program berikut untuk mengontrol nyala dan mati LED berdasarkan waktu pada file sketch.ino pada website wokwi

```
sketch.ino  diagram.json  Library Manager
1  int ledm = 17;
2  int ledk = 18;
3  int ledh = 19;
4
5
6
7  void setup() {
8      // put your setup code here, to run once:
9      Serial.begin(115200);
10     pinMode(ledm, OUTPUT);
11     pinMode(ledk, OUTPUT);
12     pinMode(ledh, OUTPUT);
13
14 }
15
16
17 void loop() {
18     // put your main code here, to run repeatedly:
19     digitalWrite(ledm, HIGH);
20     digitalWrite(ledk, LOW);
21     digitalWrite(ledh, LOW); //merah yang nyala
22     delay(30000);
23     digitalWrite(ledm, LOW);
24     digitalWrite(ledk, HIGH);
25     digitalWrite(ledh, LOW); //kuning yang nyala
26     delay(5000);
27     digitalWrite(ledm, LOW);
28     digitalWrite(ledk, LOW);
29     digitalWrite(ledh, HIGH); //hijau yang nyala
30     delay(20000);
31
32 }
```

```
int ledm = 17;
```

```
int ledk = 18;
```

```
int ledh = 19 ;
```

```
void setup() {
```

```
    // put your setup code here, to
    run once:
```

```
    Serial.begin(115200);
```

```
    pinMode(ledm, OUTPUT);
```

```
    pinMode(ledk, OUTPUT);
```

```
    pinMode(ledh, OUTPUT);
```

```
}
```

```
void loop() {
```

```
    // put your main code here, to  
    run repeatedly:
```

```
    digitalWrite(ledm, HIGH);
```

```
    digitalWrite(ledk, LOW);
```

```
    digitalWrite(ledh, LOW); //merah  
    yang nyala
```

```
    delay(30000);
```

```
    digitalWrite(ledm, LOW);
```

```
    digitalWrite(ledk, HIGH);
```

```
    digitalWrite(ledh, LOW);  
    //kuning yang nyala
```

```
    delay(5000);
```

```
    digitalWrite(ledm, LOW);
```

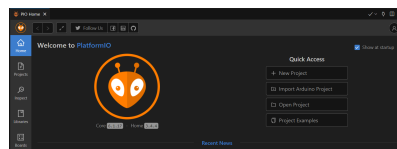
```
    digitalWrite(ledk, LOW);
```

```
    digitalWrite(ledh, HIGH); //hijau  
    yang nyala
```

```
    delay(20000);
```

```
}
```

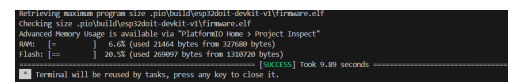
4. Buka PlatformIO IDE pada VSCode yang mana sebelumnya di install terlebih dahulu, dan mulai membuat proyek



5. Kemudian *copy code* pada file sketch.ino pada website wokwi, setelah itu *paste* pada file main.cpp pada *Visual Studio Code*



6. Selanjutnya lakukan *built* hingga *success*

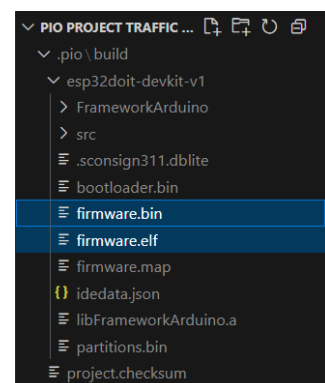
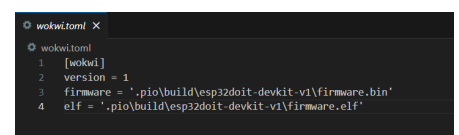


7. Buat file baru dengan format wokwi.toml pada *Visual Studio Code*. Yang mana isi file tersebut ialah salinan dari file wokwi.toml pada *google*.

