

**SISTEM PENGATURAN LAMPU LALU LINTAS BERDASARKAN
JUMLAH KENDARAAN YANG MELINTAS MENGGUNAKAN
METODE ALGORITMA PROPOSIONAL**

TUGAS AKHIR

Diajukan Oleh:

ALIM SYAHYUJAR

190705035

Mahasiswa Fakultas Sains dan Teknologi

Program Studi Teknologi Informasi



**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI AR-RANIRY**

BANDA ACEH

2025/1446

LEMBAR PERSETUJUAN

**SISTEM PENGATURAN LAMPU LALU LINTAS BERDASARKAN
JUMLAH KENDARAAN YANG MELINTAS MENGGUNAKAN METODE
ALGORITMA PROPOSIONAL**

TUGAS AKHIR

Diajukan Kepada Fakultas Sains dan Teknologi
Universitas Islam Negeri (UIN) Ar-Raniry Banda Aceh
Sebagai Salah Satu Beban Studi Memperoleh Gelar Sarjana (S1)
dalam Ilmu Teknologi Informasi

Oleh:

ALIM SYHAYUJAR
NIM.190705035

Mahasiswa Fakultas Sains dan Teknologi
Program Studi Teknologi Informasi

Disetujui untuk Dimunqasyahkan Oleh:

Pembimbing I, **جامعة الرانيري**

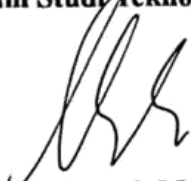
Pembimbing II,

A R - R A N I R Y


Dr. Hendri Ahmadian, S.Si., M.I.M
NIP. 198301042014031002


Ridha Ilahi, M.T
NIP. 197905302014031001

Mengetahui,
Ketua Program Studi Teknologi Informasi


Malahayati, M.T
NIP. 198301272015032003

LEMBAR PENGESAHAN
SISTEM PENGATURAN LAMPU LALU LINTAS BERDASARKAN
JUMLAH KENDARAAN YANG MELINTAS MENGGUNAKAN METODE
ALGORITMA PROPOSIONAL

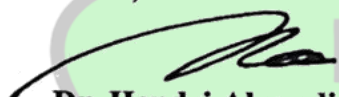
TUGAS AKHIR

Telah Diuji Oleh Panitia Ujian Munaqasyah Tugas Akhir
Fakultas Sains dan Teknologi UIN Ar-Raniry Banda Aceh dan Dinyatakan Lulus
Serta Diterima Sebagai Salah Satu Beban Studi Program Sarjana (S-1)
Dalam Program Studi Teknologi Informasi

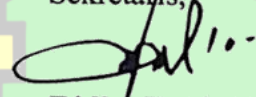
Pada Hari/Tanggal: Rabu, 20 Agustus 2025
26 Safar 1447 H

Panitia Ujian Munaqasyah Tugas Akhir:

Ketua,


Dr. Hendri Ahmadian, S. Si., M.I.M
NIP. 198301042014031002

Sekretaris,


Ridha Ilahi, M.T
NIP. 197905302014031001

Penguji I,


Malahayati, M.T
NIP. 198301272015032003

Penguji II,


Nurrizqa, S.Pd., M.T
NIP. 199704302025052001

Mengetahui:

Dekan Fakultas Sains dan Teknologi
UIN Ar-Raniry Banda Aceh,




Dr. Muhammad Dirhamsyah, M.T., IPU
NIP. 19620021988111001

LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Alim Syahyujar
NIM : 190705035
Program Studi : Teknologi Informasi
Fakultas : Sains dan Teknologi
Judul : SISTEM PENGATURAN LAMPU LALU LINTAS
BERDASARKAN JUMLAH KENDARAAN YANG
MELINTAS MENGGUNAKAN METODE
ALGORITMA PROPOSIONAL

Dengan ini menyatakan bahwa dalam penulisan tugas akhir ini, saya:

1. Tidak menggunakan ide orang lain tanpa mampu mengembangkan dan mempertanggungjawabkan;
2. Tidak melakukan plagiasi terhadap naskah karya orang lain;
3. Tidak menggunakan karya orang lain tanpa menyebutkan sumber asli atau tanpa izin pemilik karya;
4. Tidak memanipulasi dan memalsukan data;
5. Mengerjakan sendiri karya ini dan mampu bertanggungjawab atas karya ini.

Bila dikemudian hari ada tuntutan dari pihak lain atas karya saya, dan telah melalui pembuktian yang dapat dipertanggungjawabkan dan ternyata memang ditemukan bukti bahwa saya telah melanggar pernyataan ini, maka saya siap dikenai sanksi berdasarkan aturan yang berlaku di Fakultas Sains dan Teknologi UIN Ar-Raniry Banda Aceh.

Demikian pernyataan ini saya buat dengan sesungguhnya dan tanpa paksaan dari pihak manapun.

Banda Aceh,
Yang Menyatakan



ABSTRAK

Penelitian ini bertujuan untuk merancang sistem pengaturan lampu lalu lintas adaptif berbasis logika proposional pada persimpangan jalan. Sistem ini dirancang untuk menyesuaikan waktu lampu lalu lintas secara otomatis berdasarkan jumlah kendaraan yang melintas, dengan tujuan meningkatkan efisiensi arus lalu lintas dan mengurangi potensi kemacetan. Proses pengembangan sistem mencakup tahapan perancangan perangkat keras, pemrograman mikrokontroler menggunakan Arduino IDE, serta implementasi logika pengambilan keputusan berbasis algoritma proposional. Sensor inframerah digunakan untuk mendeteksi jumlah kendaraan di setiap jalur, dan data tersebut digunakan sebagai input dalam penentuan durasi lampu hijau. Sistem akan memberikan prioritas waktu lebih lama kepada jalur dengan kepadatan kendaraan yang lebih tinggi. Algoritma yang digunakan dirancang untuk mengevaluasi kondisi lalu lintas secara real-time dan mengatur siklus lampu lalu lintas secara efisien. Sistem ini tidak hanya menawarkan pendekatan yang adaptif dan efisien, tetapi juga fleksibel serta dapat diintegrasikan dengan konsep kota cerdas (*smart city*). Dengan pendekatan ini, sistem diharapkan mampu menciptakan manajemen lalu lintas yang lebih baik, mengurangi waktu tunggu pengendara, dan memberikan solusi inovatif terhadap permasalahan lalu lintas di wilayah perkotaan. Penelitian ini sekaligus mendukung pengembangan teknologi informasi yang aplikatif dan bermanfaat langsung bagi masyarakat luas.

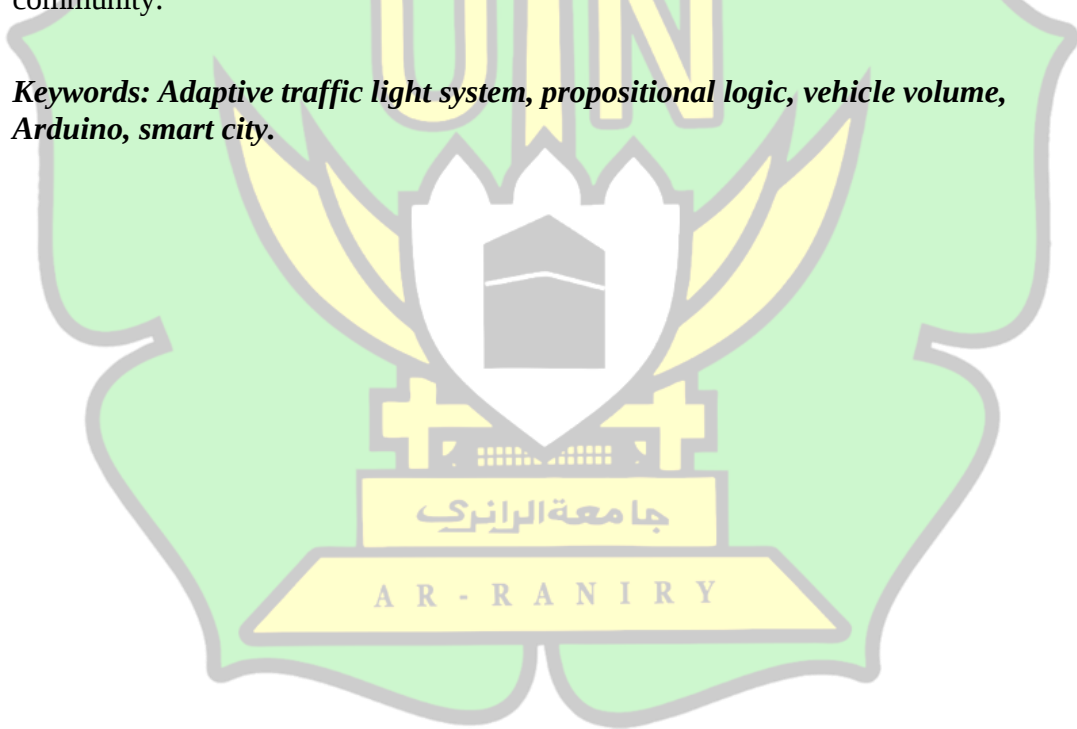
Kata kunci: *Sistem lampu lalu lintas adaptif, logika proposional, jumlah kendaraan, Arduino, smart city.*



ABSTRACT

This research aims to design an adaptive traffic light control system based on propositional logic for road intersections. The system is designed to automatically adjust traffic light timings based on vehicle volume, with the goal of improving traffic flow efficiency and reducing potential congestion. The system development process includes hardware design, microcontroller programming using Arduino IDE, and implementation of a decision-making logic based on a propositional algorithm. Infrared sensors are utilized to detect the number of vehicles in each lane, and this data serves as input for determining the green light duration. The system will prioritize longer green light times for lanes with higher vehicle density. The algorithm used is designed to evaluate traffic conditions in real-time and manage the traffic light cycle efficiently. This system not only offers an adaptive and efficient approach but is also flexible and can be integrated with the smart city concept. With this approach, the system is expected to create better traffic management, reduce driver waiting times, and provide innovative solutions to urban traffic problems. This research also supports the development of information technology that is applicable and directly beneficial to the wider community.

Keywords: *Adaptive traffic light system, propositional logic, vehicle volume, Arduino, smart city.*



KATA PENGANTAR

Alhamdulillah, Puji dan Syukur kita panjatkan kepada Allah Subhanahu Wata'ala. Dzat yang hanya kepada-Nya memohon pertolongan. Alhamdulillah atas segala pertolongan, rahmat, dan kasih sayang-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Sistem Pengaturan Lampu Lalu Lintas Berdasarkan Jumlah Kendaraan Yang Melintas Menggunakan Metode Algoritma Proposional”. Shalawat dan salam kepada Rasulullah Shallallahu Alaihi Wasallam yang senantiasa menjadi sumber inspirasi dan teladan terbaik untuk umat manusia.

Skripsi ini dibuat untuk memenuhi tugas akhir perkuliahan dan sebagai salah satu persyaratan untuk memperoleh gelar Sarjana Strata 1 di Program Studi Teknologi Informasi Fakultas Sains Dan Teknologi Universitas Islam Negeri Ar-Raniry Banda Aceh. Selain itu, skripsi ini juga dibuat sebagai salah satu wujud implementasi dari ilmu yang didapatkan selama masa perkuliahan di Program Studi Teknologi Informasi Fakultas Sains Dan Teknologi Universitas Islam Negeri Ar-Raniry Banda Aceh. Selama Menyusun skripsi ini penulis menyadari bahwa skripsi masih jauh dari sempurna. Oleh karena itu, penulis berharap dapat belajar lebih banyak lagi dalam mengimplementasikan ilmu yang didapatkan. Skripsi ini tentunya tidak lepas dari bimbingan, masukan, dan arahan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Orang tua tercinta Ayahanda Syarkani, S.Ag dan Ibunda Nur Aisah Nasution, S.Pd. Terima kasih pada setiap rupiah yang sudah dikeluarkan untuk menunjang pendidikan penulis, terima kasih pada setiap detik waktu yang sudah diluangkan untuk membesarkan penulis dan terima kasih pada setiap keringat yang orang tua keluarkan untuk memastikan penulis dapat hidup dengan layak, dan terima kasih pada setiap suara yang dikeluarkan untuk memberikan nasehat, masukan, dukungan, motivasi, doa serta telah mendidik dan membesarkan penulis dalam limpahan kasih sayang. Terima kasih atas apa yang telah diberikan kepada penulis tidak bisa dibandingkan dengan apapun.
2. Bapak Prof. Dr. Mujiburrahman, M. Ag selaku Rektor Universitas Islam Negeri Ar-Raniry.

