**PRAKTIK PEMBUATAN SISTEM TRAFICC LIGHT MENGGUNAKAN ESP32**

*Difa Aqilah*

*Fakultas Vokasi, Unibversitas Brawijaya*

*Email:* [*difa040404@gmail.com*](mailto:difa040404@gmail.com)

**ABSTRAK**

Eksperimen ini bertujuan untuk merancang dan mengimplementasikan sistem pengontrol traffic light berbasis mikrokontroler ESP32. Sistem ini menggunakan tiga LED yang masing-masing mewakili lampu merah, kuning, dan hijau, yang menyala dengan durasi tertentu untuk mensimulasikan pengaturan lampu lalu lintas. Durasi yang telah ditentukan untuk setiap lampu adalah lampu merah menyala selama 30 detik, lampu kuning selama 5 detik, dan lampu hijau selama 20 detik. Mikrokontroler ESP32 digunakan untuk mengatur waktu penyalaan LED berdasarkan durasi yang telah diprogram. Setiap LED dihubungkan dengan resistor untuk membatasi arus yang mengalir. Desain rangkaian disimulasikan menggunakan Wokwi, sebuah platform simulasi untuk mikrokontroler, sementara pemrograman dilakukan menggunakan Vscode.

Hasil eksperimen menunjukkan bahwa sistem berhasil berfungsi sesuai dengan rencana, di mana setiap lampu menyala dengan durasi yang tepat dan berganti secara otomatis tanpa intervensi manual. Sistem ini bekerja dengan baik dalam mengatur siklus lampu merah, kuning, dan hijau sesuai dengan waktu yang ditentukan. Dengan hasil ini, dapat disimpulkan bahwa penggunaan mikrokontroler ESP32 untuk mengontrol sistem traffic light dapat berjalan dengan efektif dan efisien, serta mudah diimplementasikan dalam skala yang lebih besar untuk aplikasi IoT pada pengelolaan lalu lintas.

*Kata kunci— Internet of Things, ESP32, Traffic Light, LED, Wokwi*

**BAB I**

**PENDAHULUAN**

**1.1 Latar Belakang**

Sistem lalu lintas merupakan salah satu sistem yang sangat penting untuk mengatur aliran kendaraan dan mencegah kemacetan, salah satu komponen utama dalam sistem ini adalah traffic light yang mengatur lampu merah, kuning, dan hijau pada perempatan jalan. Dengan semakin berkembangnya teknologi, penerapan Internet of Things (IoT) pada sistem traffic light menjadi hal yang menarik untuk diteliti. IoT memungkinkan pengelolaan dan kontrol yang lebih cerdas terhadap sistem traffic light menggunakan perangkat mikrokontroler seperti ESP32.

**1.2 Tujuan**

Tujuan dari eksperimen ini adalah untuk merancang dan mengimplementasikan sistem traffic light sederhana menggunakan mikrokontroler ESP32. Lampu merah akan menyala selama 30 detik, lampu kuning selama 5 detik, dan lampu hijau selama 20 detik. Sistem ini menggunakan perangkat ESP32, LED, dan resistor untuk mensimulasikan pengaturan lampu lalu lintas.

**BAB II**

**METHODOLOGY**

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

### Mikrokontroler: ESP32

### LED: Merah, Kuning, Hijau

### Resistor

### Software: Vscode, Wokwi Simulator

**2.2 Implementation Steps (Langkah Implementasi)**

1. **Persiapan Alat dan Bahan:**

* **Wokwi**: Platform simulasi online untuk mikrokontroler yang digunakan untuk mendesain rangkaian dan menguji sistem tanpa perangkat keras fisik
* **VSCode**: Text editor yang digunakan untuk menulis dan meng-upload kode ke ESP32 menggunakan PlatformIO
* **ESP32 Board**: Mikrokontroler yang digunakan untuk mengontrol traffic light
* **LED (Merah, Kuning, Hijau)**: Digunakan untuk mensimulasikan lampu lalu lintas
* **Resistor**: Membatasi arus listrik yang mengalir ke LED
* Sambungkan masing-masing LED dengan resistor ke pin GPIO pada ESP32 ( GPIO 21 untuk merah, GPIO 22 untuk kuning, dan GPIO 23 untuk hijau)

1. **Desain Sistem:**

* Kunjungi situs Wokwi ([wokwi.com](https://wokwi.com/)) dan buat proyek baru
* Pilih mikrokontroler ESP32
* Tambahkan 3 LED (merah, kuning, hijau) ke rangkaian. Sambungkan masing-masing LED ke pin GPIO ESP32 (GPIO 21 untuk merah, GPIO 22 untuk kuning, dan GPIO 23 untuk hijau)
* Pasang resistor pada masing-masing LED untuk membatasi arus
* Pastikan kabel GND ESP32 terhubung ke setiap LED

