

LAPORAN PROYEK PENERAPAN SISTEM KERJA TRAFFIC LIGHT MENGUNAKAN PLATFORMIO

Dosen Pengampu :
Ir. Subairi, ST., MT., IPM

MATA KULIAH :
Internet of Things



Disusun Oleh :

Tiara Kasih Keiko Sulistya
NIM : 233140707111129

**TEKNOLOGI INFORMASI
DEPARTEMEN INDUSTRI KREATIF DAN DIGITAL
FAKULTAS VOKASI
UNIVERSITAS BRAWIJAYA
MALANG
2025**

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Lampu lalu lintas telah menjadi bagian tak terpisahkan dari sistem transportasi modern yang berfungsi untuk mengatur pergerakan kendaraan dan pejalan kaki di persimpangan jalan. Tanpa adanya pengaturan yang tepat, lalu lintas dapat menjadi kacau, meningkatkan risiko kecelakaan dan kemacetan yang merugikan. Di kota-kota besar, kemacetan lalu lintas menjadi masalah utama yang menyebabkan pemborosan waktu, energi, serta meningkatkan tingkat polusi udara.

Seiring dengan perkembangan teknologi, sistem pengaturan lampu lalu lintas kini bisa diotomatisasi dengan menggunakan mikrokontroler dan perangkat lunak yang lebih canggih. Salah satu solusi yang dapat diimplementasikan adalah simulasi lampu lalu lintas menggunakan Platform Input/Output (PIO). Dengan menggunakan mikrokontroler seperti ESP32, sistem ini memungkinkan kontrol yang lebih efisien dan lebih mudah dimodifikasi sesuai dengan kebutuhan.

1.2 Tujuan

Adapun tujuan dari proyek ini adalah sebagai berikut:

1. Merancang dan mengembangkan sistem lampu lalu lintas berbasis Platform Input/Output (PIO) yang dapat mengatur siklus nyala lampu sesuai dengan urutan dan durasi tertentu.
2. Menciptakan sistem simulasi lampu lalu lintas yang mudah dipahami dan diterapkan menggunakan mikrokontroler.
3. Menjadi dasar untuk pengembangan sistem pengaturan lalu lintas yang lebih pintar di masa depan.

BAB II

PERANCANGAN SISTEM

2.1 Peralatan dan Bahan

1. Platform Wokwi sebagai simulasi rangkaian elektronik dan pengujian kode pemrograman yang akan digunakan. Wokwi menyediakan antarmuka yang memudahkan untuk merancang dan menguji sistem secara virtual.
2. Mikrokontroler ESP32 dipilih sebagai mikrokontroler yang digunakan untuk mengendalikan lampu lalu lintas.
3. Tiga LED digunakan untuk mewakili lampu lalu lintas yang masing-masing berwarna merah, kuning, dan hijau. LED ini akan dikendalikan untuk menyala sesuai dengan siklus yang telah ditentukan.
4. Software Pemrograman

2.2 Skema Sistem

1. Mikrokontroler (ESP32): Bertindak sebagai pengendali utama yang mengatur aliran listrik ke LED berdasarkan siklus yang telah diprogram.
2. LED Merah, Kuning, dan Hijau: Mewakili lampu lalu lintas yang dikendalikan oleh mikrokontroler untuk menyala sesuai dengan durasi yang telah ditentukan.
3. Resistor: Digunakan untuk membatasi arus yang mengalir ke LED, sehingga mencegah kerusakan pada komponen.