3. Bapak Dr. Ir. Muhammad Dirhamsyah, MT, IPU selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Ar-Raniry.
4. Ibuk Malahayati, M. T Selaku Ketua Program Studi Teknolog Informasi Fakultas Sains Dan Teknologi Universitas Islam Negeri Ar-Raniry.
5. Bapak Khairan Selaku Sekretaris Program Studi Teknologi Informasi Fakultas Sains Dan Teknologi Universitas Islam Negeri Ar-Raniry.
6. Bapak Hendri Ahmadian, M.I.M selaku dosen pembimbing yang telah membantu penulis dalam penulisan skripsi ini dengan meluangkan waktu serta memberikan bimbingan, dan arahan sehingga penulis dapat menyelesaikan skripsi ini.
7. Ridha Ilahi, S.Kom., M.T., M.T. selaku dosen pembimbing kedua yang telah membantu penulis dalam menyelesaikan penulisan skripsi ini dengan meluangkan waktu serta arahan dan bimbingan serta ilmu baru untuk penulis sehingga dapat menyelesaikan skripsi ini.
8. Seluruh keluarga besar penulis yang telah mendoakan, memotivasi, dan memberikan dukungan untuk kemajuan penulis.
9. Seluruh dosen pengajar pada Program Studi Teknologi Informasi Fakultas Sains Dan Teknologi Universitas Islam Negeri Ar-Raniry atas bimbingan dan ilmu – ilmu yang penulis dapat selama masa perkuliahan.
10. Staf Program Studi Teknologi Informasi Buk Cut Ida yang selalu memberikan semangat serta mempermudah urusan penulis dalam melalui setiap proses proses dan prosedur yang harus penulis lalui.
11. Para sahabat yang telah menemani penulis dalam menyelesaikan skripsi.
12. Serta semua teman dan pihak yang tidak dapat penulis sebutkan satu persatu.

Penulis sangat menyadari bahwa dalam penulisan skripsi ini masih banyak kekurangan serta jauh dari kata sempurna yang disebabkan oleh kemampuan penulis miliki. Untuk itu penulis mengharapkan kritik dan saran yang bermanfaat demi memperbaiki diri dimasa mendatang. Demikian penulis sangat berharap agar skripsi ini dapat memberikan manfaat bagi seluruh pihak yang berkepentingan.

Banda Aceh, 14 Agustus 2025

Penulis



Alim Syahyujar
190705035

URAIAN SINGKAT

Sistem yang dirancang dalam skripsi ini merupakan sistem pengaturan lampu lalu lintas adaptif yang mampu menyesuaikan waktu nyala lampu berdasarkan jumlah kendaraan yang melintas. Sistem ini dibangun dengan pendekatan logika proposional sebagai dasar pengambilan keputusan, serta memanfaatkan teknologi mikrokontroler dan sensor pendeteksi kendaraan untuk mengumpulkan data lalu lintas secara real-time. Sistem ini merepresentasikan penerapan nyata dari prinsip teknologi cerdas dalam mengelola permasalahan lalu lintas perkotaan.

Desain sistem mencakup komponen-komponen utama seperti Arduino Mega, sensor inframerah, dan tampilan LCD yang berfungsi untuk menampilkan status lalu lintas dan jumlah kendaraan. Logika proposional digunakan untuk menentukan durasi lampu hijau berdasarkan tingkat kepadatan masing-masing jalur. Sistem ini tidak hanya berfungsi sebagai solusi teknis terhadap kemacetan, tetapi juga sebagai media pembelajaran dan inovasi dalam bidang teknologi informasi.

Diharapkan sistem ini dapat meningkatkan efisiensi lalu lintas, mengurangi waktu tunggu kendaraan, serta berkontribusi dalam pengembangan kota cerdas (*smart city*). Implementasi yang terstruktur dan adaptif, sistem ini memiliki potensi untuk digunakan dalam berbagai skenario lalu lintas nyata, serta menjadi rujukan dalam pengembangan sistem transportasi berbasis teknologi lokal dan cerdas.

جامعة الرانري

A R - R A N I R Y

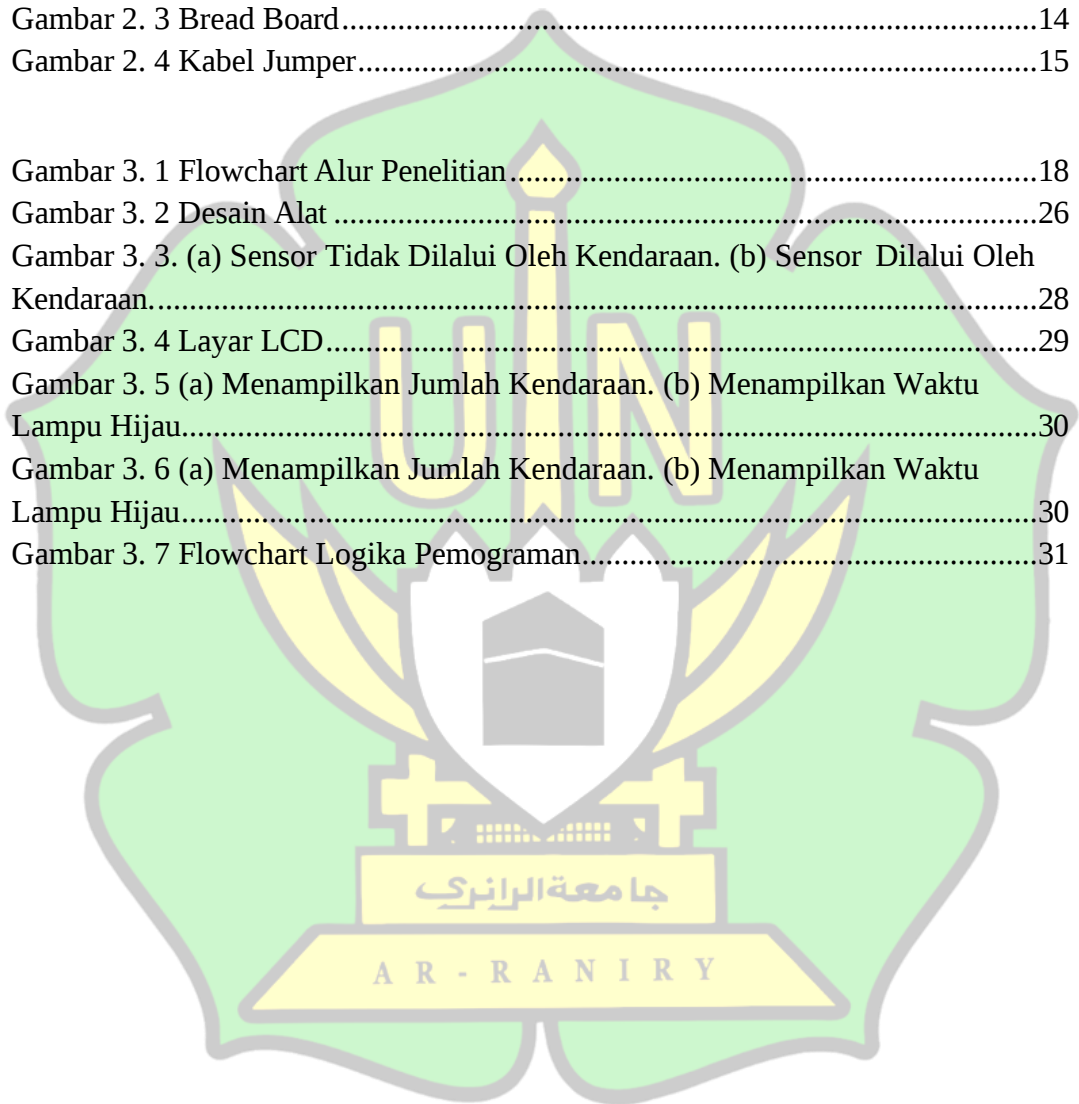
DAFTAR ISI

LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR	iv
ABSTRAK	v
ABSTRACT	vi
KATA PENGANTAR.....	vii
URAIAN SINGKAT	x
DAFTAR ISI.....	xi
DAFTAR GAMBAR.....	xiii
DAFTAR TABEL.....	xiv
BAB I	1
PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan Penelitian.....	3
1.4. Batasan Masalah.....	3
1.5. Manfaat Penelitian.....	4
BAB II.....	5
TINJAUAN PUSTAKA	5
2.1. Penelitian Terdahulu.....	5
2.2. Arduino Ide.....	7
2.3. Arduino Mega.....	9
2.2.1. Daya (Power) Daya	9
2.2.2. Memori	10
2.2.3. Input dan Output.....	10
2.2.4. Komunikasi	10
2.2.5. Reset Otomatis	11
2.2.6. Karakteristik Fisik	12
2.4. Mikrokontroler	12
2.5. Sensor Inframerah	13

2.6.	Bread Board.....	14
2.7.	Resistor.....	14
2.8.	Kabel Jumper.....	14
2.9.	Algoritma Proposional	15
BAB III.....		18
METODE PENELITIAN		18
3.1.	Alur Penelitian.....	18
3.1.1.	Analisis Masalah	19
3.1.2.	Studi Literatur & Riset Algoritma	19
3.1.3.	Perancangan Sistem (Desain Rangkaian dan Algoritma)	19
3.1.4.	Perakitan Alat (Sensor, Arduino, LCD, LED)	20
3.1.5.	Pemrograman Logika Proposional (Arduino IDE).....	20
3.1.6.	Simulasi & Pengujian Sistem.....	20
3.1.7.	Analisis Kinerja Sistem.....	20
3.1.8.	Evaluasi & Dokumentasi.....	21
3.2.	Jadwal Kegiatan	21
BAB IV		23
HASIL DAN PEMBAHASAN		23
4.1.	Langkah Kerja Implementasi Sistem	23
4.1.1.	Persiapan Alat Dan Bahan	23
4.1.2.	Rangkaian Sistem.....	26
4.1.3.	Prinsip Kerja Sistem.....	28
4.1.4.	Logika Pemrograman	31
4.2.	Uji Coba Dan Evaluasi	32
4.3.	Hasil Produk.....	34
BAB 5.....		35
KELEBIHAN DAN KEKURANGAN		35
5.1.	Kelebihan.....	35
5.2.	Kekurangan.....	36
DAFTAR PUSTAKA		37
LAMPIRAN		39

DAFTAR GAMBAR

Gambar 2. 1 Arduino Mega.....	9
Gambar 2. 2 Sensor Inframerah.....	13
Gambar 2. 3 Bread Board.....	14
Gambar 2. 4 Kabel Jumper.....	15
 Gambar 3. 1 Flowchart Alur Penelitian.....	 18
Gambar 3. 2 Desain Alat	26
Gambar 3. 3. (a) Sensor Tidak Dilalui Oleh Kendaraan. (b) Sensor Dilalui Oleh Kendaraan.....	28
Gambar 3. 4 Layar LCD.....	29
Gambar 3. 5 (a) Menampilkan Jumlah Kendaraan. (b) Menampilkan Waktu Lampu Hijau.....	30
Gambar 3. 6 (a) Menampilkan Jumlah Kendaraan. (b) Menampilkan Waktu Lampu Hijau.....	30
Gambar 3. 7 Flowchart Logika Pemograman.....	31



DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu.....	5
Tabel 3. 1 Bahan Penelitian.....	24
Tabel 3. 2 Koneksi PIN	27
Tabel 3. 3 Hasil Uji.....	33



BAB I

PENDAHULUAN

1.1. Latar Belakang

Kemacetan lalu lintas merupakan tantangan besar bagi kota-kota metropolitan di Indonesia. Peningkatan jumlah kendaraan yang tidak sejalan dengan kapasitas infrastruktur jalan, ditambah dengan sistem pengaturan lalu lintas yang kurang responsif, menjadi penyebab utama masalah ini. Lampu lalu lintas, sebagai elemen vital dalam sistem transportasi perkotaan, memiliki peran krusial dalam mengatur arus kendaraan di persimpangan demi ketertiban dan keamanan. Namun, sebagian besar sistem yang ada masih mengandalkan pengaturan waktu yang statis (*fixed-time*), sehingga kurang mampu beradaptasi dengan perubahan volume kendaraan. Kondisi ini seringkali berakibat pada pembagian durasi lampu hijau dan merah yang tidak efisien, menyebabkan antrean panjang di beberapa ruas jalan dan menghambat kelancaran pergerakan lalu lintas.