1. **Kode Program:**

* Tulis code pada sketch.ino di wokwi.com untuk mengontrol Led

Code:  
#include <Arduino.h>

#define RED\_LED 21

#define YELLOW\_LED 22

#define GREEN\_LED 23

void setup() {

    Serial.begin(115200);

    Serial.println("ESP32 Traffic Light Simulation");

    pinMode(RED\_LED, OUTPUT);

    pinMode(YELLOW\_LED, OUTPUT);

    pinMode(GREEN\_LED, OUTPUT);

}

void loop() {

    // Lampu Merah menyala selama 30 detik

    digitalWrite(RED\_LED, HIGH);

    digitalWrite(YELLOW\_LED, LOW);

    digitalWrite(GREEN\_LED, LOW);

    Serial.println("Lampu Merah ON");

    delay(30000);

    // Lampu Kuning menyala selama 5 detik

    digitalWrite(RED\_LED, LOW);

    digitalWrite(YELLOW\_LED, HIGH);

    digitalWrite(GREEN\_LED, LOW);

    Serial.println("Lampu Kuning ON");

    delay(5000);

    // Lampu Hijau menyala selama 20 detik

    digitalWrite(RED\_LED, LOW);

    digitalWrite(YELLOW\_LED, LOW);

    digitalWrite(GREEN\_LED, HIGH);

    Serial.println("Lampu Hijau ON");

    delay(20000);

}

* Instal Vscode dan pastikan sudah menginstal PlatformIO di dalam Vscode
* Buka Vscode dan buat proyek baru menggunakan PlatformIO
* Pilih board **DOIT ESP32 DEVKIT V1** di PlatformIO
* Masukan code yang telah di buat pada wokwi.com (sketch.ino) kedalam file main.ccp di folder src Vscode
* Buat file diagram.json pada vscode dan masukan code yang ada pada wokwi.com (diagram.json) kedalam file diagram.json pada Vscode
* Buat file wokwi.toml lalu masukan code berikut:

[wokwi]

version = 1

firmware = 'path-to-your-firmware.hex'

elf = 'path-to-your-firmware.elf'

* Build code pada file wokwi.toml
* Jika sudah sukses, ubah code pada file wokwi.toml dengan code berikut:

[wokwi]

version = 1

firmware = '.pio\build\esp32doit-devkit-v1\firmware.bin'

* Lalu coba jalan kan file diagram.json pada Vscode
* Jika Untuk menjalankan program membutuhkan lisensi, maka aktifkan lisensi terlebih dulu
* Llau coba jalankan kembali

1. **Pengujian:**

* Periksa apakah LED menyala sesuai dengan urutan dan durasi yang telah ditentukan
* Jika ada masalah, periksa kembali rangkaian dan kode program, pastikan tidak ada kesalahan dalam pengkabelan atau sintaks kode