2.3 Alur Kerja Sistem

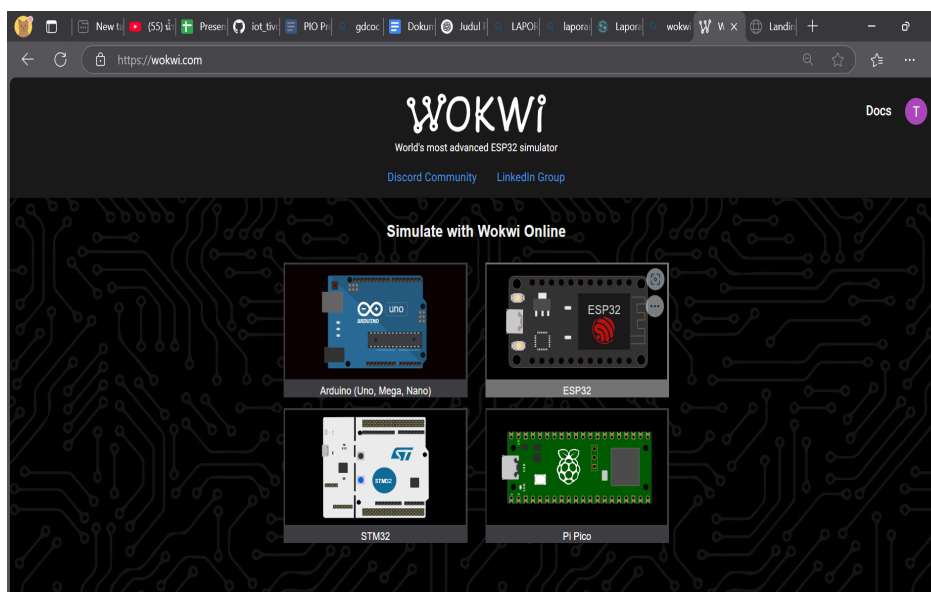
1. **Lampu hijau** menyala selama 20 detik, menandakan bahwa kendaraan dapat melintas.
2. Setelah 20 detik, **Lampu kuning** menyala selama 5 detik, memberikan tanda peringatan bahwa lampu akan berganti.
3. Kemudian, **Lampu merah** menyala selama 30 detik, memberi tanda bahwa kendaraan harus berhenti.
4. Siklus ini akan terus berulang secara otomatis.