Seiring kemajuan teknologi, berbagai inovasi telah dikembangkan untuk menyempurnakan manajemen lampu lalu lintas. Salah satu pendekatan yang semakin populer adalah sistem adaptif, yang secara otomatis menyesuaikan durasi lampu berdasarkan kondisi lalu lintas terkini. Metode ini memungkinkan pengelolaan lalu lintas yang lebih responsif dan dinamis, berkontribusi pada pengurangan tingkat kepadatan jalan dan peningkatan efisiensi perjalanan. Dalam studi ini, kami mengusulkan penerapan Algoritma Proporsional, sebuah pendekatan yang mengatur durasi lampu berdasarkan jumlah kendaraan yang sedang melintas.

Prinsip kerja Algoritma Proporsional ini adalah memberikan durasi lampu hijau yang lebih panjang pada jalur dengan volume kendaraan yang lebih tinggi. Dengan cara ini, alokasi waktu lampu lalu lintas dapat dioptimalkan sesuai dengan kepadatan riil di setiap jalur persimpangan. Pendekatan ini kontras dengan metode konvensional yang mengabaikan kondisi lalu lintas langsung. Implementasi sistem yang didasarkan pada algoritma ini diharapkan dapat meningkatkan efisiensi lalu

lintas secara signifikan, seperti mempersingkat waktu tunggu pengendara, memperlancar pergerakan kendaraan, dan meminimalkan antrean panjang, khususnya di area padat.

Berbagai riset telah dilakukan untuk mengembangkan sistem pengaturan lampu lalu lintas yang lebih efektif. Sebagai contoh, penelitian yang menggunakan logika fuzzy menunjukkan potensi sistem adaptif dalam mengurangi kemacetan melalui deteksi kepadatan kendaraan secara waktu nyata. Selain itu, metode berbasis kecerdasan buatan (AI), termasuk jaringan saraf tiruan dan algoritma genetika, juga telah diterapkan untuk meningkatkan efisiensi pengaturan lalu lintas. Akan tetapi, penggunaan Algoritma Proporsional secara spesifik dalam sistem pengaturan lampu lalu lintas belum banyak dieksplorasi, menjadikan penelitian ini penting untuk dikembangkan lebih lanjut.

Selanjutnya, sistem yang diusulkan akan memanfaatkan sensor untuk mengumpulkan data kepadatan lalu lintas. Informasi ini kemudian akan diproses oleh mikrokontroler seperti Arduino, yang akan mengatur durasi lampu lalu lintas secara proporsional sesuai dengan data volume kendaraan. Melalui pendekatan ini, kami berharap sistem yang dikembangkan akan menjadi lebih adaptif dan fleksibel dalam merespons dinamika kondisi lalu lintas yang terus berubah.

Berdasarkan pemaparan di atas, penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem pengaturan lampu lalu lintas berbasis Algoritma Proporsional yang mampu beradaptasi dengan volume kendaraan. Penelitian ini tidak hanya akan memperkaya literatur di bidang Sistem Transportasi Cerdas (ITS), tetapi juga berpotensi menjadi solusi efektif dalam mengatasi masalah kemacetan di berbagai wilayah perkotaan yang menghadapi isu serupa.

1.2. Rumusan Masalah

1. Bagaimana cara mendeteksi jumlah kendaraan yang melintas di setiap jalur dengan menggunakan sensor berbasis Arduino?

2. Bagaimana merancang sistem pengaturan lampu lalu lintas berbasis Algoritma Proporsional yang dapat beradaptasi dengan jumlah kendaraan yang melintas secara real-time?

1.3. Tujuan Penelitian

1. Meneliti dan merancang sistem berbasis Arduino yang mampu mendeteksi jumlah kendaraan yang melintas di setiap jalur menggunakan sensor yang sesuai,
2. Mengembangkan sistem pengaturan lampu lalu lintas berbasis Algoritma Proporsional yang dapat menyesuaikan durasi lampu berdasarkan jumlah kendaraan di setiap jalur secara real-time.

1.4. Batasan Masalah

Agar penelitian ini lebih terarah dan fokus, maka terdapat batasan masalah yang diterapkan mencakup:

1. Cakupan area pengujian pada penelitian ini hanya berfokus pada sistem pengaturan lampu lalu lintas di satu simpang jalan dengan maksimal empat persimpangan. Implementasi dilakukan dalam lingkungan simulasi tanpa penerapan langsung di kondisi lalu lintas nyata.
2. Parameter deteksi kendaraan dimana Sistem hanya mempertimbangkan jumlah kendaraan yang melintas sebagai variabel utama dalam penentuan durasi lampu lalu lintas. Faktor eksternal seperti kondisi cuaca, jenis kendaraan, dan perilaku pengemudi tidak menjadi bagian dari analisis.
3. Metode pengaturan lalu lintas menggunakan perangkat lunak (software) mencakup pemrograman pada Arduino IDE dengan bahasa C++, serta penggunaan Algoritma Proporsional untuk pengolahan data lalu lintas.

1.5. Manfaat Penelitian

Adapun manfaat penelitian ini dapat dikategorikan secara teoritis dan praktis, sebagai berikut ini:

1. Secara Teoritis: Penelitian ini memberikan kontribusi terhadap pengembangan ilmu pengetahuan dalam bidang sistem kendali lalu lintas dengan:
 - Menambah referensi mengenai penerapan metode Algoritma Proporsional dalam sistem pengaturan lampu lalu lintas berbasis jumlah kendaraan yang melintas.
 - Memberikan pemahaman lebih lanjut tentang implementasi sensor berbasis Arduino dalam sistem deteksi kendaraan secara real-time.
 - Menjadi dasar bagi penelitian lanjutan dalam optimasi sistem lalu lintas berbasis kecerdasan buatan atau metode adaptif lainnya.
2. Secara Praktis: Penelitian ini diharapkan dapat memberikan manfaat langsung dalam penerapan sistem pengaturan lalu lintas yang lebih efisien, yaitu:
 - Mengurangi kemacetan dengan menyesuaikan durasi lampu lalu lintas berdasarkan kepadatan kendaraan di setiap jalur.
 - Meningkatkan efisiensi lalu lintas, sehingga waktu tempuh kendaraan menjadi lebih singkat dan konsumsi bahan bakar dapat ditekan.
 - Membantu pemerintah dan instansi terkait dalam merancang sistem lalu lintas yang lebih adaptif.
 - Menjadi alternatif solusi bagi daerah dengan tingkat kepadatan lalu lintas tinggi yang masih menggunakan sistem lampu lalu lintas actuated.

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Pada Penelitian ini penulis membutuhkan referensi untuk dijadikan panduan agar terhindar dari plagiasi dan duplikasi. penelitian terdahulu berfungsi sebagai bahan perbandingan dan patokan serta untuk membedakan dengan penelitian penulis saat ini. Jadi, sebagai dasar teori, penulis mencantumkan temuan penelitian sebelumnya sebagai berikut :

Tabel 2. 1 Penelitian Terdahulu

No	Referensi	Latar Belakang	Hasil
1	J.P. Manalu (2023)	Sistem lampu hijau belum disesuaikan dengan jenis kendaraan yang melintas.	Studi kasus implementasi fuzzy logic untuk kontrol durasi lampu sesuai jenis kendaraan.
2	N. Saputra (2023)	Sistem transportasi membutuhkan optimalisasi berbasis AI dan logika formal.	Sistem transportasi membutuhkan optimalisasi berbasis AI dan logika formal.
3	Shamsi & Kenari (2022)	Algoritma kecerdasan buatan dapat meningkatkan efisiensi pengaturan lalu lintas.	Menggunakan Reinforcement Learning untuk mengoptimalkan sistem lampu lalu lintas dan memprioritaskan kendaraan darurat.

No	Referensi	Latar Belakang	Hasil
4	Abubakar et al. (2023)	Sistem lampu lalu lintas berbasis kepadatan perlu dikembangkan lebih lanjut untuk meningkatkan efisiensi lalu lintas.	Mengembangkan sistem lampu lalu lintas berbasis kepadatan dengan pendekatan sederhana dan biaya rendah.
5	Hartati, (2021)	Waktu nyala lampu lalu lintas tidak mempertimbangkan jumlah kendaraan aktual, menyebabkan ketidakefisienan di simpang.	Menerapkan fuzzy logic berbasis proposisi IF-THEN untuk mengatur durasi lampu hijau berdasarkan jumlah kendaraan.
6	Rahmat Alimuddin, (2022)	Area blindspot truk sering menyebabkan kecelakaan karena sistem monitoring tidak akurat.	Mengembangkan sistem pengambilan keputusan berbasis logika fuzzy dan proposional untuk deteksi objek di area blindspot kendaraan berat.

Tinjauan terhadap beberapa studi sebelumnya menunjukkan bahwa penelitian J.P. Manalu (2023), yang membahas "Optimalisasi Waktu Nyala Lampu Hijau Menggunakan *Fuzzy Logic*", memiliki kemiripan paling dekat dengan topik skripsi ini. Studi Manalu mengeksplorasi adaptasi pengaturan waktu lampu lalu lintas berdasarkan kategori kendaraan yang melintas, seperti mobil, motor, atau kendaraan berat. Meskipun menggunakan pendekatan logika *fuzzy*, metode tersebut tetap mengandalkan struktur aturan berbasis proposisi logika dalam format pernyataan IF-THEN, yang sejalan dengan prinsip dasar algoritma proposional. Oleh karena itu, pola dan proses pengambilan keputusan dalam riset Manalu dapat dianggap relevan dan mendekati metodologi yang diusulkan dalam skripsi ini, yaitu

pengaturan lampu lalu lintas berdasarkan kuantitas kendaraan dengan pendekatan algoritma proposional.

Selain itu, penelitian oleh Hartati (2021) juga menunjukkan kedekatan konseptual dengan skripsi ini. Studi tersebut menggunakan logika *fuzzy* berbasis proposisi untuk menyesuaikan durasi lampu hijau berdasarkan jumlah kendaraan yang melewati persimpangan. Hal ini mengindikasikan bahwa pemanfaatan logika formal berbasis proposisi telah mulai diterapkan dalam sistem kontrol lalu lintas, meskipun terminologi "algoritma proposional" belum secara eksplisit digunakan dalam kerangka formal matematika atau logika komputasi klasik.

Dari kedua penelitian tersebut, dapat diidentifikasi beberapa persamaan signifikan. Keduanya sama-sama berupaya meningkatkan efisiensi lalu lintas dengan menyesuaikan durasi lampu lalu lintas berdasarkan kondisi lalu lintas aktual. Lebih lanjut, kedua studi menggunakan pendekatan berbasis logika formal, baik itu *fuzzy* maupun proposional, untuk merumuskan aturan pengambilan keputusan secara otomatis. Kesamaan ini menegaskan potensi besar pendekatan berbasis logika dalam pengembangan sistem lalu lintas cerdas.

Namun demikian, terdapat perbedaan penting. Studi Manalu lebih menitikberatkan pada pengelompokan jenis kendaraan, bukan pada penghitungan kuantitatif kendaraan seperti yang menjadi fokus utama skripsi ini. Sementara itu, penelitian Hartati memang berfokus pada jumlah kendaraan, tetapi masih mengadopsi logika *fuzzy*, bukan logika proposional murni dan eksplisit yang akan diterapkan dalam skripsi ini. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi yang lebih spesifik dalam mengembangkan sistem pengaturan lampu lalu lintas berdasarkan jumlah kendaraan dengan memanfaatkan struktur logika proposional murni, sebuah area yang belum banyak dibahas dalam penelitian sebelumnya.