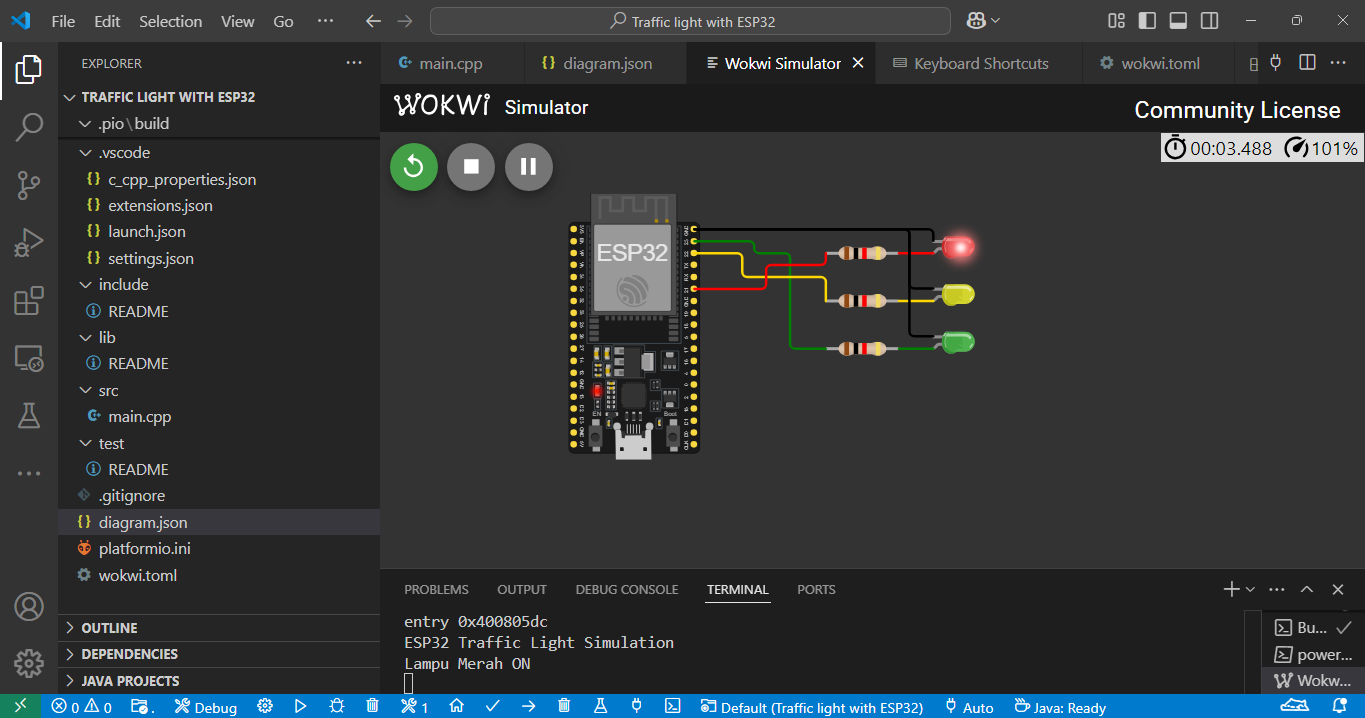
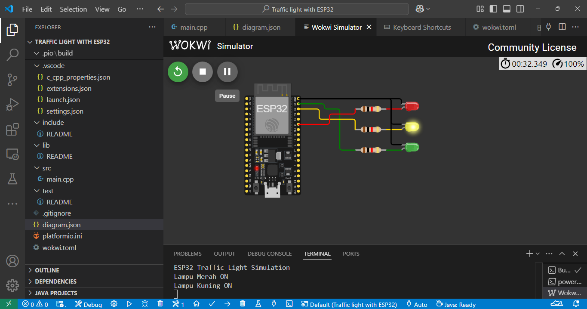
**BAB III**

**RESULTS AND DISCUSSION (Hasil dan Pembahasan)**

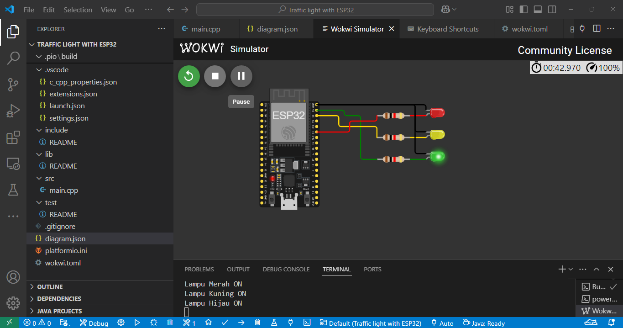
**3.1 Experimental Results (Hasil Eksperimen)**

Hasil eksperimen menunjukkan bahwa sistem traffic light berjalan sesuai dengan desain. Lampu merah menyala selama 30 detik, lampu kuning menyala selama 5 detik, dan lampu hijau menyala selama 20 detik. Berikut adalah urutan dan waktu nyala setiap LED:

|  |  |
| --- | --- |
| LED | Durasi |
| Merah | 30 |
| Kuning | 5 |
| Hijau | 20 |

(lampu merah on) (lampu kuning on)



(lampu hijau on)

**BAB IV**

**APPENDIX**

**4.1 Kode program**

* Kode file main.ccp pada vscode

#include <Arduino.h>

#define RED\_LED 21

#define YELLOW\_LED 22

#define GREEN\_LED 23

void setup() {

Serial.begin(115200);

Serial.println("ESP32 Traffic Light Simulation");

pinMode(RED\_LED, OUTPUT);

pinMode(YELLOW\_LED, OUTPUT);

pinMode(GREEN\_LED, OUTPUT);

}

void loop() {

// Lampu Merah menyala selama 30 detik

digitalWrite(RED\_LED, HIGH);

digitalWrite(YELLOW\_LED, LOW);

digitalWrite(GREEN\_LED, LOW);

Serial.println("Lampu Merah ON");

delay(30000);

// Lampu Kuning menyala selama 5 detik

digitalWrite(RED\_LED, LOW);

digitalWrite(YELLOW\_LED, HIGH);

digitalWrite(GREEN\_LED, LOW);