BAB III

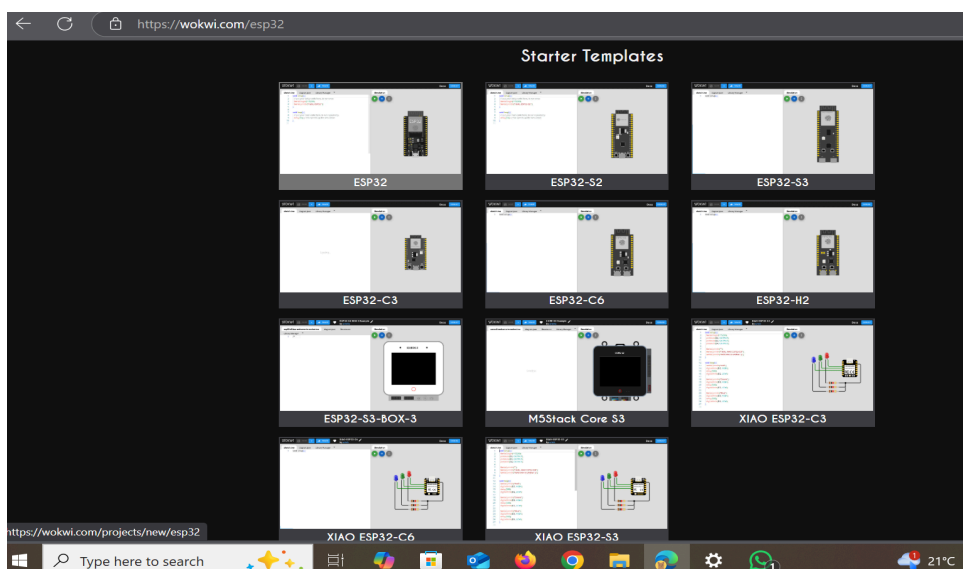
IMPLEMENTASI

3.1 Instalasi dan Pengaturan

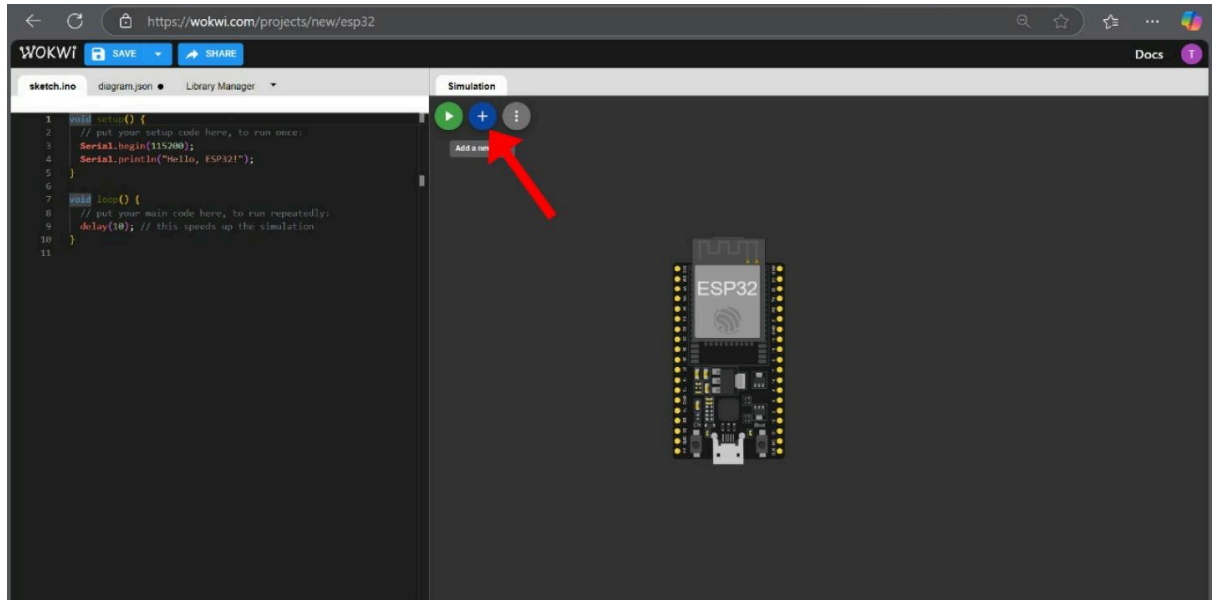
1. buka website **Wokwi**: www.wokwi.com.
2. Jika belum memiliki akun, buat akun terlebih dahulu dengan memilih **Sign Up**.
3. Lalu memilih Mikrokontroler ESP32



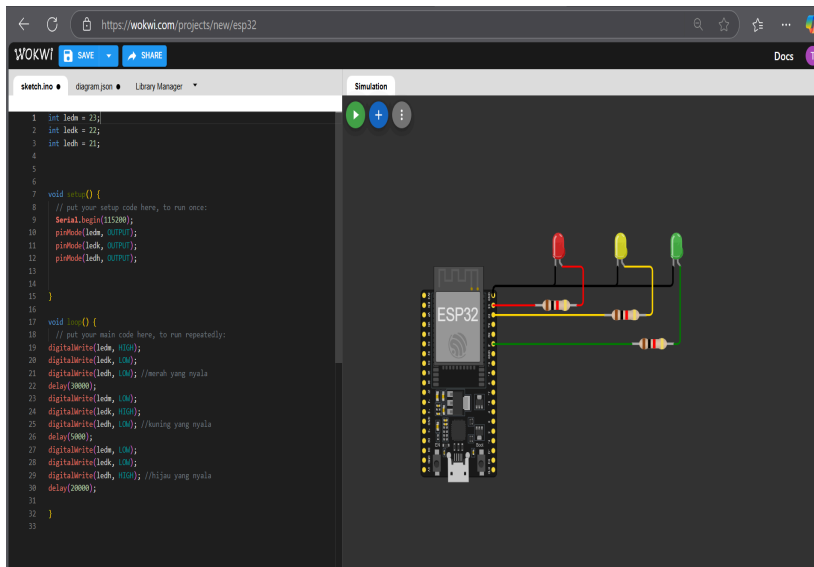
4. Pilih **Starter Template** dan klik **ESP32**



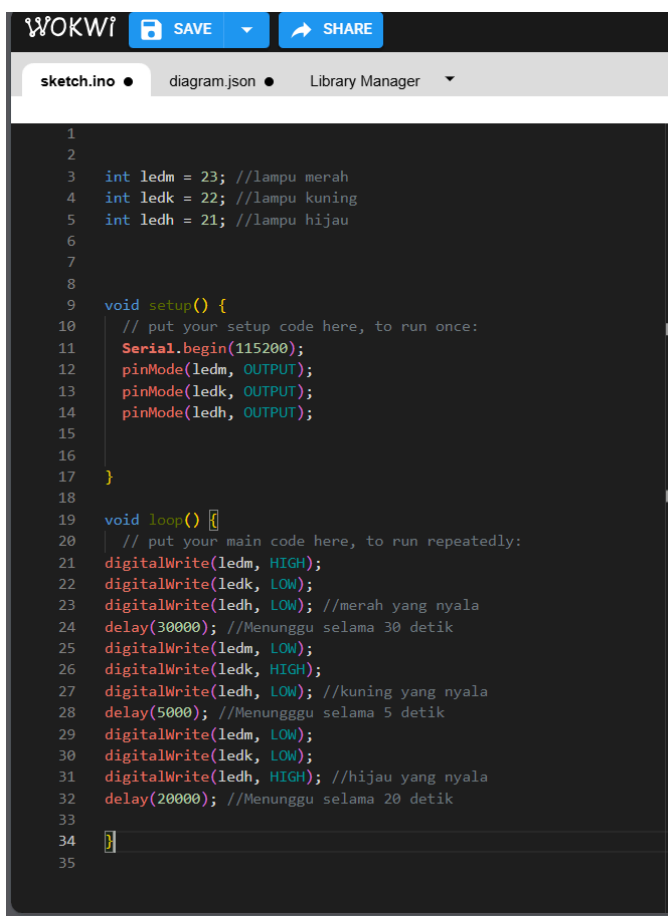
- Setelah memilih starter template akan masuk ke editor Wokwi yang menampilkan mikrokontroler ESP32.
- Di sisi kiri editor, pilih **Add Components** atau klik ikon + untuk menambahkan komponen lain ke dalam rangkaian simulasi.