2.2. Arduino Ide

Arduino IDE (*Integrated Development Environment*) adalah perangkat lunak yang digunakan untuk menulis, mengedit, mengompilasi, dan mengunggah kode program ke papan mikrokontroler Arduino. Aplikasi ini menyediakan

antarmuka pemrograman berbasis teks yang menggunakan bahasa C/C++ yang telah disederhanakan, sehingga mudah dipahami oleh pemula maupun pengguna tingkat lanjut. Arduino IDE bersifat open-source, dapat diunduh secara gratis, dan mendukung berbagai sistem operasi seperti Windows, Linux, dan macOS. Dalam pengembangannya, Arduino IDE juga telah dilengkapi dengan berbagai pustaka (*library*) untuk mendukung komunikasi dengan sensor, aktuator, serta perangkat eksternal lainnya.

Dalam konteks skripsi ini yang berjudul “*Sistem Pengaturan Lampu Lalu Lintas Berdasarkan Jumlah Kendaraan yang Melintas Menggunakan Metode Algoritma Proposional*”, Arduino IDE berperan sebagai platform utama untuk mengembangkan dan menguji program kendali lampu lalu lintas secara otomatis. Melalui Arduino IDE, logika proposional dapat dituliskan dalam bentuk pernyataan bersyarat (*conditional statement*) seperti IF-THEN, yang akan mengatur durasi nyala lampu berdasarkan jumlah kendaraan yang terdeteksi oleh sensor. Misalnya, jika jumlah kendaraan yang melintas melebihi ambang batas tertentu, maka durasi lampu hijau akan diperpanjang secara otomatis.

Selain itu, Arduino IDE juga digunakan untuk menghubungkan sistem dengan sensor kendaraan, seperti sensor ultrasonik atau inframerah, yang bertugas mendeteksi jumlah kendaraan pada masing-masing jalur. Data dari sensor ini kemudian diolah oleh mikrokontroler menggunakan algoritma yang telah diprogram melalui Arduino IDE. Hasil olahan ini selanjutnya akan digunakan untuk mengatur status lampu lalu lintas secara real-time. Arduino IDE juga memungkinkan pengguna untuk melakukan proses upload kode dan uji coba langsung ke papan mikrokontroler, sehingga proses *debugging* dan evaluasi sistem dapat dilakukan secara efisien. Dengan demikian, Arduino IDE menjadi komponen penting yang menghubungkan rancangan logika algoritma proposional dengan implementasi nyata dalam bentuk perangkat keras sistem lalu lintas otomatis.

2.3. Arduino Mega

Arduino merupakan sistem pengendalian berbasis mikro single-board yang bersifat *open-source*. Platform ini menggabungkan konsep wiring dan dirancang untuk memudahkan penggunaan perangkat elektronik di berbagai bidang. Dari sisi perangkat keras, Arduino menggunakan prosesor Atmel AVR, sementara perangkat lunaknya memiliki bahasa pemrograman khusus. Arduino sendiri termasuk dalam keluarga mikrokontroler ATmega yang dikembangkan oleh Atmel. Selain itu, terdapat berbagai jenis Arduino yang diproduksi oleh Perusahaan lain dengan mikrokontroler berbeda, namun tetap kompatibel dengan ekosistem perangkat keras Arduino. Fleksibilitasnya memungkinkan pengguna mengunggah program melalui bootloader, meskipun terdapat opsi lain seperti menggunakan downloader sebagai perantara untuk memprogram mikrokontroler langsung melalui port ISP.



Gambar 2. 1 Arduino Mega

2.2.1. Daya (Power) Daya

Arduino Mega 2560 dapat menerima pasokan daya melalui konektor sumber daya yang akan dipilih secara otomatis tergantung pada kondisi dan prioritas daya yang tersedia. Sumber daya eksternal (non-USB) dapat berasal dari adaptor AC ke DC atau baterai, yang keduanya dapat digunakan untuk memberi daya pada papan Arduino secara langsung. Adaptor daya dapat dihubungkan menggunakan plug center-positive ke jack daya (barrel jack) yang tersedia di papan Arduino Mega. Sedangkan kabel baterai dapat disambungkan ke papan melalui pin Vin dan Ground (GND) yang terdapat pada header konektor daya. Papan Arduino Mega dapat menangani suplai daya eksternal dalam rentang tegangan antara 6 hingga 20 Volt. Namun, jika tegangan suplai berada di bawah 7 Volt, pin 5V pada

papan kemungkinan besar hanya akan mengeluarkan tegangan di bawah 5V, yang dapat mengakibatkan Arduino Mega tidak berjalan dengan optimal. Sebaliknya, jika suplai tegangan melebihi 12 Volt, regulator tegangan pada papan dapat menjadi terlalu panas, berisiko menimbulkan kerusakan komponen dan ketidakstabilan sistem. Oleh karena itu, tegangan yang paling direkomendasikan untuk digunakan pada Arduino Mega adalah antara 7 hingga 12 Volt, guna memastikan kestabilan daya, kinerja regulator yang efisien, dan umur perangkat yang lebih panjang.

2.2.2. Memori

Pada perangkat jenis AT-Mega328 memiliki 32 kilobyet (dengan 0,5 KB untuk digunakan bootloader). Dan pada AT-mega 328 juga mempunyai 2 KB SRAM dan juga 1 KB EEPROM (yang bisa dibaca dan ditulis (RW atau read and written) dengan EEPROM library).

2.2.3. Input dan Output

Pada papan Arduino Mega 2560, terdapat 54 pin digital yang masing-masing dapat dikonfigurasi sebagai input maupun output menggunakan fungsi bawaan seperti `pinMode()`, `digitalWrite()`, dan `digitalRead()`. Seluruh fungsi digital ini beroperasi pada tegangan kerja 5 Volt, sesuai dengan spesifikasi logika TTL standar. Setiap pin digital pada Arduino Mega mampu memberikan atau menerima arus maksimum hingga 40 mA, yang cukup untuk mengendalikan komponen eksternal seperti LED, transistor, atau relay kecil. Selain itu, setiap pin digital juga dilengkapi dengan resistor pull-up internal bernilai antara 20 k Ω hingga 50 k Ω , namun resistor ini dalam keadaan non-aktif secara default dan dapat diaktifkan melalui konfigurasi perangkat lunak. Pin digital pada Arduino Mega tidak hanya meliputi fungsi I/O dasar, tetapi juga memiliki pin-pin khusus untuk interupsi eksternal, komunikasi serial (UART), PWM, serta SPI dan I²C, yang menambah fleksibilitas dalam berbagai proyek elektronika dan automasi yang lebih kompleks dibandingkan Arduino UNO.

2.2.4. Komunikasi

Arduino Mega 2560 dilengkapi dengan berbagai fasilitas komunikasi yang memungkinkan koneksi dengan komputer, papan Arduino lainnya, maupun

mikrokontroler eksternal. Berbeda dari Arduino UNO yang hanya memiliki satu saluran komunikasi serial, Arduino Mega menyediakan empat port komunikasi UART TTL (5V), yang masing-masing dapat diakses melalui pasangan pin digital: Serial0 pada pin 0 (RX) dan 1 (TX), Serial1 pada pin 19 (RX) dan 18 (TX), Serial2 pada pin 17 (RX) dan 16 (TX), serta Serial3 pada pin 15 (RX) dan 14 (TX). Untuk koneksi ke komputer, Arduino Mega menggunakan chip ATmega16U2 sebagai penghubung komunikasi serial ke USB, yang dikenali sebagai port COM virtual di perangkat lunak komputer. Chip ini menggunakan driver USB-to-serial standar, sehingga pada kebanyakan sistem operasi modern seperti Windows, MacOS, dan Linux, tidak diperlukan instalasi driver tambahan. Dalam lingkungan pengembangan Arduino IDE, tersedia fitur Serial Monitor yang memungkinkan pertukaran data teks antara papan Arduino dan komputer secara langsung. Pada papan Arduino Mega juga terdapat indikator LED TX dan RX yang akan menyala ketika terjadi aktivitas transmisi atau penerimaan data melalui chip ATmega16U2 pada jalur USB. Namun, penting untuk dicatat bahwa indikator LED ini tidak akan menyala jika komunikasi terjadi melalui port serial lainnya (Serial1, Serial2, Serial3) yang terhubung langsung ke pin digital dan bukan melalui koneksi USB.

2.2.5. Reset Otomatis

Alih-alih menekan tombol reset secara manual saat mengunggah program, Arduino dilengkapi dengan sistem reset otomatis yang memungkinkan perangkat melakukan reset secara mandiri ketika menerima instruksi berbeda, terutama saat terhubung ke komputer. Pada kontrol aliran perangkat keras, sinyal DTR pada ATmega8U2 atau ATmega16U2 dikoneksikan ke jalur reset ATmega328 melalui kapasitor 100 nanofarad. Ketika sinyal ini diaktifkan dengan menariknya ke level rendah, jalur reset akan diaktifkan cukup lama untuk mereset mikrokontroler. Perangkat lunak Arduino mempermudah proses unggah program dengan memanfaatkan penurunan sinyal DTR sebagai pemicu, sehingga pengiriman data dapat berjalan lancar dan unggahan program dapat dilakukan secara otomatis.

2.2.6. Karakteristik Fisik

Arduino Mega 2560 memiliki dimensi fisik yang lebih besar dibandingkan Arduino UNO, dengan ukuran maksimum papan PCB sekitar 4.0 x 2.1 inci (sekitar 101.5 mm x 53.3 mm). Meskipun ukuran PCB-nya sudah cukup besar, dimensi keseluruhan perangkat dapat sedikit bertambah akibat keberadaan konektor USB tipe B dan power jack yang menonjol dari sisi papan. Untuk memudahkan pemasangan pada permukaan atau casing, Arduino Mega juga dilengkapi dengan empat lubang sekrup (mounting holes) yang tersebar di sudut-sudut papan. Hal ini memungkinkan papan dipasang dengan stabil pada enclosure, panel proyek, atau permukaan kerja lainnya. Sama seperti Arduino UNO, perlu diperhatikan bahwa terdapat penyimpangan jarak standar antara beberapa pin—terutama antara pin digital 7 dan 8, yang memiliki jarak tidak standar sebesar 0.16 inci (160 mil), berbeda dari pola 0.1 inci antar pin lainnya. Ketidakteraturan ini penting dicatat, terutama bagi pengguna yang ingin menggunakan breadboard atau papan ekspansi yang memerlukan jarak pin yang presisi.

2.4. Mikrokontroler

Mikrokontroler adalah sebuah chip kecil yang berfungsi sebagai otak dari sistem elektronik. Alat ini digunakan untuk mengendalikan berbagai perangkat secara otomatis, seperti pada sistem lampu lalu lintas, mesin cuci, atau remote control. Penggunaan mikrokontroler membuat rangkaian elektronik menjadi lebih sederhana karena dapat menggantikan banyak komponen pendukung seperti IC TTL dan CMOS. Selain itu, mikrokontroler juga membantu menekan biaya dan meningkatkan efisiensi sistem.

Secara umum, mikrokontroler terbagi menjadi dua jenis berdasarkan arsitekturnya: CISC (Complex Instruction Set Computer) dan RISC (Reduced Instruction Set Computer). Mikrokontroler CISC memiliki instruksi yang lengkap dan kompleks, namun proses kerjanya cenderung lebih lambat karena lebih rumit. Sementara itu, mikrokontroler RISC memiliki instruksi yang lebih sedikit dan sederhana, tetapi bekerja lebih cepat dan efisien. RISC biasanya digunakan pada sistem yang membutuhkan kecepatan tinggi dan hemat daya.

2.5. Sensor Inframerah

Sensor inframerah (infrared sensor) merupakan perangkat elektronik yang digunakan untuk mendeteksi keberadaan atau pergerakan objek dengan memanfaatkan sinar inframerah. Sensor ini bekerja dengan memancarkan sinyal inframerah melalui LED IR, yang memiliki panjang gelombang sekitar 850–950 nm, kemudian mendeteksi pantulan sinyal tersebut menggunakan komponen penerima berupa photodiode atau phototransistor. Ketika sebuah objek berada dalam jangkauan sensor, sinyal inframerah akan dipantulkan kembali ke penerima. Pantulan ini kemudian diubah menjadi sinyal listrik yang akan diproses oleh rangkaian penguat dan komparator untuk menghasilkan sinyal digital berupa logika HIGH (1) atau LOW (0), tergantung ada atau tidaknya objek di depan sensor. Sensor inframerah ini biasanya bekerja dalam jarak pendek, yakni sekitar 2 hingga 30 cm, tergantung pada kekuatan pancaran dan sensitivitas penerimanya.