Serial.println("Lampu Kuning ON");

delay(5000);

// Lampu Hijau menyala selama 20 detik

digitalWrite(RED\_LED, LOW);

digitalWrite(YELLOW\_LED, LOW);

digitalWrite(GREEN\_LED, HIGH);

Serial.println("Lampu Hijau ON");

delay(20000);

}

* Kode file diagram.json pada vscode

{

"version": 1,

"author": "Anonymous maker",

"editor": "wokwi",

"parts": [

{ "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 0, "left": -43.16, "attrs": {} },

{

"type": "wokwi-led",

"id": "led1",

"top": 18,

"left": 246.6,

"rotate": 90,

"attrs": { "color": "red" }

},

{

"type": "wokwi-led",

"id": "led2",

"top": 94.8,

"left": 246.6,

"rotate": 90,

"attrs": { "color": "limegreen" }

},

{

"type": "wokwi-led",

"id": "led3",

"top": 56.4,

"left": 246.6,

"rotate": 90,

"attrs": { "color": "yellow" }

},

{

"type": "wokwi-resistor",

"id": "r1",

"top": 42.35,

"left": 163.2,

"attrs": { "value": "1000" }

},

{

"type": "wokwi-resistor",

"id": "r2",

"top": 119.15,

"left": 163.2,

"attrs": { "value": "1000" }

},

{

"type": "wokwi-resistor",

"id": "r3",

"top": 80.75,

"left": 163.2,

"attrs": { "value": "1000" }

}

],

"connections": [

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

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

[ "led1:C", "esp:GND.2", "black", [ "v-9.2", "h-192" ] ],

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

[ "led3:A", "r3:2", "gold", [ "h0" ] ],

[ "led2:A", "r2:2", "green", [ "h0" ] ],

[ "r2:1", "esp:23", "green", [ "h-28.8", "v-76.8", "h-28.8", "v-9.6" ] ],

[ "r3:1", "esp:22", "gold", [ "v-19.2", "h-67.2", "v-19.2" ] ],

[ "r1:1", "esp:21", "red", [ "v9.6", "h-48", "v19.2" ] ],

[ "led3:C", "esp:GND.2", "black", [ "h-19.2", "v-47.6" ] ],

[ "led2:C", "esp:GND.2", "black", [ "h-19.2", "v-86" ] ]

],

"dependencies": {}

}

* Kode file wokwi.toml

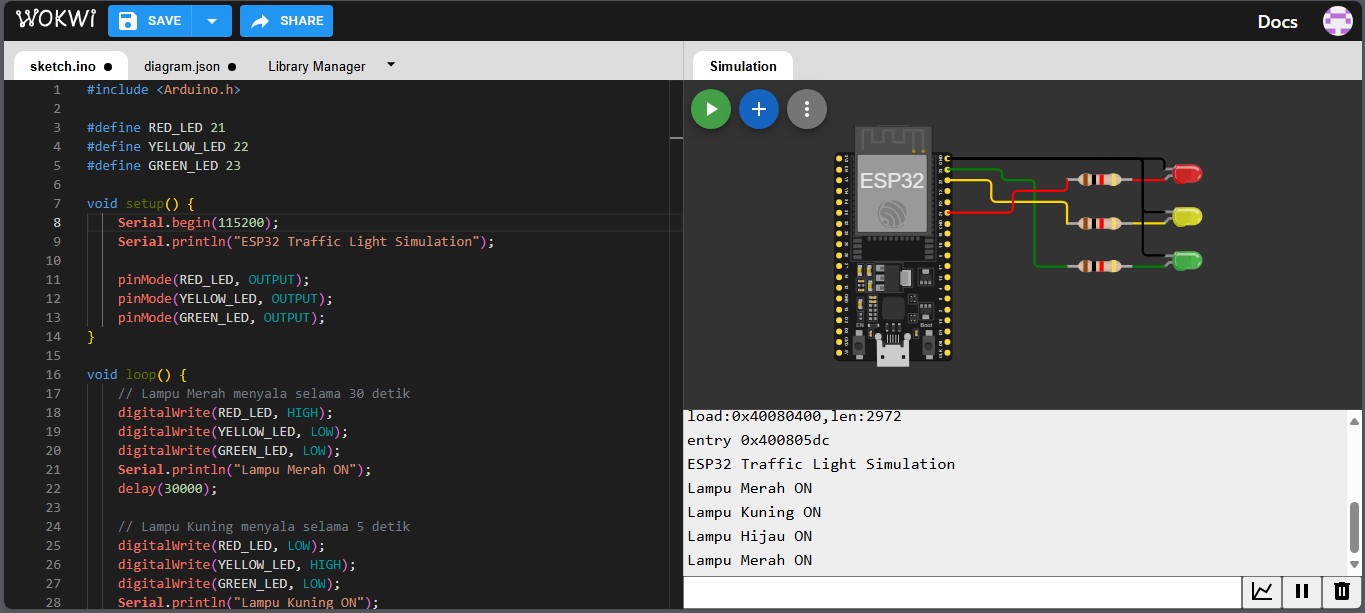
[wokwi]