- Pilih LED dari daftar komponen yang muncul.
- Tambahkan tiga buah LED berwarna yang akan mewakili lampu lalu lintas. Setiap LED akan dihubungkan dengan pin yang berbeda pada ESP32.
- Klik pada LED yang dipilih, lalu letakkan di area rangkaian.
- Lalu pilih Resistor dari daftar komponen dan tambahkan tiga resistor untuk membatasi arus yang mengalir ke setiap LED.
- Hubungkan setiap LED ke pin yang berbeda di ESP32 (misalnya pin esp: GND, 2, 23, 22, dan 21 untuk LED Merah, Kuning, dan Hijau).
- Hubungkan resistor ke pin yang sesuai dan pastikan semua komponen terhubung dengan baik.



13. Untuk menulis program di editor kode (**sketch.ino**) yang akan mengendalikan LED berdasarkan urutan dan durasi tertentu. Tulis kode untuk mengendalikan siklus lampu lalu lintas sesuai dengan urutan (merah, kuning, hijau) dan durasi (hijau selama 20 detik, kuning selama 5 detik, merah selama 30 detik).



Kode program traffic light:

```
int ledm = 23; //lampu merah

int ledk = 22; //lampu kuning

int ledh = 21; //lampu hijau

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); // lampu merah yang nyala

    delay(30000); //Menunggu selama 30 detik

    digitalWrite(ledm, LOW);

    digitalWrite(ledk, HIGH);

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

    delay(5000); //Menunggu selama 5 detik

    digitalWrite(ledm, LOW);

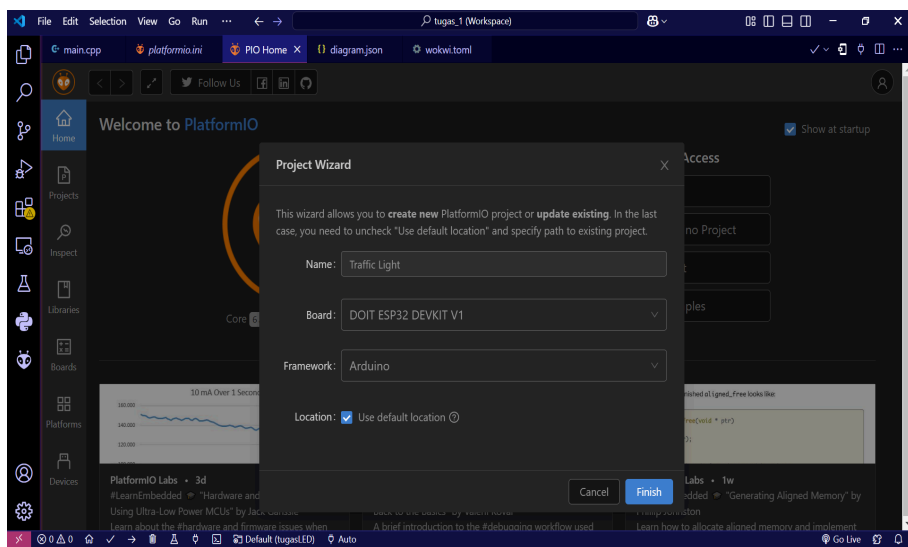
    digitalWrite(ledk, LOW);

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

    delay(20000); //Menunggu selama 20 detik
```

}

14. Kemudian, buka VSCode dan instal PlatformIO IDE di menu extensions
15. Klik ikon PlatformIO di sisi kiri untuk mengakses PlatformIO.
16. Pilih New Project dan nama proyek
17. Pilih Board: Pilih DOIT ESP32 DEVKIT V1
18. Pilih Framework: Pilih Arduino sebagai framework.
19. Klik Finish untuk membuat proyek baru.