Dalam penerapan pada sistem pengaturan lampu lalu lintas berbasis jumlah kendaraan, sensor infrared digunakan untuk mendeteksi kendaraan yang melintas pada jalur tertentu. Data hasil deteksi kendaraan tersebut akan menjadi input bagi mikrokontroler, seperti Arduino, untuk menentukan durasi lampu lalu lintas secara adaptif. Sensor ini memiliki keunggulan seperti harga yang terjangkau, konsumsi daya rendah, respon cepat, serta kemudahan integrasi dengan sistem mikrokontroler. Namun, sensor ini juga memiliki keterbatasan, seperti sensitivitas terhadap cahaya lingkungan yang tinggi—misalnya sinar matahari langsung—yang dapat mempengaruhi keakuratan pembacaan, serta kurang efektif dalam mendeteksi objek berwarna gelap atau permukaan yang menyerap sinyal inframerah. Oleh karena itu, pemanfaatan sensor infrared perlu disesuaikan dengan

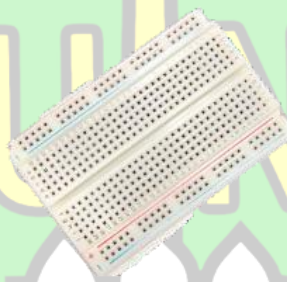


Gambar 2. 2 Sensor Inframerah

kondisi lingkungan dan dikalibrasi secara optimal agar dapat digunakan secara efektif dalam berbagai aplikasi, khususnya dalam sistem lalu lintas cerdas.

2.6. Bread Board

Breadboard adalah papan uji coba yang digunakan dalam perakitan modul elektronik, terutama oleh pemula yang baru mengenal perangkat elektronik. Papan ini memungkinkan pengujian dan penyusunan rangkaian sementara tanpa perlu penyolderan permanen. Selain itu, komponen yang telah digunakan dalam suatu rangkaian dapat dilepas dan digunakan kembali untuk keperluan lainnya.



Gambar 2. 3 Bread Board

2.7. Resistor

Resistor merupakan sebuah komponen elektronika yang berfungsi mengatur atau menghambat alur listrik yang mengalir pada perangkat elektronika. Resistor memiliki sifat resistif dan masuk kedalam kategori perangkat elektronika yang bersifat pasif. Satuan dalam hitungan resistor yaitu Ohm dan di lambangkan dengan simbol omega.

2.8. Kabel Jumper

Kabel jumper merupakan penyambung antara sebuah module yang telah di rangkai dalam sebuah perangkat elektro yang telah di buat dan kabel jumper di transmisikan pada sebuah komponen yang menghubungkan satu komponen dengan komponen lainnya ataupun sesuatu yang menghubungkan jalur rangkaian yang terputus pada breadboard.



Gambar 2. 4 Kabel Jumper

2.9. Algoritma Proporsional

Algoritma proporsional adalah metode pengendalian yang menyesuaikan respons sistem secara sebanding dengan besarnya kesalahan yang terjadi, yaitu selisih antara nilai yang diinginkan (setpoint) dan nilai aktual. Dalam konteks pengaturan lampu lalu lintas, algoritma ini dapat digunakan untuk menyesuaikan durasi lampu hijau berdasarkan jumlah kendaraan yang terdeteksi pada setiap jalur. Prinsip kerja algoritma proporsional melibatkan penentuan konstanta proporsional (K_p) yang mengalikan nilai kesalahan untuk menghasilkan output kontrol. Semakin besar nilai K_p , semakin sensitif respons sistem terhadap perubahan jumlah kendaraan. Namun, nilai K_p yang terlalu tinggi dapat menyebabkan sistem menjadi tidak stabil, sedangkan nilai yang terlalu rendah dapat membuat respons sistem lambat terhadap perubahan kondisi lalu lintas.

Implementasi algoritma proporsional dalam sistem lampu lalu lintas bertujuan untuk mengurangi waktu tunggu kendaraan dan meningkatkan efisiensi arus lalu lintas. Dengan menyesuaikan durasi lampu hijau secara dinamis berdasarkan kepadatan kendaraan, diharapkan dapat mengurangi kemacetan dan meningkatkan efisiensi lalu lintas. Meskipun algoritma proporsional menawarkan solusi yang lebih adaptif dibandingkan pengaturan lampu lalu lintas dengan waktu tetap, tantangan dalam penerapannya meliputi penentuan nilai K_p yang optimal dan respons terhadap fluktuasi lalu lintas yang cepat. Oleh karena itu, penelitian lebih lanjut diperlukan untuk mengatasi tantangan ini dan meningkatkan kinerja sistem pengaturan lampu lalu lintas berbasis algoritma proporsional.

Dalam beberapa kasus, algoritma proporsional dapat dikombinasikan

dengan metode kontrol lainnya, seperti kontrol integral dan derivatif, untuk membentuk kontrol PID yang lebih kompleks dan adaptif terhadap berbagai kondisi lalu lintas. Penelitian terbaru menunjukkan bahwa penerapan algoritma proporsional dalam sistem pengaturan lampu lalu lintas dapat meningkatkan efisiensi arus kendaraan dan mengurangi waktu tunggu, terutama pada persimpangan dengan volume lalu lintas yang tinggi.

Algoritma proporsional dalam pengaturan lalu lintas bekerja dengan prinsip dasar:

$$\text{Waktu Hijau untuk Jalur } i = \left(\frac{\text{Kendaraan di Jalur } i}{\text{Total Kendaraan}} \right) \times \text{Total Waktu Hijau}$$

Misalkan jumlah kendaraan yang menunggu di setiap jalur adalah:

- Jalur 1 = 25 kendaraan
- Jalur 2 = 35 kendaraan
- Jalur 3 = 50 kendaraan
- Jalur 4 = 40 kendaraan
- Total waktu hijau yang tersedia untuk dibagi = 160 detik

Hitung total kendaraan

$$\text{Total Kendaraan} = 25 + 35 + 50 + 40 = 150$$

Hitung proporsi setiap jalur

$$\text{Proporsi Jalur } i = \frac{\text{Jumlah Kendaraan di Jalur } i}{\text{Total Kendaraan}}$$

- Jalur 1: $\{25\} \{150\} = 0.1667 \{ (16.67\%) \}$
- Jalur 2: $\{35\} \{150\} = 0.2333 \{ (23.33\%) \}$
- Jalur 3: $\{50\} \{150\} = 0.3333 \{ (33.33\%) \}$
- Jalur 4: $\{40\} \{150\} = 0.2667 \{ (26.67\%) \}$

Hitung waktu hijau untuk setiap jalur

Waktu Hijau untuk Jalur i = Proporsi $\times$ Total Waktu Hijau

- Jalur 1: $0.1667 \times 160 = 26.67$ detik
- Jalur 2: $0.2333 \times 16 = 37.33$ detik
- Jalur 3: $0.3333 \times 160 = 53.33$ detik
- Jalur 4: $0.2667 \times 160 = 42.67$ detik

Dengan menggunakan algoritma ini, distribusi waktu lampu hijau lampu dapat disesuaikan secara proposional berdasarkan jumlah kendaraan yang mengantri di setiap jalur. Jalur dengan volume kendaraan lebih tinggi akan memperoleh waktu hijau yang lebih Panjang, sehingga membantu mengurangi kemacetan dan meningkatkan kelancaran arus lalu lintas secara efektif.



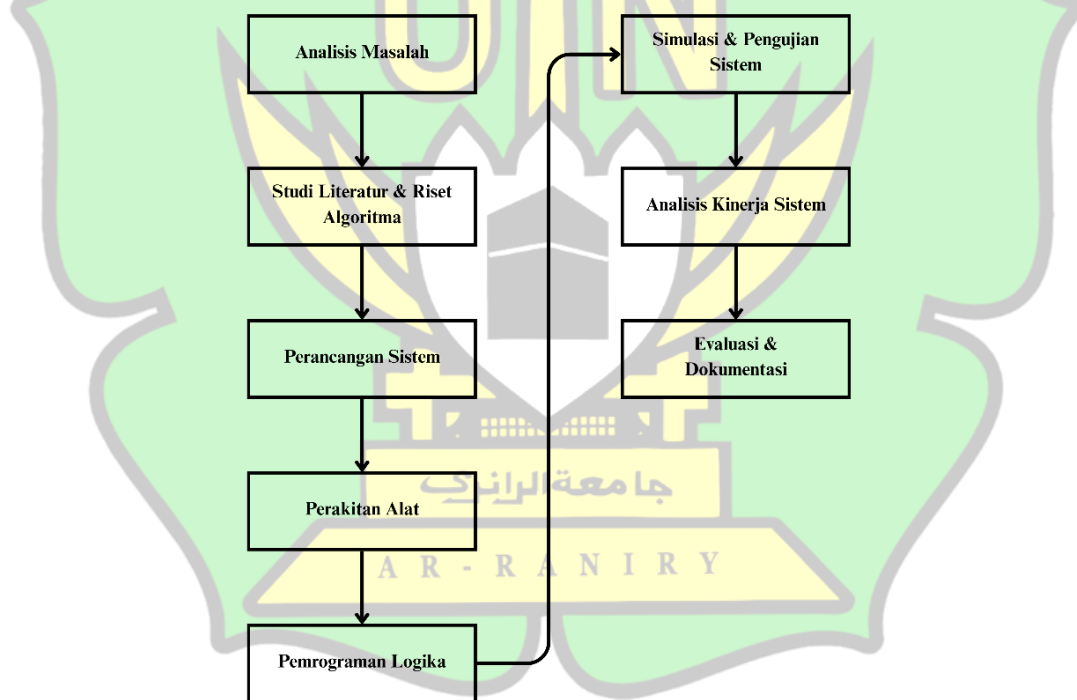
BAB III

METODE PENELITIAN

3.1. Alur Penelitian

Sistem ini dirancang untuk mengoptimalkan pengaturan lampu lalu lintas berdasarkan jumlah kendaraan yang melintas menggunakan Arduino dan sensor ultrasonik. Dengan pendekatan ini, durasi lampu hijau dapat disesuaikan secara otomatis sesuai dengan kepadatan lalu lintas, sehingga dapat mengurangi kemacetan pada persimpangan jalan.

Proses kerja sistem ini melibatkan beberapa langkah utama:



Gambar 3. 1 Flowchart Alur Penelitian

Adapun penjelasan lebih lanjut tentang tahapan penelitian dijelaskan sebagai berikut :

3.1.1. Analisis Masalah

Pada tahap ini, peneliti mengamati permasalahan nyata yang terjadi di lapangan, khususnya terkait sistem lampu lalu lintas yang masih bersifat fixed time. Sistem tersebut tidak mempertimbangkan volume kendaraan yang berbeda di setiap jalur, sehingga kerap menimbulkan antrian panjang dan ketidakefisienan arus lalu lintas. Masalah ini menjadi dasar dari kebutuhan akan sistem yang adaptif dan responsif.

3.1.2. Studi Literatur & Riset Algoritma

Peneliti mengumpulkan referensi dari berbagai sumber, seperti jurnal, artikel ilmiah, dan laporan penelitian sebelumnya, yang berkaitan dengan sistem lalu lintas adaptif, logika proposional, serta penggunaan mikrokontroler dan sensor. Tahapan ini membantu dalam menentukan pendekatan algoritma yang tepat—dalam hal ini adalah algoritma proporsional berbasis logika proposional—untuk diterapkan pada pengaturan lampu lalu lintas.

3.1.3. Perancangan Sistem (Desain Rangkaian dan Algoritma)

Tahapan ini mencakup perencanaan teknis dan visual terhadap sistem, seperti:

- Diagram blok alur kerja sistem,
- Penentuan jenis sensor (IR),
- Rancang logika IF-THEN,
- Skema koneksi antar komponen: Arduino Mega, sensor, LED, dan LCD.

Peneliti merancang sistem agar dapat mengatur waktu lampu hijau secara proporsional sesuai jumlah kendaraan.

