**LAPORAN PRAKTIKUM**

**IOT MINGGU KE-2**

**Pengembangan Sistem Kontrol Lampu Lalu Lintas**

**Menggunakan Tombol dan LED**

****

**Dosen Pengampu :**

**Ir. Subairi, ST., MT., IPM**

**Disusun Oleh:**

**Qaila Salsabila**

**(233140707111089)**

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS VOKASI**

**UNIVERSITAS BRAWIJAYA**

**2025**

**Abstrak**

Lampu lalu lintas sangat penting untuk mengatur arus antara kendaraan dan pejalan kaki, yang meningkatkan keselamatan dan efisiensi lalu lintas. Menggunakan mikrokontroler ESP32 yang diprogram melalui IDE Arduino, penelitian ini mengembangkan sistem kontrol lampu lalu lintas sederhana yang menggunakan tombol dan LED. Ini dimaksudkan untuk mensimulasikan pengaturan lalu lintas secara manual dengan menyalakan LED merah, kuning, dan hijau berdasarkan input tombol. Hasil eksperimen menunjukkan bahwa sistem berjalan dengan baik dan memiliki urutan nyala LED yang tepat dan responsif terhadap tombol. Dengan resistor pull-down yang tepat, masalah seperti koneksi breadboard yang tidak stabil dapat diselesaikan. Eksperimen ini tidak hanya menunjukkan dasar-dasar integrasi perangkat lunak dan perangkat keras dalam sistem kontrol yang mudah digunakan, tetapi juga memberikan manfaat pendidikan untuk memahami pemrograman mikrokontroler dan logika kontrol.

*Keywords - Lampu lalu lintas, sistem kontrol, tombol, LED, mikrokontroler ESP32, Arduino IDE, pengaturan lalu lintas, integrasi perangkat keras, perangkat lunak, logika kontrol, pemrograman mikrokontroler, edukasi.*

**BAB I**

**PENDAHULUAN**

**1.1 Latar Belakang**

Salah satu bagian penting dari manajemen lalu lintas di jalan raya adalah sistem kontrol lampu lalu lintas. Lampu lalu lintas mengatur arus antara kendaraan dan pejalan kaki, mengurangi risiko kecelakaan, dan meningkatkan efisiensi penggunaan jalan. Dengan semakin meningkatnya kebutuhan akan sistem pengaturan lalu lintas yang fleksibel, pengembangan solusi sederhana menggunakan tombol dan LED dapat menjadi pendekatan yang menarik untuk berbagai situasi. Teknologi berbasis tombol yang digunakan untuk mengatur pola nyala lampu lalu lintas memungkinkan Anda untuk mengatur durasi dan urutan nyala merah, kuning, dan hijau. Sistem ini dapat digunakan untuk mengajarkan siswa dasar-dasar pengaturan lalu lintas atau untuk proyek kecil yang memerlukan pengaturan lalu lintas sementara di lingkungan tertentu. Dengan menggunakan tombol manual, pengguna dapat belajar dasar-dasar elektronika dan pemrograman mikrokontroler dengan cara yang interaktif dan mudah dipahami.

Pendidikan teknik elektro dan informatika juga memerlukan pengembangan sistem ini. Sistem ini menyediakan model pembelajaran praktis untuk memahami logika kontrol dasar melalui penggunaan tombol dan LED. Hal ini dapat membantu siswa atau pemula dalam mempelajari konsep seperti logika berurutan, pengendalian input dan output, dan penerapan perangkat lunak dalam pengelolaan perangkat keras. Proyek ini, meskipun sederhana, dapat berguna sebagai langkah awal untuk memahami pengembangan sistem kontrol yang lebih kompleks di masa depan.

**1.2 Tujuan Eksperimen**

1. Mengembangkan sistem lampu lalu lintas berbasis tombol yang dapat mengontrol pola nyala LED merah, kuning, dan hijau secara bergantian. Ini akan menggambarkan cara mengontrol lalu lintas secara manual.
2. Memberikan pemahaman praktis tentang cara tombol berfungsi sebagai input dan LED berfungsi sebagai output, serta cara pemrograman mengatur pola nyala lampu.
3. Mempelajari konsep dasar pengendalian perangkat keras melalui perangkat lunak dan memperoleh keterampilan dalam merancang dan menerapkan logika kontrol sederhana dengan mikrokontroler.

**BAB II**

**METODOLOGI**

**2.1 Alat dan Bahan**

* ESP32
* LED Merah, Kuning, Hijau
* 3 Resistor
* Software Arduino IDE
* PushButton

**2.3 Implementasi Sistem**

1. Mengikuti diagram rangkaian, hubungkan komponen seperti mikrokontroler, resistor, tombol, dan LED ke breadboard.
2. mengunggah kode ke mikrokontroler sehingga dapat mengatur pola nyala LED berdasarkan tombol yang dimasukkan.
3. Untuk memastikan pola nyala LED sesuai dengan urutan lampu lalu lintas, uji sistem dengan menekan tombol.
4. Perbaiki rangkaian atau kode jika terjadi kesalahan hingga sistem beroperasi dengan baik.