A basic `wokwi.toml` file looks like this:

```
[wokwi]
version = 1
firmware = 'path-to-your-firmware.hex'
elf = 'path-to-your-firmware.elf'
```

8. Rubah *firmware* dan *elf* sesuai versi *Visual Studio Code* masing-masing



9. Selanjutnya buat file diagram pada *Visual Studio Code* yang mana isi file tersebut ialah hasil *copy* dari file diagram.json pada *wokwi*

```
1 {
2   "version": 1,
3   "author": "Rumiris Butarbutar (riris)",
4   "editor": "wokwi",
5   "parts": [
6     {
7       "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 0, "left": 0, "attrs": {}
8     },
9     {
10      "type": "wokwi-led",
11      "id": "led1",
12      "top": -70.8,
13      "left": 195.8,
14      "attrs": { "color": "green" }
15    },
16    {
17      "type": "wokwi-led",
18      "id": "led2",
19      "top": -80.4,
20      "left": 282.2,
21      "attrs": { "color": "yellow" }
22    },
23    {
24      "type": "wokwi-led", "id": "led3", "top": -80.4, "left": 359, "attrs": { "color":
25    },
26    {
27      "type": "wokwi-resistor",
28      "id": "r1",
29      "top": 24,
30      "left": 220.25,
31      "rotate": 90,
32      "attrs": { "value": "1000" }
33    },
34    {
35      "type": "wokwi-resistor",
36      "id": "r2",
37      "top": 24,
38      "left": 306.65,
39      "rotate": 90,
40      "attrs": { "value": "1000" }
41    },
42    {
43      "type": "wokwi-resistor",
44      "id": "r3",
45      "top": 24,
46      "left": 393.05,
47      "rotate": 90,
48      "attrs": { "value": "1000" }
49    }
50  ]
51 }
```

```
{
  "version": 1,
  "author": "Rumiris Butarbutar (riris)",
  "editor": "wokwi",
  "parts": [
    {
      "type":
"board-esp32-devkit-c-v4", "id":
"esp", "top": 0, "left": 0, "attrs":
{} },
    {
      "type": "wokwi-led",
      "id": "led1",
      "top": -70.8,
      "left": 195.8,
      "attrs": { "color": "green" }
    },
    {
      "type": "wokwi-led",
      "id": "led2",
      "top": -80.4,
      "left": 282.2,
```

```
      "attrs": { "color": "yellow" }
    },
    {
      "type": "wokwi-led", "id":
"led3", "top": -80.4, "left": 359,