3.1.4. Perakitan Alat (Sensor, Arduino, LCD, LED)

Tahap ini melibatkan perakitan fisik semua komponen elektronik:

- Sensor IR untuk mendeteksi kendaraan,
- LED sebagai indikator lampu lalu lintas (merah, kuning, hijau),
- LCD 16x2 untuk menampilkan informasi jumlah kendaraan dan status jalur,
- Semua dirangkai menggunakan breadboard dan kabel jumper, lalu disambungkan ke Arduino Mega.

3.1.5. Pemrograman Logika Proposional (Arduino IDE)

Peneliti menuliskan program dalam Arduino IDE menggunakan struktur logika IF-THEN. Di sini, sistem membaca jumlah kendaraan dari sensor dan menentukan durasi lampu hijau berdasarkan algoritma proporsional. Program kemudian di-upload ke mikrokontroler untuk mengatur lampu dan menampilkan data di LCD.

3.1.6. Simulasi & Pengujian Sistem

Sistem diuji dalam kondisi simulatif. Kendaraan dibuat melintas pada sensor untuk mengamati apakah:

- Sistem mendeteksi jumlah kendaraan dengan akurat,
- Waktu lampu hijau menyesuaikan jumlah kendaraan,
- Tampilan LCD berfungsi sesuai data real-time.

3.1.7. Analisis Kinerja Sistem

Data dari hasil simulasi dikumpulkan dan dianalisis:

- Seberapa cepat sistem merespons perubahan jumlah kendaraan,
- Apakah distribusi waktu lampu sudah sesuai proporsi,
- Stabilitas logika sistem dalam jangka waktu tertentu,
- Tingkat akurasi sensor IR dalam berbagai kondisi.

3.1.8. Evaluasi & Dokumentasi

Tahap akhir melibatkan:

- Revisi sistem jika ditemukan kesalahan atau anomali,
- Finalisasi desain sistem dan program,
- Penyusunan dokumentasi lengkap (gambar rangkaian, flowchart, hasil pengujian),
- Persiapan presentasi hasil untuk seminar tugas akhir.

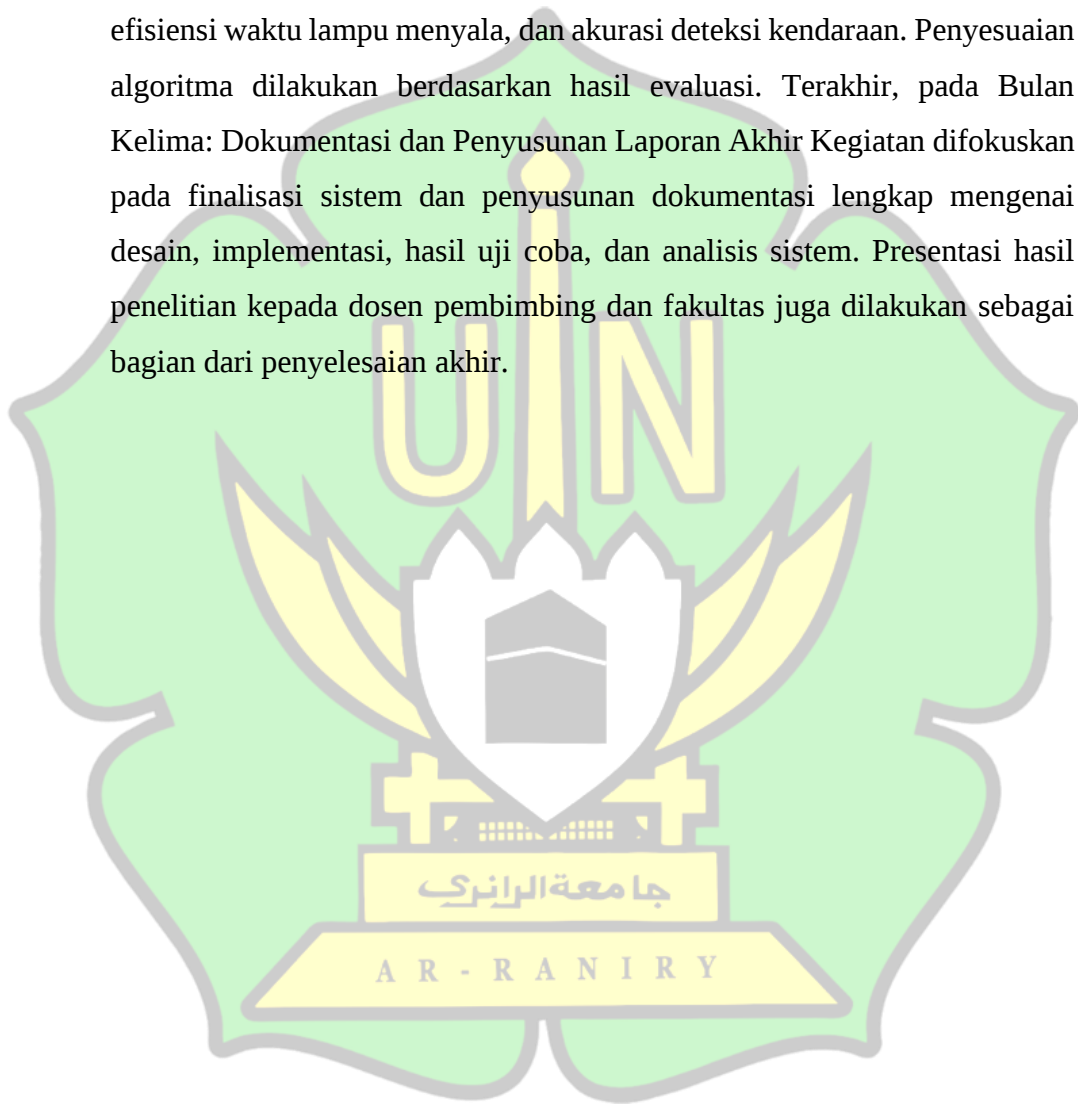
3.2. Jadwal Kegiatan

Jadwal kegiatan dalam penelitian ini dirancang untuk dilaksanakan selama enam bulan, yang mencakup seluruh proses mulai dari analisis hingga evaluasi sistem pengaturan lampu lalu lintas berbasis jumlah kendaraan. Pada Bulan Pertama, Analisis dan Studi Literatur Fokus utama adalah melakukan studi literatur terkait sistem pengaturan lampu lalu lintas otomatis dan penerapan algoritma logika proposional. Selanjutnya, di Bulan Kedua, Perancangan Sistem dan Perangkat Keras Pada tahap ini dilakukan perancangan arsitektur sistem, termasuk penentuan jenis sensor yang digunakan (misalnya sensor ultrasonik) untuk menghitung jumlah kendaraan. Diagram alur logika dan rancangan blok sistem disusun menggunakan pendekatan algoritma proposional.

Memasuki Bulan Ketiga, Implementasi Perangkat Keras dan Perangkat Lunak Penulis mulai merakit perangkat keras menggunakan Arduino, serta menulis program dalam Arduino IDE untuk membaca data

sensor dan mengatur lampu lalu lintas secara otomatis. Integrasi antara sensor, mikrokontroler, dan lampu indikator disiapkan.

Bulan Keempat, Evaluasi dan Optimasi Algoritma Pada tahap ini dilakukan pengujian lebih lanjut untuk mengukur efektivitas sistem dalam mengatur lalu lintas. Evaluasi dilakukan terhadap kecepatan respons, efisiensi waktu lampu menyala, dan akurasi deteksi kendaraan. Penyesuaian algoritma dilakukan berdasarkan hasil evaluasi. Terakhir, pada Bulan Kelima: Dokumentasi dan Penyusunan Laporan Akhir Kegiatan difokuskan pada finalisasi sistem dan penyusunan dokumentasi lengkap mengenai desain, implementasi, hasil uji coba, dan analisis sistem. Presentasi hasil penelitian kepada dosen pembimbing dan fakultas juga dilakukan sebagai bagian dari penyelesaian akhir.



BAB IV

HASIL DAN PEMBAHASAN

4.1. Langkah Kerja Implementasi Sistem

Untuk membangun sistem pengaturan lampu lalu lintas berbasis Arduino dengan sensor pendeteksi kendaraan, diperlukan berbagai komponen elektronik yang bekerja secara terintegrasi. Arduino Mega berfungsi sebagai mikrokontroler utama yang mengontrol nyala lampu lalu lintas berdasarkan jumlah kendaraan yang terdeteksi oleh sensor ultrasonik. LED digunakan sebagai indikator lampu merah, kuning, dan hijau, sementara resistor berperan dalam mengontrol arus listrik agar LED tidak mengalami kerusakan. Breadboard dan kabel jumper membantu dalam penyusunan rangkaian tanpa perlu penyolderan permanen.

4.1.1. Persiapan Alat Dan Bahan

A. Alat

Pada penelitian ini penulis menggunakan aplikasi Arduino IDE. Arduino IDE memiliki peran penting sebagai perangkat lunak utama untuk pengembangan dan pemrograman sistem. Arduino IDE berfungsi sebagai platform tempat penulisan kode program (dikenal sebagai sketch) yang digunakan untuk mengatur logika pengendalian lampu lalu lintas secara otomatis. Melalui Arduino IDE, peneliti dapat mengimplementasikan algoritma proposional dalam bentuk logika IF-THEN yang menentukan kondisi lampu berdasarkan jumlah kendaraan yang terdeteksi oleh sensor.

Selain sebagai media pemrograman, Arduino IDE juga digunakan untuk mengunggah (upload) program ke papan mikrokontroler Arduino seperti Arduino Uno atau Mega. Mikrokontroler tersebut bertugas membaca data dari sensor kendaraan (misalnya sensor ultrasonik atau sensor inframerah) dan kemudian memberikan perintah ke aktuator berupa lampu lalu lintas miniatur (LED atau relay modul). Fitur Serial Monitor yang

tersedia dalam Arduino IDE memungkinkan peneliti untuk memantau data sensor secara real-time dan melakukan proses debugging, sehingga dapat menganalisis respons sistem dan memastikan bahwa logika kontrol berjalan dengan baik sesuai tujuan penelitian.

Dengan demikian, Arduino IDE tidak hanya berfungsi sebagai alat bantu teknis dalam menuliskan dan menguji kode, tetapi juga berperan sebagai media integrasi antara perangkat keras dan logika pengambilan keputusan. Tanpa Arduino IDE, sistem otomatisasi lampu lalu lintas berbasis jumlah kendaraan dan algoritma proposional tidak dapat dijalankan secara efektif dan efisien.

B. Bahan Penelitian

Berikut adalah daftar bahan yang digunakan dalam penelitian ini:

Tabel 3. 1 Bahan Penelitian

No	Nama Bahan	Jumlah	Tampilan	Keterangan
1	Arduino Mega	1		Mikrokontroler utama untuk mengendalikan sistem lampu lalu lintas.
2	LED Merah	4		Indikator lampu merah pada sistem lalu lintas.
3	LED Kuning	4		Indikator lampu kuning pada sistem lalu

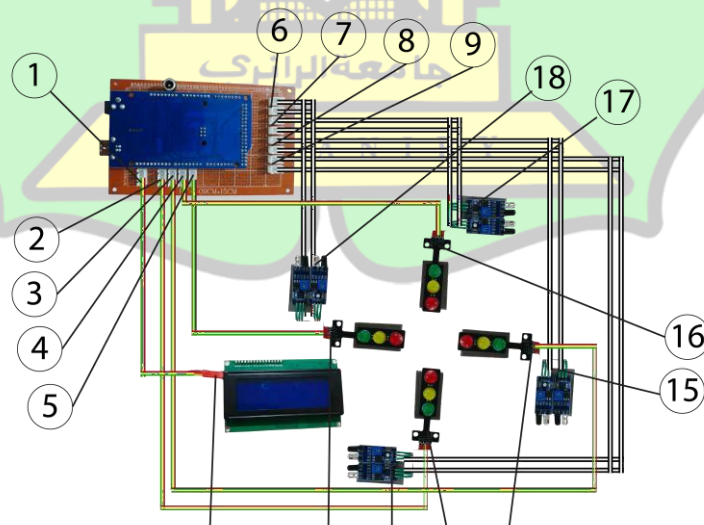
No	Nama Bahan	Jumlah	Tampilan	Keterangan
				lintas.
4	LED Hijau	4		Indikator lampu hijau pada sistem lalu lintas.
5	LCD 16x2 dengan modul I2C	12		Menampilkan jumlah kendaraan yang terdeteksi di tiap jalur.
6	Breadboard	1		Papan tempat merangkai dan menghubungkan komponen tanpa solder.
7	Kabel Jumper	±10		Kabel penghubung antara Arduino, LED, dan breadboard.