**BAB III**

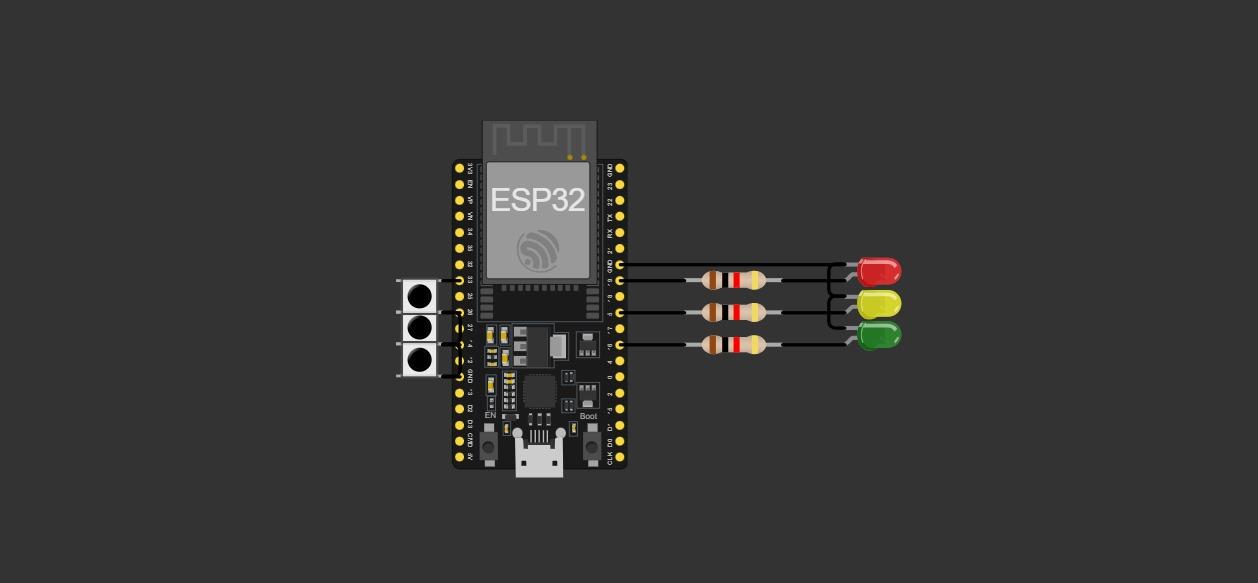
**HASIL DAN PEMBAHASAN**

**3.1 Hasil Eksperimen**

Hasil eksperimen menunjukkan bahwa sistem kontrol lampu lalu lintas konvensional yang menggunakan tombol dan LED bekerja dengan baik. Ketika Anda menekan tombol, LED merah, kuning, dan hijau menyala secara berurutan sesuai dengan pilihan program. Sebuah respons yang baik dari sistem menunjukkan bahwa rangkaian dan logika program telah diterapkan dengan benar.

Selama pengujian, beberapa masalah ditemukan, seperti respons tombol yang tidak konsisten karena koneksi yang tidak stabil ke breadboard. Menggunakan resistor pull-down yang sesuai atau memperbaiki sambungan dapat menyelesaikan masalah ini. Secara keseluruhan, eksperimen ini berhasil menunjukkan ide-ide dasar tentang sistem kontrol lampu lalu lintas dan menawarkan pemahaman praktis tentang integrasi perangkat keras dan perangkat lunak untuk proyek yang lebih sederhana.

**Dokumentasi eksperimen meliputi screenshoot simulasi ESP32 :**

****

**Lampiran**

#include <Arduino.h>

int button1 = 33;

int button2 = 26;

int button3 = 14;

int red = 19;

int yellow = 5;

int green = 16;

void kedipred(int jumlahKedip);

void kedipredgreen(int jumlahKedip);

void kedipredyellowgreen(int jumlahKedip);

void setup() {

pinMode(red, OUTPUT);

pinMode(yellow, OUTPUT);

pinMode(green, OUTPUT);

pinMode(button1, INPUT\_PULLUP);

pinMode(button2, INPUT\_PULLUP);

pinMode(button3, INPUT\_PULLUP);

Serial.begin(115200);

}

void loop() {

if (digitalRead(button1) == LOW) {

Serial.println("Button 1");

kedipred(5);

}

else if (digitalRead(button2) == LOW) {

kedipredgreen(5);

}

else if (digitalRead(button3) == LOW) {

kedipredyellowgreen(5);

}

}

void kedipred(int jumlahKedip) {

for (int i = 0; i < jumlahKedip; i++) {

digitalWrite(red, HIGH);

delay(500);

digitalWrite(red, LOW);

delay(500);

}

}

void kedipredgreen(int jumlahKedip) {

for (int i = 0; i < jumlahKedip; i++) {

digitalWrite(red, HIGH);

digitalWrite(green, LOW);

delay(500);

digitalWrite(red, LOW);

digitalWrite(green, HIGH);

delay(500);

}

}

void kedipredyellowgreen(int jumlahKedip) {

for (int i = 0; i < jumlahKedip; i++) {

digitalWrite(red, HIGH);

digitalWrite(yellow, LOW);

digitalWrite(green, LOW);

delay(500);

digitalWrite(red, LOW);

digitalWrite(yellow, HIGH);

delay(500);

digitalWrite(yellow, LOW);

digitalWrite(green, HIGH);

delay(500);

digitalWrite(green, LOW);

}

}