version = 1

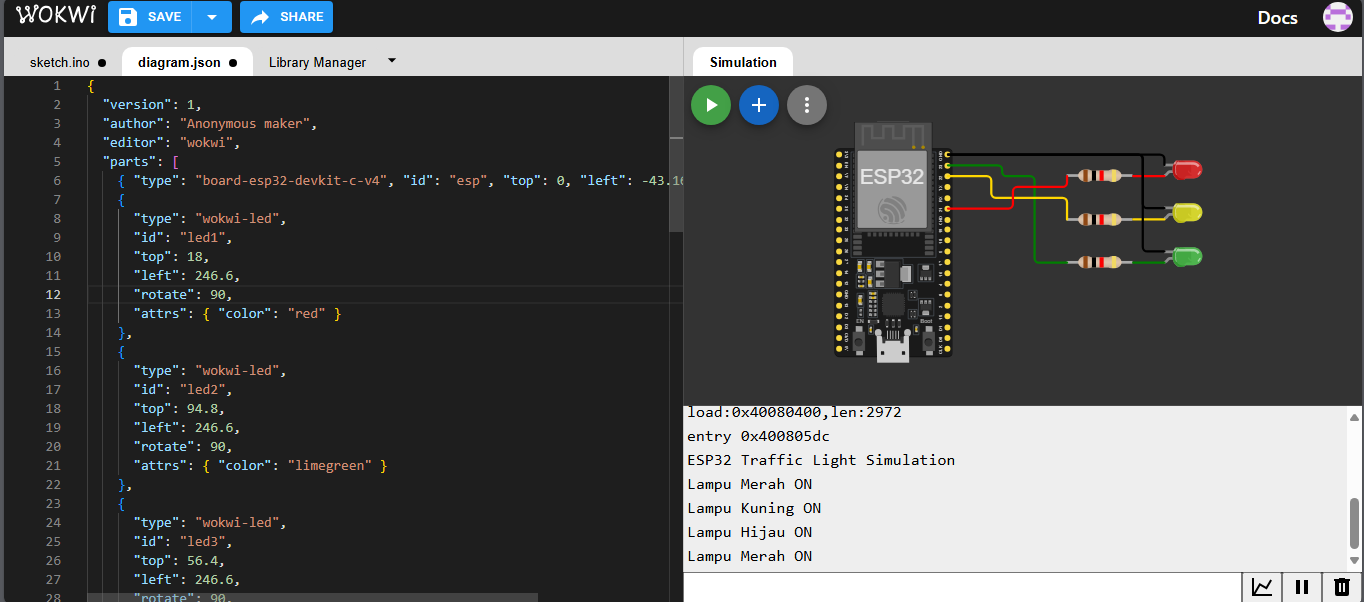
firmware = '.pio\build\esp32doit-devkit-v1\firmware.bin'

elf = '.pio\build\esp32doit-devkit-v1\firmware.elf'

* Screenshoot tampilan code sketch.ino dan desain pada wokwi.com

****

* Screenshoot tampilan code diagram.json dan desain pada wokwi.com

****

**BAB V**

**PENUTUP**

**5.1 Kesimpulan**

Dalam praktikum ini, telah berhasil dibuat sistem pengontrol traffic light sederhana menggunakan mikrokontroler ESP32 dengan LED sebagai indikator lampu merah, kuning, dan hijau. Sistem ini telah diimplementasikan dengan durasi masing-masing lampu yang sesuai, yaitu lampu merah selama 30 detik, lampu kuning selama 5 detik, dan lampu hijau selama 20 detik. Penggunaan perangkat ESP32 dan pemrograman dengan PlatfromIO memudahkan pengembangan sistem ini.

Sistem ini juga telah berhasil disimulasikan menggunakan Wokwi, yang memungkinkan untuk melakukan uji coba tanpa perlu perangkat keras fisik. Hasil pengujian menunjukkan bahwa sistem berfungsi sesuai dengan yang diharapkan, konsep dasar pengontrolan lalu lintas dengan ESP32 telah berhasil diterapkan.