No	Nama Bahan	Jumlah	Tampilan	Keterangan
8	Sensor Infrared	2		Digunakan untuk mendeteksi jumlah kendaraan yang melintas.
9	Adaptor USB Arduino	1		Menyediakan daya dan komunikasi antara Arduino dan komputer.

4.1.2. Rangkaian Sistem

A. Desain Alat

Bagian ini menyajikan gambaran skematik dari perakitan alat, yang menunjukkan koneksi antara mikrokontroler, sensor, dan lampu lalu lintas.



Gambar 3. 2 Desain Alat

B. Koneksi Pin

Tabel di bawah ini memaparkan rincian koneksi pin yang digunakan untuk menghubungkan Arduino Mega dengan semua komponen pendukung sistem.

Tabel 3. 2 Koneksi PIN

NO	Dari		Menuju	
	Komponen	Pin	Komponen	Pin
1	Arduino Mega	5V	LCD	VCC
2	Arduino Mega	GND	LCD	GND
3	Arduino Mega	SDA (pin 20)	LCD	SDA
4	Arduino Mega	SCL (pin 21)	LCD	SCL
5	Arduino Mega	5V	Sensor IR 1	VCC
6	Arduino Mega	GND	Sensor IR 1	GND
7	Arduino Mega	D22	Sensor IR 1	OUT
8	Arduino Mega	5V	Sensor IR 2	VCC
9	Arduino Mega	GND	Sensor IR 2	GND
10	Arduino Mega	D23	Sensor IR 2	OUT
11	Arduino Mega	5V	Sensor IR 3	VCC
12	Arduino Mega	GND	Sensor IR 3	GND
13	Arduino Mega	D24	Sensor IR 3	OUT
14	Arduino Mega	5V	Sensor IR 4	VCC
15	Arduino Mega	GND	Sensor IR 4	GND
16	Arduino Mega	D25	Sensor IR 4	OUT
17	Arduino Mega	D30	Lampu Merah Utara	IN
18	Arduino Mega	D31	Lampu Kuning Utara	IN
19	Arduino Mega	D32	Lampu Hijau Utara	IN
20	Arduino Mega	D33	Lampu Merah Timur	IN
21	Arduino Mega	D34	Lampu Kuning Timur	IN
22	Arduino Mega	D35	Lampu Hijau Timur	IN
23	Arduino Mega	D36	Lampu Merah Selatan	IN
24	Arduino Mega	D37	Lampu Kuning Selatan	IN
25	Arduino Mega	D38	Lampu Hijau Selatan	IN
26	Arduino Mega	D39	Lampu Merah Barat	IN
27	Arduino Mega	D40	Lampu Kuning Barat	IN
28	Arduino Mega	D41	Lampu Hijau Barat	IN
29	Arduino Mega	GND	Semua Lampu LED	GND
30	Arduino Mega	5V	Semua Lampu LED	VCC

4.1.3. Prinsip Kerja Sistem

A. Rangkaian Sistem

1. Hubungkan masing-masing sensor infrared ke pin digital Arduino Mega.
2. Hubungkan lampu lalu lintas (LED) ke pin digital untuk masing-masing jalur (merah, kuning, hijau).
3. Hubungkan LCD 16x2 menggunakan I2C ke pin SDA dan SCL Arduino Mega untuk menampilkan status lalu lintas.
4. Pastikan semua ground (GND) dan power (VCC) sensor, lampu, dan LCD terhubung dengan baik.

B. Prinsip Kerja Sistem

1. Sensor IR akan mendeteksi keberadaan kendaraan di setiap jalur.

Proses kerja sistem dimulai dengan modul sensor inframerah yang bertugas untuk mengenali objek seperti gambar (a) dan (b) dibawah ini



Gambar 3. 3. (a) Sensor Tidak Dilalui Oleh Kendaraan. (b) Sensor Dilalui Oleh Kendaraan.

Pada gambar pertama menunjukkan sensor IR tidak sedang dilewati kendaraan, sedangkan gambar selanjutnya menunjukkan sensor yang baru saja dilewati oleh kendaraan

2. Arduino Mega akan menghitung jumlah kendaraan yang terdeteksi pada masing-masing jalur.

Data yang telah dikumpulkan dari sensor kemudian diolah oleh mikrokontroler untuk menentukan kondisi lalu lintas. Kemudian, teks Anda akan berlanjut sebagai berikut:



Gambar 3. 4 Layar LCD

Layar LCD menampilkan:

J1 = Jalur 1

J2 = Jalur 2

J3 = Jalur 3

J4 = Jalur 4

M = Lampu Merah

K = Lampu Kuning

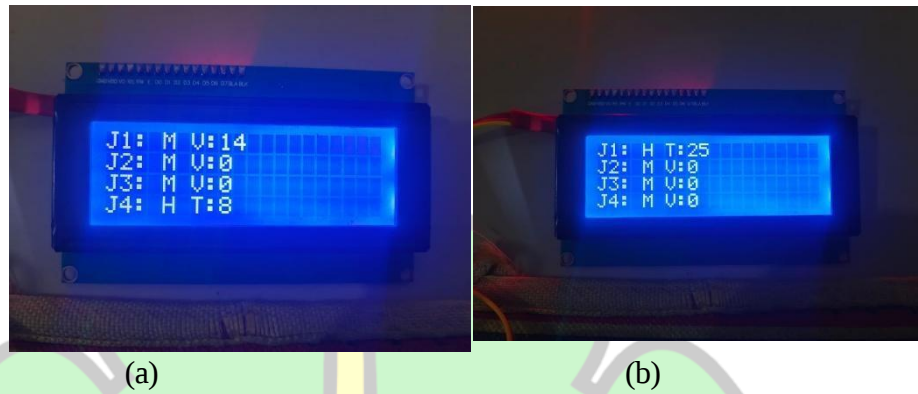
H = Lampu Hijau

T = Time/Waktu

U = Juamlah Kendaraan Yang Berada Pada Jalur

3. Jalur dengan jumlah kendaraan lebih banyak akan mendapatkan durasi lampu hijau lebih lama.

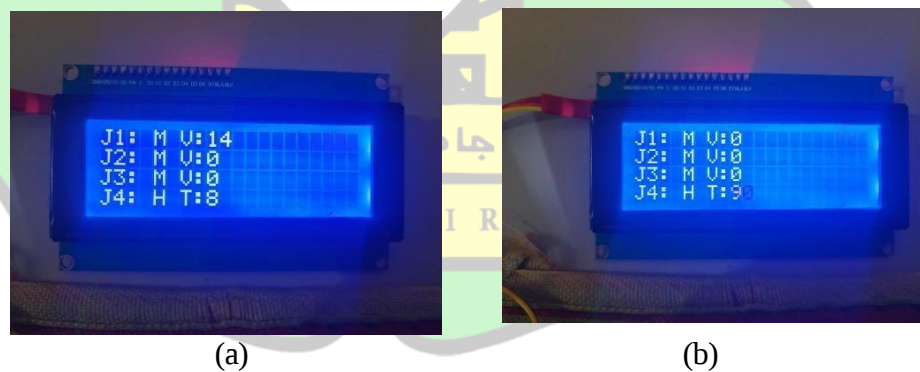
Data jumlah kendaraan yang terdeteksi kemudian menjadi dasar sistem dalam mengimplementasikan logika kontrol. Jalur dengan jumlah kendaraan lebih banyak akan mendapatkan durasi lampu hijau lebih lama.



Gambar 3. 5 (a) Menampilkan Jumlah Kendaraan. (b) Menampilkan Waktu Lampu Hijau

4. Jalur dengan kendaraan lebih sedikit akan mendapatkan durasi lampu hijau lebih pendek.

Sebagai konsekuensi dari algoritma proporsional, durasi lampu hijau juga disesuaikan untuk jalur dengan kepadatan kendaraan yang rendah. Jalur dengan kendaraan lebih sedikit akan mendapatkan durasi lampu hijau lebih pendek.



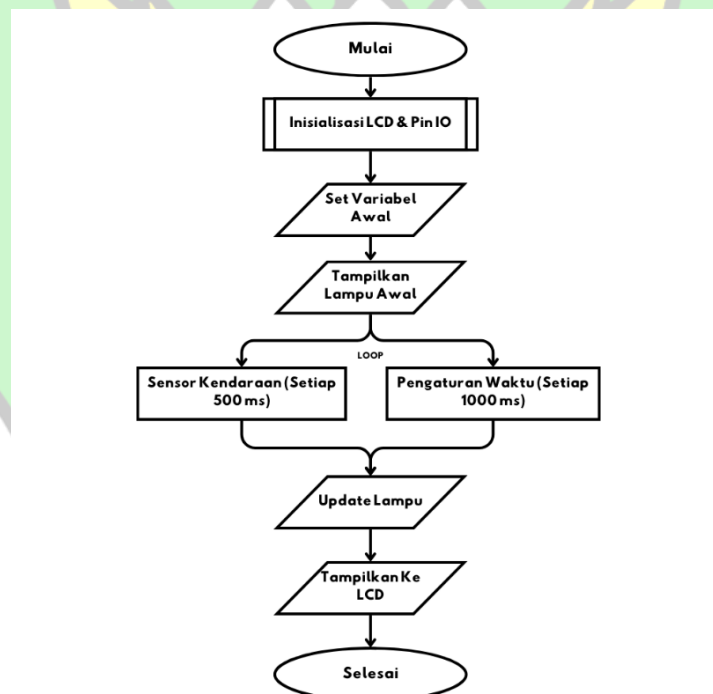
Gambar 3. 6 (a) Menampilkan Jumlah Kendaraan. (b) Menampilkan Waktu Lampu Hijau

5. LCD 16x2 akan menampilkan informasi seperti:
- Jumlah kendaraan di masing-masing jalur.
 - Jalur mana yang sedang hijau.

4.1.4. Logika Pemrograman

A. Flowchart Logika Pemrograman

1. Inisialisasi semua pin untuk sensor, lampu lalu lintas, dan LCD.
2. Baca nilai input dari setiap sensor IR.
3. Hitung jumlah kendaraan berdasarkan sensor yang aktif.
4. Tentukan jalur mana yang mendapat prioritas hijau berdasarkan jumlah kendaraan terbanyak.
5. Atur siklus nyala lampu:
Hijau → Kuning → Merah → Ganti jalur.
6. Update tampilan LCD sesuai perubahan lalu lintas.



Gambar 3. 7 Flowchart Logika Pemograman

4.2. Uji Coba Dan Evaluasi

Uji coba dan evaluasi merupakan tahap penting dalam pengembangan sistem pengaturan lampu lalu lintas berbasis logika proposional. Tujuan utama dari tahap ini adalah untuk memastikan bahwa sistem berfungsi dengan baik, akurat dalam mendeteksi jumlah kendaraan, serta responsif dalam mengatur siklus lampu berdasarkan kepadatan lalu lintas.

Beberapa langkah yang dilakukan dalam proses uji coba dan evaluasi antara lain:

1. Pengujian Fungsi Sensor

Setiap sensor infrared diuji secara individu untuk memastikan kemampuannya dalam mendeteksi kendaraan dengan benar di setiap jalur.

2. Simulasi Kepadatan Lalu Lintas

Dilakukan simulasi dengan meningkatkan jumlah kendaraan secara bertahap pada salah satu jalur untuk menguji apakah sistem memberikan durasi lampu hijau yang lebih panjang secara otomatis sesuai algoritma.

3. Verifikasi Output Tampilan LCD

LCD 16x2 diverifikasi untuk menampilkan informasi real-time terkait jumlah kendaraan yang terdeteksi dan status lampu (merah, kuning, hijau) pada masing-masing jalur.

4. Analisis Respons Sistem

Dievaluasi apakah sistem merespons perubahan jumlah kendaraan dengan cepat dan tepat tanpa adanya delay atau kesalahan switching lampu.