20. Setelah proyek dibuat, PlatformIO akan mengonfigurasi direktori dan file yang diperlukan.
21. Di dalam direktori proyek, buka file src/main.cpp untuk menulis kode untuk mengontrol lampu lalu lintas.
22. Copy kode di sketch.ino di Wokwi ke main.cpp


```

1 #include <Arduino.h>
2
3 int ledm = 23;
4 int ledk = 22;
5 int ledh = 21;
6
7
8
9 void setup() {
10     // put your setup code here, to run once:
11     Serial.begin(115200);
12     pinMode(ledm, OUTPUT);
13     pinMode(ledk, OUTPUT);
14     pinMode(ledh, OUTPUT);
15
16
17
18
19 void loop() {
20     // put your main code here, to run repeatedly:
21     digitalWrite(ledm, HIGH);
22     digitalWrite(ledk, LOW);
23     digitalWrite(ledh, LOW); //merah yang nyala
24     delay(3000);
25     digitalWrite(ledm, LOW);
26     digitalWrite(ledk, HIGH);
27     digitalWrite(ledh, LOW); //kuning yang nyala
28     delay(5000);
29     digitalWrite(ledm, LOW);
30     digitalWrite(ledk, LOW);
31     digitalWrite(ledh, LOW);

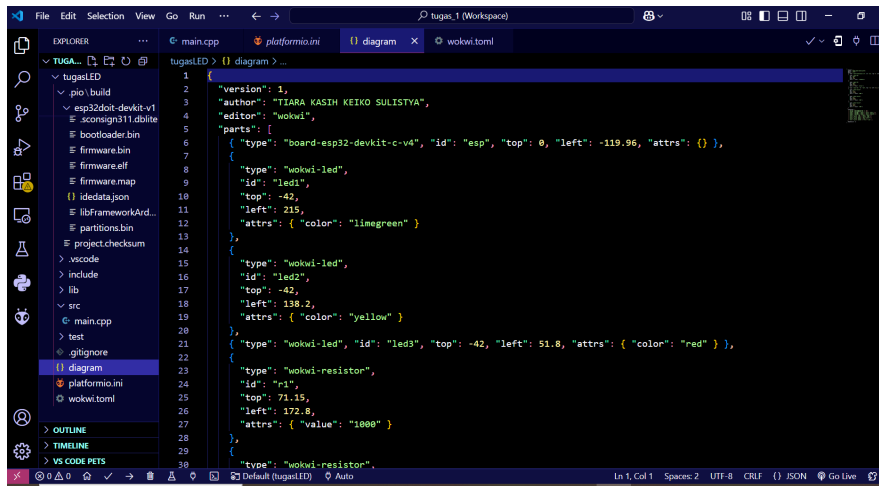
```

23. Kemudian buat file baru dengan nama ‘diagram’ di direktori proyek. lalu copy kode diagram.json di Wokwi.

```

1 {
2   "version": 1,
3   "author": "TIARA KASIH KEIKO SULISTYA",
4   "editor": "wokwi",
5   "parts": [
6     { "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 0, "left": 167.5, "width": 100, "height": 50 },
7     { "type": "wokwi-led", "id": "led1", "top": -32.4, "left": 167.5, "width": 10, "height": 10 },
8     {
9       "type": "wokwi-led",
10      "id": "led2",
11      "top": -32.4,
12      "left": 224.6,
13      "attrs": { "color": "yellow" }
14    },
15    {
16      "type": "wokwi-led",
17      "id": "led3",
18      "top": -32.4,
19      "left": 282.2,
20      "attrs": { "color": "limegreen" }
21    },
22    {
23      "type": "wokwi-resistor",
24      "id": "r1",
25      "top": 42.35,
26      "left": 144,
27      "attrs": { "value": "1000" }
28    },
29    {
30      "type": "wokwi-resistor",
31      "id": "r2",
32      "top": 61.55,
33      "left": 201.6,
34      "attrs": { "value": "1000" }
35    },
36    {
37      "type": "wokwi-resistor",