      "attrs": { "color": "red" } },
    {
      "type": "wokwi-resistor",
      "id": "r1",
      "top": 24,
      "left": 220.25,
      "rotate": 90,
      "attrs": { "value": "1000" }
    },
    {
      "type": "wokwi-resistor",
      "id": "r2",
      "top": 24,
      "left": 306.65,
      "rotate": 90,
      "attrs": { "value": "1000" }
    },
    {
      "type": "wokwi-resistor",
      "id": "r3",
      "top": 24,
      "left": 393.05,
      "rotate": 90,
      "attrs": { "value": "1000" }
    }
  ],
  "connections": [
```

```

        [ "esp:TX",
"$serialMonitor:RX", "", [] ],

        [ "esp:RX",
"$serialMonitor:TX", "", [] ],

        [ "led1:C", "esp:GND.2",
"black", [ "v57.6", "h-110.04" ] ],

        [ "led2:C", "esp:GND.2",
"black", [ "v67.2", "h-196.44" ] ],

        [ "led3:C", "esp:GND.2",
"black", [ "v67.2", "h-273.24" ] ],

        [ "esp:19", "r1:2", "green", [
"h148.84", "v-37.2" ] ],

        [ "led1:A", "r1:1", "green", [
"h28.8", "v28.8" ] ],

        [ "r2:2", "esp:18", "green", [
"v46.8", "h-235.24" ] ],

        [ "led2:A", "r2:1", "green", [
"v0", "h28.8" ] ],

        [ "led3:A", "r3:1", "green", [
"v0", "h38.4" ] ],

        [ "r3:2", "esp:17", "green", [
"v66", "h-321.64" ] ],

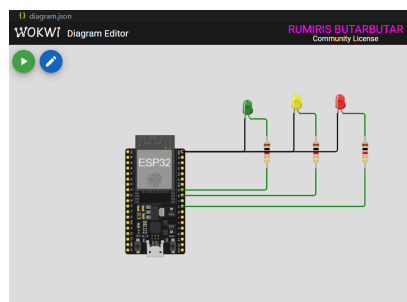
    ],

    "dependencies": {}

}

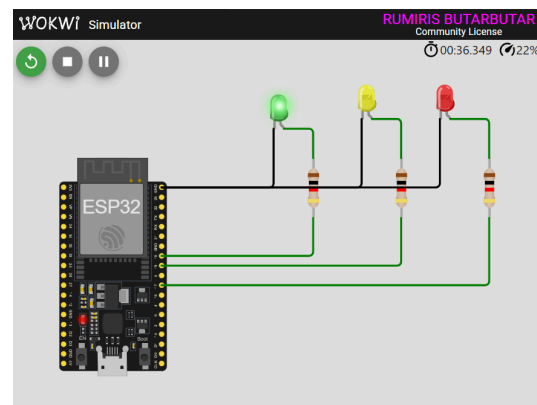
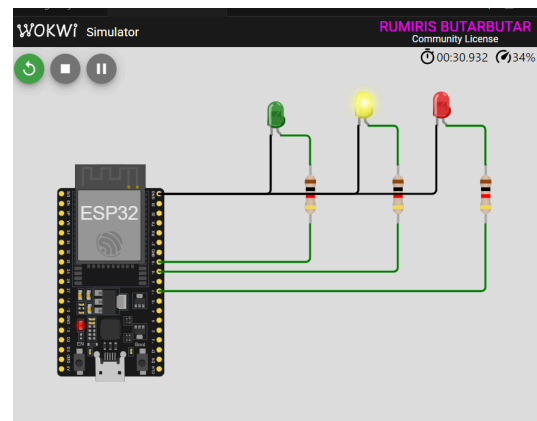
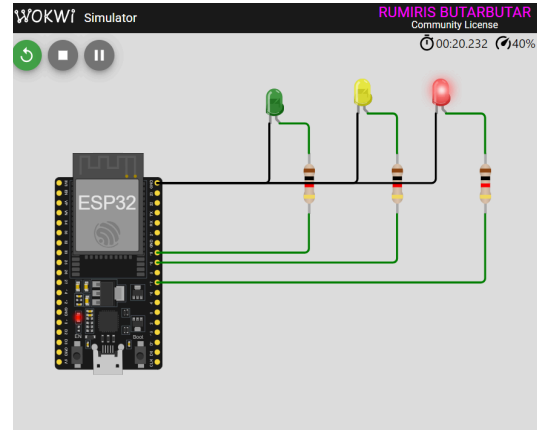
```

10. Selanjutnya *rename* file diagram pada Visual Studio Code menjadi diagram.json, maka akan muncul simulasi seperti berikut



### 3. Pengujian Hasil dan Pembahasan

Dari hasil pengujian, sistem lampu lalu lintas bekerja sesuai dengan rancangan. Lampu hijau, kuning, dan merah menyala sesuai waktu yang ditentukan dan berulang tanpa kendala.



### 4. Kesimpulan

Proyek traffic lights menggunakan PIO telah berhasil dibuat dan diuji dengan hasil yang sesuai harapan. Sistem ini dapat menjadi dasar untuk pengembangan lebih lanjut, seperti implementasi sensor kendaraan untuk pengaturan lampu lalu lintas yang lebih cerdas.