5. Evaluasi Efisiensi Waktu

Waktu rata-rata siklus lampu dan distribusi waktu hijau dievaluasi untuk memastikan efisiensi maksimum dan keadilan bagi seluruh jalur lalu lintas.

6. Identifikasi Kelemahan dan Perbaikan

Jika ditemukan anomali, seperti keterlambatan switching atau kesalahan dalam prioritas jalur, maka dilakukan revisi pada logika pemrograman atau konfigurasi sensor.

Melalui tahapan evaluasi ini, sistem tidak hanya dapat diuji secara teknis, tetapi juga disesuaikan agar dapat bekerja optimal dalam lingkungan nyata. Tahap ini sangat penting untuk validasi performa sistem sebelum diterapkan secara lebih luas, seperti pada proyek kota pintar (smart city).

Tabel 3. 3 Hasil Uji

NO	Item Uji	Harapan	Hasil
1	Deteksi jumlah kendaraan di setiap jalur	Sistem mampu menghitung kendaraan secara akurat di setiap jalur	Terdeteksi dengan akurasi $\pm 95\%$ dalam kondisi pencahayaan baik
2	Perhitungan waktu hijau adaptif	Waktu hijau berubah sesuai dengan kepadatan kendaraan	Waktu hijau terbagi proporsional sesuai jumlah kendaraan di tiap jalur
3	Respon sistem terhadap lalu lintas dinamis	Waktu lampu dapat menyesuaikan perubahan kondisi lalu lintas secara real-time	Waktu lampu berubah setiap 10 detik berdasarkan pembaruan data sensor

NO	Item Uji	Harapan	Hasil
4	Stabilitas logika proposional	Sistem tidak memberikan konflik dalam logika pengaturan lampu	Tidak ditemukan konflik logika selama 60 menit pengujian
5	Toleransi terhadap kesalahan sensor	Sistem tetap berfungsi meski terjadi kehilangan data sesaat	Sistem mampu menggunakan data sebelumnya untuk estimasi sementara

4.3. Hasil Produk

Penelitian ini menciptakan sebuah sistem pengaturan lampu lalu lintas dalam bentuk prototipe miniatur. Prototipe ini berfungsi sebagai model fungsional yang memungkinkan pengujian dan observasi langsung terhadap cara kerja sistem. Dengan model ini, implementasi logika pemrograman dan algoritma proporsional dapat disaksikan secara nyata, memberikan bukti konkret bahwa sistem mampu beradaptasi dan mengatur durasi lampu lalu lintas berdasarkan jumlah kendaraan yang terdeteksi. Model miniatur ini merupakan platform validasi yang penting untuk memastikan bahwa konsep teoretis dapat diwujudkan dalam perangkat keras, dan bahwa sistem dapat berfungsi secara efektif dalam lingkungan simulasi.

BAB 5

KELEBIHAN DAN KEKURANGAN

5.1. Kelebihan

1. Adaptif terhadap Kondisi Nyata Lalu Lintas

Sistem ini mampu menyesuaikan waktu nyala lampu berdasarkan jumlah kendaraan secara real-time, sehingga lebih efisien dibanding sistem statis konvensional.

2. Implementasi Sederhana dan Efisien

Penggunaan algoritma proposional (logika IF-THEN) memungkinkan implementasi logika kontrol yang mudah dipahami dan tidak membutuhkan komputasi kompleks.

3. Biaya Relatif Murah

Sistem dibangun menggunakan perangkat Arduino dan sensor sederhana seperti sensor ultrasonik, sehingga dapat direalisasikan dengan biaya rendah dan teknologi yang mudah diakses.

4. Mudah Dikembangkan dan Dimodifikasi

Karena sistem berbasis mikrokontroler dan logika terbuka, maka sistem ini dapat dengan mudah dikembangkan untuk mendukung lebih banyak persimpangan atau digabungkan dengan sistem IoT.

5. Bersifat Realtime dan Responsif

Sistem memberikan respons langsung terhadap perubahan jumlah kendaraan, meningkatkan efisiensi lalu lintas dan mengurangi waktu tunggu yang tidak perlu.

5.2. Kekurangan

1. Terbatas pada Satu Jalur atau Persimpangan

Penelitian ini masih terbatas pada simulasi satu arah atau satu titik lampu lalu lintas dan belum mengakomodasi koordinasi antar banyak persimpangan (traffic network).

2. Rentan terhadap Gangguan Lingkungan

Sensor ultrasonik atau inframerah bisa mengalami gangguan akibat cuaca, getaran, atau objek non-kendaraan yang memengaruhi akurasi penghitungan kendaraan.

3. Logika Proposional Kurang Fleksibel untuk Kasus Kompleks

Meskipun sederhana, logika IF-THEN tidak dapat menangani skenario lalu lintas yang lebih kompleks atau membutuhkan prediksi berbasis data historis.

4. Tidak Ada Integrasi dengan Basis Data atau Sistem Monitoring

Sistem belum terhubung ke database atau server untuk merekam data lalu lintas, sehingga tidak memungkinkan analisis jangka panjang atau integrasi smart city.

5. Simulasi Masih Terbatas

Sistem diuji dalam skenario terbatas atau simulatif dan belum diuji langsung pada kondisi lalu lintas nyata, sehingga efektivitas di lapangan masih perlu diuji lebih lanjut.

DAFTAR PUSTAKA

- Fahrurnnisa, Z., Rahmadwati, R., & Setyawan, R. A. (2024). Adaptive Traffic Light Signal Control Using Fuzzy Logic Based on Real-Time Vehicle Detection from Video Surveillance. *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)*, 10(2), 235–251.
- Xu, H. (2025). Intelligent Traffic Light Control System and Application. *Applied and Computational Engineering*, 128, 24–30.
- Abood, A. N. (2024). Traffic Light Control in Vehicular Network Systems using Fuzzy Logic. *European Journal of Information Technologies and Computer Science*, 4(4), 1–6. <https://doi.org/10.24018/compute.2024.4.4.132>
- Trivedi, J. D., et al. (2021). A Vision-Based Real-Time Adaptive Traffic Light Control System Using Vehicular Density Value and Statistical Block Matching Approach. *Transport and Telecommunication Journal*, 22(1), 87–97.
- Rahman, M., & Sari, D. (2024). Adaptive traffic light control using proportional algorithm for congestion reduction. *International Journal of Intelligent Transportation Systems*, 18(2), 55–63.
- Zhang, L., & Chen, W. (2024). Vehicle-count based traffic signal optimization with proportional control strategy. *IEEE Transactions on Intelligent Transportation Systems*, 25(1), 112–120.
- Putra, R. A., & Hidayat, M. (2023). Rancang bangun sistem lampu lalu lintas berbasis IoT dengan algoritma proporsional. *Jurnal Teknologi dan Sistem Komputer*, 11(4), 275–283.
- Choudhary, A., & Kumar, S. (2023). Real-time traffic signal control using proportional–integral logic based on vehicle density. *Sensors and Actuators A: Physical*, 359, 114595.

- Nugroho, B., & Santosa, P. I. (2022). Implementasi algoritma proporsional pada sistem pengendalian lampu lalu lintas berbasis kamera. *Jurnal Ilmiah Teknologi Informasi Asia*, 16(2), 89–98.
- Alam, M., & Rahman, M. (2022). Intelligent proportional traffic control with vehicular flow prediction. *Journal of Transportation Engineering, Part A: Systems*, 148(11), 04022112.



LAMPIRAN

A. Source Code

```
#include <Wire.h>

#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 20, 4);

// Lampu per jalur
const int redPins[] = {4, 7, 10, 13};
const int yellowPins[] = {3, 6, 9, 12};
const int greenPins[] = {2, 5, 8, 11};

// Sepasang sensor per jalur: sensorA dan sensorB
const int sensorA[] = {23, 27, 31, 35};
const int sensorB[] = {25, 29, 33, 37};

const int numLanes = 4;
const int yellowTime = 5;
const int baseGreenTime = 10;
const int maxGreenTime = 45;
const int greenBonusPerVehicle = 5;
const int maxVehicleCountForTime = 7; // Maksimal kendaraan yang
dihitung untuk durasi hijau

int currentLane = 0;

int state = 0; // 0 = Merah, 1 = Kuning, 2 = Hijau

int timer = 0;
```

```

int vehicleCount[numLanes] = {0};           // Menyimpan jumlah
kendaraan per jalur

bool prevDetected[numLanes] = {false};      // Deteksi sebelumnya
(untuk debounce)

unsigned long lastMillis = 0;                // Untuk countdown lampu
unsigned long lastSensorMillis = 0;          // Untuk delay pembacaan
sensor
const unsigned long sensorInterval = 500;   // 0,5 detik delay sensor

void setup() {
    lcd.init();
    lcd.backlight();

    for (int i = 0; i < numLanes; i++) {
        pinMode(redPins[i], OUTPUT);
        pinMode(yellowPins[i], OUTPUT);
        pinMode(greenPins[i], OUTPUT);
        pinMode(sensorA[i], INPUT);
        pinMode(sensorB[i], INPUT);
    }

    currentLane = 0;
    state = 2;
    timer = baseGreenTime;
    updateLights();
}

void loop() {
    // ----- SENSOR DETEKSI (SETIAP 0.5 DETIK) -----
    if (millis() - lastSensorMillis >= sensorInterval) {
        lastSensorMillis = millis();
    }
}

```

```

for (int i = 0; i < numLanes; i++) {
    if (i != currentLane && state == 2) {
        bool detectedA = digitalRead(sensorA[i]) == LOW;
        bool detectedB = digitalRead(sensorB[i]) == LOW;

        if (detectedA && detectedB && !prevDetected[i]) {
            vehicleCount[i]++; // Boleh lebih dari 7
            prevDetected[i] = true;
        }

        if (!detectedA || !detectedB) {
            prevDetected[i] = false;
        }
    }
}

// ----- LOGIKA LAMPU PER DETIK -----
if (millis() - lastMillis >= 1000) {
    lastMillis = millis();
    timer--;

    if (timer <= 0) {
        switch (state) {
            case 2: // Hijau → Kuning
                state = 1;
                timer = yellowTime;
                break;

```

```

case 1: // Kuning → Merah

    state = 0;

    timer = 1;

    break;

case 0: // Merah → Jalur berikutnya → Hijau

    currentLane = (currentLane + 1) % numLanes;

    state = 2;

    // Hitung waktu hijau berdasarkan kendaraan
    int kendaraan = vehicleCount[currentLane]; // Total asli
    int hitunganWaktu = kendaraan;

    if (hitunganWaktu > maxVehicleCountForTime) {
        hitunganWaktu = maxVehicleCountForTime; // Batas
maksimal perhitungan
    }

    int calculatedGreen = baseGreenTime + (hitunganWaktu *
greenBonusPerVehicle);

    if (calculatedGreen > maxGreenTime) calculatedGreen =
maxGreenTime;

    timer = calculatedGreen;
    kendaraanCount[currentLane] = 0; // Reset hitungan
kendaraan setelah dilayani

    break;

}

updateLights();

}

displayLCD();

```

```

    }
}

void updateLights() {
    for (int i = 0; i < numLanes; i++) {
        digitalWrite(redPins[i], HIGH);
        digitalWrite(yellowPins[i], LOW);
        digitalWrite(greenPins[i], LOW);
    }

    switch (state) {
        case 1:
            digitalWrite(redPins[currentLane], LOW);
            digitalWrite(yellowPins[currentLane], HIGH);
            break;
        case 2:
            digitalWrite(redPins[currentLane], LOW);
            digitalWrite(greenPins[currentLane], HIGH);
            break;
    }
}

void displayLCD() {
    lcd.clear();
    for (int i = 0; i < numLanes; i++) {
        lcd.setCursor(0, i);
        lcd.print("J");
        lcd.print(i + 1);
        lcd.print(": ");
    }
}

```

```

if (i == currentLane) {
    char s = (state == 2) ? 'H' : (state == 1) ? 'K' : 'M';
    lcd.print(s);
    lcd.print(" T:");
    lcd.print(timer);
} else {
    lcd.print("M V:");
    lcd.print(vehicleCount[i]);    // Tampilkan jumlah kendaraan
    asli, walau > 7
}
}
}

```