```



```
{  
  "version": 1,  
  "author": "TIARA KASIH KEIKO SULISTYA",  
  "editor": "wokwi",  
  "parts": [  
    { "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 0, "left": -119.96, "attrs": {} },  
    {  
      "type": "wokwi-led",  
      "id": "led1",  
      "top": -42,  
      "left": 215,  
      "attrs": { "color": "limegreen" }  
    },  
    {  
      "type": "wokwi-led",  
      "id": "led2",  
      "top": -42,  
      "left": 138.2,  
      "attrs": { "color": "yellow" }  
    },  
    { "type": "wokwi-led", "id": "led3", "top": -42, "left": 51.8, "attrs": { "color": "red" } },  
    {  
      "type": "wokwi-resistor",  
      "id": "r1",  
      "top": 71.15,  
      "left": 172.8,  
      "attrs": { "value": "1000" }  
    },  
    {  
      "type": "wokwi-resistor",  
      "id": "r2",  
      "top": 71.15,  
      "left": 172.8,  
      "attrs": { "value": "1000" }  
    }  
  ]  
}
```

```
},  
  
{"type": "wokwi-led", "id": "led3", "top": -42, "left": 51.8, "attrs": { "color": "red" } },  
  
{  
  "type": "wokwi-resistor",  
  
  "id": "r1",  
  
  "top": 71.15,  
  
  "left": 172.8,  
  
  "attrs": { "value": "1000" }  
},  
  
{  
  "type": "wokwi-resistor",  
  
  "id": "r2",  
  
  "top": 42.35,  
  
  "left": 134.4,  
  
  "attrs": { "value": "1000" }  
},  
  
{  
  "type": "wokwi-resistor",  
  
  "id": "r3",  
  
  "top": 32.75,  
  
  "left": 38.4,  
  
  "attrs": { "value": "1000" }  
}  
],  
  
"connections": [  
  
  [ "esp:TX", "$serialMonitor:RX", "", [] ],
```

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

[ "led3:C", "esp:GND.2", "black", [ "v19.2", "h-76.4" ] ],

[ "led2:C", "esp:GND.2", "black", [ "v19.2", "h-172.4" ] ],

[ "led1:C", "esp:GND.2", "black", [ "v19.2", "h-258.8" ] ],

[ "led3:A", "r3:2", "red", [ "h28.8", "v38.4" ] ],

[ "r3:1", "esp:23", "red", [ "v-9.6", "h-48" ] ],

[ "led2:A", "r2:2", "gold", [ "h38.4", "v48" ] ],

[ "r2:1", "esp:22", "gold", [ "v0", "h-153.6" ] ],

[ "led1:A", "r1:2", "green", [ "v0" ] ],

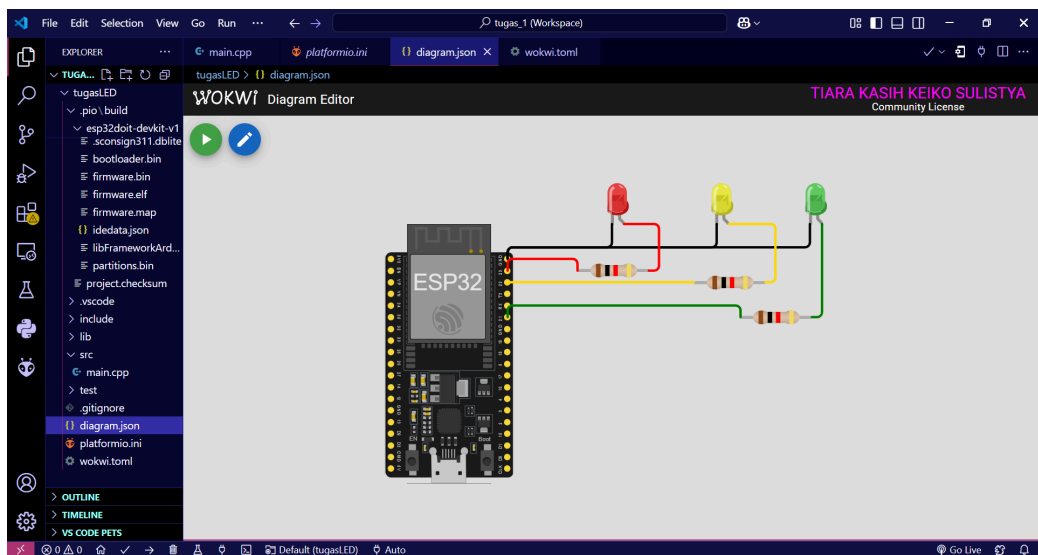
[ "r1:1", "esp:21", "green", [ "v-9.6", "h-192" ] ]

