**Praktik Pembuatan Akun Wokwi dan Github**

**Disusun Untuk Memenuhi Tugas Mata Kuliah :**

**Internet Of Things**

**Dosen Pengampu : Rachmad Andri Atmoko, S.ST, M.T, MCF**



**Disusun Oleh :**

1. Rifqi Athallah G.R(233140701111008)

PROGRAM STUDI TEKNOLOGI INFORMASI(

FAKULTAS VOKASI

UNIVERSITAS BRAWIJAYA

2025

**Abstract**

Percobaan ini bertujuan untuk merancang dan mengimplementasikan sistem Lampu Lalu Lintas (Traffic Light) berbasis mikrokontroler ESP32. Sistem ini menggunakan tiga LED dengan warna hijau, merah, dan kuning yang menyala secara bergantian dalam satu loop dengan waktu tunda (delay) 1000 ms atau 1 detik. Proses pengembangan dilakukan menggunakan Visual Studio Code (VS Code) dengan bantuan ekstensi PlatformIO dan Wokwi untuk simulasi perangkat keras secara virtual.

Dalam implementasi ini, langkah awal yang dilakukan adalah menginstal dan mengonfigurasi PlatformIO serta Wokwi untuk mendukung pengembangan berbasis ESP32. Selanjutnya, proyek baru dibuat di PlatformIO dengan memilih board ESP32 DOIT Devkit v1 serta menggunakan framework Arduino.

Hasil dari percobaan ini menunjukkan bahwa sistem Traffic Light dapat berjalan dengan baik dalam simulasi Wokwi di VS Code. Tiga LED menyala secara bergantian sesuai dengan pengaturan waktu yang telah ditentukan, serta hasil dari proses eksekusi dapat diamati melalui tampilan print di terminal yang menunjukkan status perubahan warna LED, yaitu red, orange, dan green. Percobaan ini membuktikan bahwa penggunaan ESP32 dalam pengendalian lampu lalu lintas berbasis simulasi dapat dilakukan secara efisien dengan bantuan PlatformIO dan Wokwi.

# **BAB I**

# **PENDAHULUAN**

## **1.1** **Latar Belakang**

Internet of Things (IoT) merupakan salah satu teknologi yang semakin berkembang dan memiliki peran penting dalam berbagai bidang, termasuk dalam sistem otomasi dan kontrol. Salah satu implementasi sederhana dari sistem IoT adalah simulasi Lampu Lalu Lintas (Traffic Light) menggunakan mikrokontroler. Percobaan ini dilakukan untuk memahami prinsip dasar pengendalian perangkat elektronik menggunakan ESP32 serta mengembangkan sistem kontrol berbasis pemrograman dalam lingkungan Visual Studio Code (VS Code) dengan ekstensi PlatformIO dan Wokwi.

Pada percobaan ini, sistem Traffic Light dirancang menggunakan tiga LED dengan warna hijau, merah, dan kuning yang menyala secara bergantian dengan waktu tunda (delay) 1000 ms atau 1 detik. Langkah pertama dalam perancangan sistem ini adalah menginstal dan mengonfigurasi ekstensi PlatformIO dan Wokwi pada VS Code untuk mendukung pengembangan berbasis ESP32. Selanjutnya, proyek baru dibuat di PlatformIO dengan menentukan board ESP32 DOIT Devkit v1 dan menggunakan framework Arduino.

Tahapan berikutnya adalah pembuatan dua file konfigurasi utama, yaitu wokwi.toml dan diagram.json, yang berfungsi untuk mengatur simulasi perangkat keras. Setelah itu, relative path dari file firmware.elf dan firmware.bin disalin dan dimasukkan ke dalam wokwi.toml agar sistem dapat dijalankan dengan benar dalam simulator. Langkah terakhir adalah melakukan konfigurasi kode pada file main.cpp untuk mengontrol nyala LED sesuai dengan urutan yang telah ditentukan dalam sistem Traffic Light.

Hasil dari percobaan ini menunjukkan bahwa simulasi Traffic Light dapat berjalan dengan baik di Wokwi menggunakan VS Code. Tiga LED berhasil menyala secara bergantian sesuai dengan pengaturan waktu yang telah ditentukan. Percobaan ini memberikan wawasan mengenai pemrograman mikrokontroler dan integrasi perangkat keras dalam simulasi berbasis IoT, yang dapat menjadi dasar untuk pengembangan proyek otomatisasi lebih lanjut.

## **1.2 Tujuan eksperimen**

Percobaan ini bertujuan untuk merancang dan mengimplementasikan sistem Lampu Lalu Lintas (*Traffic Light*) menggunakan mikrokontroler ESP32. Sistem ini terdiri dari tiga LED berwarna hijau, merah, dan kuning yang menyala secara bergantian dalam satu *loop* dengan waktu tunda (*delay*) 1000 ms atau 1 detik. Selain itu, percobaan ini juga bertujuan untuk memahami proses konfigurasi dan simulasi sistem menggunakan *Visual Studio Code (VS Code)* dengan ekstensi *PlatformIO* dan *Wokwi*. Hasil dari percobaan ini dapat diamati melalui tampilan *print* di terminal yang menunjukkan perubahan status LED dengan indikator warna *red*, *orange*, dan *green*.

# 

# **BAB II**

**Metodologi**

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

* mikrokontroler ESP32
* Lampu Led Red, Orange dan Green
* Kabel Lampu Led ke mikrokontroler ESP32

**2.2 Implementation Steps**

Pada percobaan ini, sistem Traffic Light dirancang menggunakan tiga LED dengan warna hijau, merah, dan kuning yang menyala secara bergantian dengan waktu tunda (delay) 1000 ms atau 1 detik. Langkah pertama dalam perancangan sistem ini adalah menginstal dan mengonfigurasi ekstensi PlatformIO dan Wokwi pada VS Code untuk mendukung pengembangan berbasis ESP32. Selanjutnya, proyek baru dibuat di PlatformIO dengan menentukan board ESP32 DOIT Devkit v1 dan menggunakan framework Arduino.

Tahapan berikutnya adalah pembuatan dua file konfigurasi utama, yaitu wokwi.toml dan diagram.json. Setelah itu copy relative path dari file firmware.elf dan firmware.bin disalin dan dimasukkan ke dalam wokwi.toml agar sistem dapat dijalankan dengan benar dalam simulator. Langkah terakhir adalah melakukan konfigurasi kode pada file main.cpp untuk mengontrol nyala LED sesuai dengan urutan yang telah ditentukan dalam sistem Traffic Light.

Code Program

#include <Arduino.h>

void setup() {

    Serial.begin(115200);

    pinMode(33, OUTPUT);

    pinMode(26, OUTPUT);

    pinMode(25, OUTPUT);

    Serial.println("");

    Serial.println("Hello, rekk");

    Serial.println("Welcome to traffic light :-)");

}

void loop() {

    Serial.println("Red");

    digitalWrite(33, HIGH);

    delay(1000);

    digitalWrite(33, LOW);

    Serial.println("Orange");

    digitalWrite(25, HIGH);

    delay(1000);

    digitalWrite(25, LOW);

    Serial.println("Green");

    digitalWrite(26, HIGH);

    delay(1000);

    digitalWrite(26, LOW);

}

# **BAB III**

## **Results and Discussion**

**3.1 Experimental Results**