],

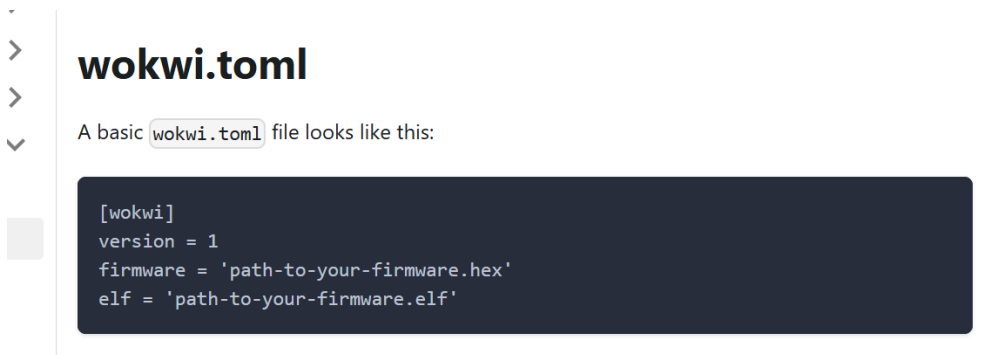
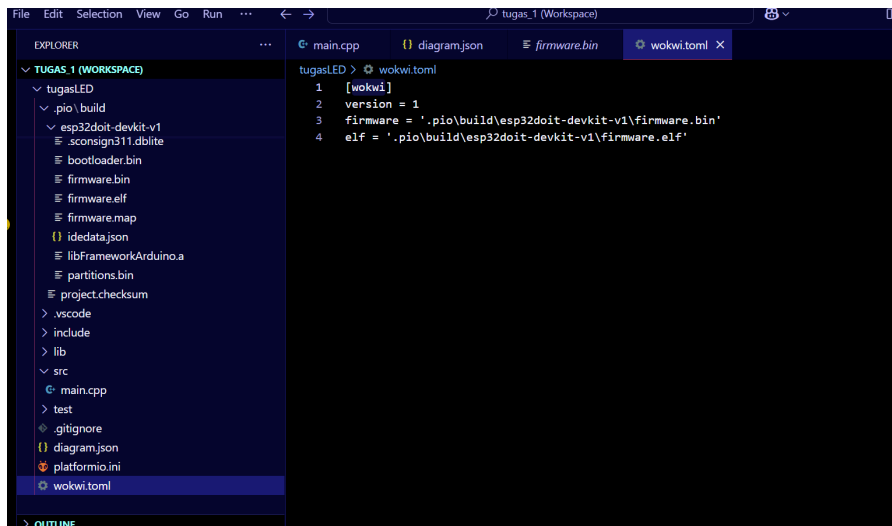
"dependencies": {}

}
```

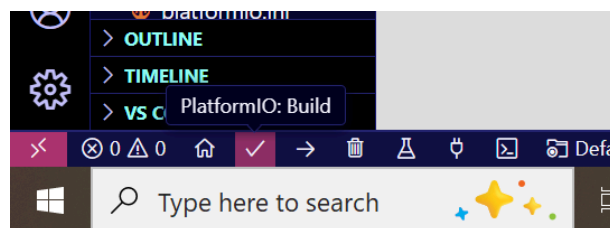
24. Setelah itu, rename file diagram menjadi diagram.json untuk menampilkan simulasi.



25. Buat file baru dengan judul 'wokwi.toml' di direktori proyek. Kemudian buka [Configuring Your Project \(wokwi.toml\) | Wokwi Docs](#) lalu copy kode wokwi.toml di file baru wokwi.toml tadi.



26. Lalu lakukan build dengan cara klik tanda centang, seperti dibawah ini.



27. Setelah melakukan build, cari file firmware.bin dan firmware.elf kemudian copy relative path.



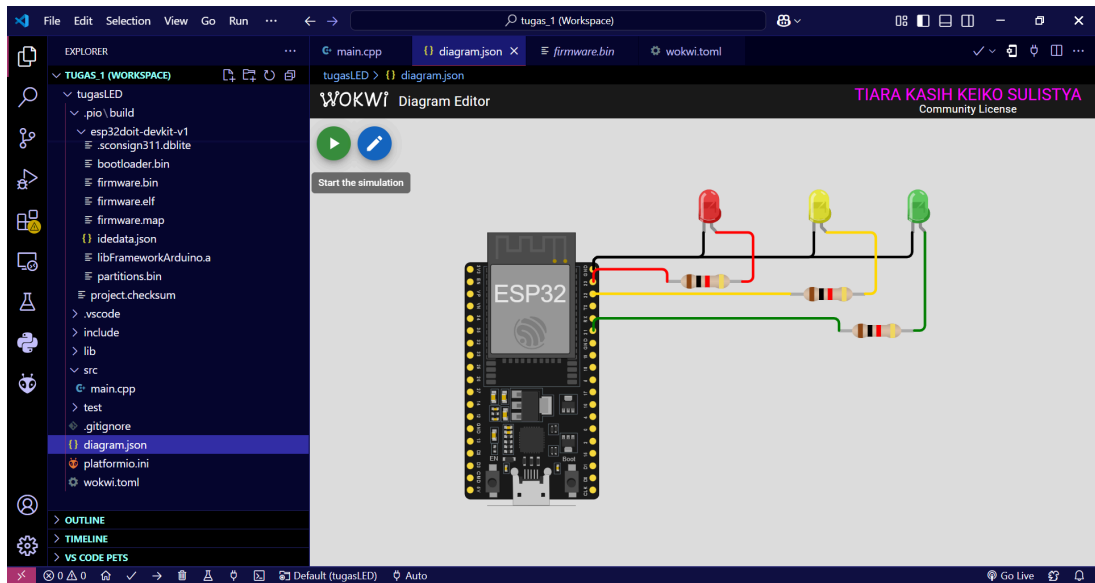
28. Lalu buka file wokwi.toml dan ganti firmware = 'path-to-your-firmware.hex' elf = 'path-to-your-firmware.elf' dengan copy path firmware.bin dan firmware.elf.

```

main.cpp  diagram.json  firmware.bin  wokwi.toml
tugasLED > wokwi.toml
1  [wokwi]
2  version = 1
3  firmware = '.pio\build\esp32doit-devkit-v1\firmware.bin'
4  elf = '.pio\build\esp32doit-devkit-v1\firmware.elf'

```

29. Setelah itu buka file diagram.json dan klik tombol hijau (start the simulation). Lalu tunggu hingga proses upload selesai. Jika berhasil, ESP32 akan menjalankan program dan lampu lalu lintas akan berfungsi sesuai dengan yang telah diprogram.



BAB IV

Hasil dan Pembahasan

4.1 Hasil Pengujian

Siklus lampu lalu lintas berhasil berjalan dengan baik. Lampu hijau menyala selama 20 detik, lampu kuning menyala selama 5 detik, dan lampu merah menyala selama 30 detik. Siklus ini berjalan terus menerus tanpa ada gangguan, sesuai dengan durasi yang telah diprogram dalam kode.

WOKWI Simulator

TIARA KASIH KEIKO SULISTYA
Community License

00:07.479 100%



ets Jul 29 2019 12:21:46

```
rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
confgipsip: 0, SPIWP:0xee
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
```

Default (tugasLED) Auto

Go Live

main.cpp diagram.json Wokwi Simulator X firmware.bin wokwi.toml

WOKWI Simulator

TIARA KASIH KEIKO SULISTYA
Community License

00:32.804 96%



ets Jul 29 2019 12:21:46

```
rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
confgipsip: 0, SPIWP:0xee
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
```




4.2 Pembahasan

Berdasarkan hasil pengujian, sistem lampu lalu lintas yang dikembangkan dengan menggunakan mikrokontroler ESP32 bekerja dengan baik dan sesuai dengan yang diharapkan. Menggunakan mikrokontroler ESP32 untuk mengendalikan lampu lalu lintas memungkinkan pengontrolan yang lebih fleksibel dan dapat dikembangkan lebih lanjut, misalnya dengan menambahkan sensor kendaraan untuk pengaturan lampu yang lebih adaptif. Menggunakan PlatformIO dan Wokwi untuk membuat simulasi rangkaian mempermudah proses pengembangan dan memungkinkan prototyping sistem lampu lalu lintas dengan cepat tanpa memerlukan hardware fisik.

BAB V

PENUTUP

5.1 Kesimpulan

Proyek sistem lampu lalu lintas berbasis mikrokontroler ESP32 berhasil diimplementasikan dengan baik. Lampu hijau, kuning, dan merah menyala sesuai dengan waktu yang telah diprogram, yakni 20 detik untuk hijau, 5 detik untuk kuning, dan 30 detik untuk merah, tanpa gangguan. Penggunaan ESP32 terbukti efektif sebagai pengendali utama, dan dengan menggunakan PlatformIO, pengembangan sistem menjadi lebih mudah.