

RANCANG BANGUN SISTEM SMART TRAFFIC LIGHT BERBASIS IOT UNTUK MEMPRIORITASKAN KENDARAAN DARURAT MENGGUNAKAN FUZZY TYPE-2

Alfi Ramadhanar*¹⁾, Basuki Rahmat²⁾, Henni Endah Wahanani³⁾

1. Informatika, Fakultas Ilmu Komputer, UPN Veteran Jawa Timur, Indonesia
2. Magister Teknologi Informasi, Fakultas Ilmu Komputer, UPN Veteran Jawa Timur, Indonesia
3. Informatika, Fakultas Ilmu Komputer, UPN Veteran Jawa Timur, Indonesia

Article Info

Kata Kunci: fuzzy tipe-2, internet of things, kendaraan darurat, smart traffic light

Keywords: fuzzy type-2, internet of things, emergency vehicle, smart traffic light

Article history:

Received 7 May 2025

Revised 11 June 2025

Accepted 16 July 2025

Available online 1 March 2026

DOI :

<https://doi.org/10.29100/jipi.v11i1.7790>

* Corresponding author.

Corresponding Author

E-mail address:

alfiradhamanar33@gmail.com

ABSTRAK

Kemacetan lalu lintas di area perkotaan menjadi salah satu hambatan utama bagi kendaraan darurat yang membutuhkan akses cepat untuk menyelamatkan nyawa atau menuju tempat terjadinya insiden. Penelitian ini bertujuan merancang dan mengimplementasikan prototipe sistem *smart traffic light* berbasis *Internet of Things (IoT)* yang dapat memprioritaskan kendaraan darurat dengan mendeteksi suara sirine menggunakan sensor suara KY-037. Sistem ini dikendalikan oleh mikrokontroler Arduino Uno R3 ditambah dengan *WiFi ESP8266* dan menggunakan metode *Fuzzy Logic Type-2* untuk menangani ketidakpastian tingkat intensitas suara yang diterima sensor, serta menentukan durasi lampu hijau pada arah kendaraan darurat menggunakan *delaytime*. Selain itu, sistem dilengkapi dengan fitur kontrol manual berbasis aplikasi *Blynk IoT* untuk memungkinkan intervensi lampu lalu lintas secara langsung dalam situasi tertentu. Hasil pengujian menunjukkan bahwa sistem prototipe mampu mengidentifikasi suara sirine kendaraan darurat dan merespon dengan mengubah sinyal lampu lalu lintas menjadi hijau untuk memprioritaskan laju kendaraan darurat, serta kembali ke mode normal setelah kendaraan melintas. Sistem ini diharapkan dapat meningkatkan efisiensi layanan darurat dan mengurangi dampak dari kendaraan darurat mengalami keterlambatan pada lampu lalu lintas.

ABSTRACT

Traffic congestion in urban areas is one of the main obstacles for emergency vehicles such as ambulances, firefighters, and police who need quick access to save lives or get to the scene of an incident. This research aims to design and implement an Internet of Things (IoT)-based smart traffic light system prototype that can prioritize emergency vehicles by detecting siren sounds using a KY-037 sound sensor. The system is controlled by an Arduino Uno R3 microcontroller coupled with an ESP8266 WiFi and uses the Fuzzy Logic Type-2 method to handle the uncertainty of the sound intensity level received by the sensor, as well as determining the duration of the green light in the direction of emergency vehicles using delaytime. In addition, the system is equipped with a manual control feature based on the Blynk IoT application to allow direct intervention in certain situations applied directly by officers. The test results show that the prototype system is able to identify the sound of emergency vehicle sirens and respond by changing the traffic light signal to green to prioritize the flow of emergency vehicles, as well as returning to normal mode after the vehicle passes. The system is expected to improve the efficiency of emergency services and reduce the impact of emergency vehicles stuck in traffic jams at intersections.

I. PENDAHULUAN

Kemacetan lalu lintas merupakan permasalahan yang umum terjadi di wilayah perkotaan dengan kepadatan penduduk dan volume kendaraan yang tinggi. Permasalahan ini paling sering terjadi pada area persimpangan jalan, di mana sistem lampu lalu lintas konvensional tidak mampu menyesuaikan waktu nyala berdasarkan kondisi lalu lintas aktual. Sistem yang tidak efisien ini menyebabkan antrean panjang kendaraan

yang berdampak luas, tidak hanya terhadap waktu tempuh masyarakat, tetapi juga terhadap kendaraan darurat yang membutuhkan respons cepat, seperti ambulans, mobil pemadam kebakaran, dan kendaraan kepolisian [1].

Kendaraan darurat seharusnya memperoleh hak prioritas di jalan sebagaimana tercantum dalam Pasal 134 Undang-Undang Nomor 22 Tahun 2009. Namun, pada kenyataannya, tingkat kepatuhan pengguna jalan terhadap aturan tersebut masih rendah. Banyak pengendara yang tidak memberikan ruang bagi kendaraan darurat, baik karena kurangnya edukasi, egoisme saat lalu lintas padat, maupun kondisi jalan yang tidak memungkinkan pengendara lain untuk menepi [2]. Salah satu penyebab utama keterlambatan kendaraan darurat adalah sistem lampu lalu lintas yang menggunakan metode waktu tetap (fixed time), Metode ini masih banyak digunakan di Indonesia berdasarkan Manual Kapasitas Jalan Indonesia (MKJI), namun tidak lagi relevan untuk mengatasi volume kendaraan yang terus meningkat [3]. Keterlambatan kendaraan darurat dalam mencapai lokasi tujuan dapat berakibat fatal, khususnya dalam situasi darurat medis. Sebuah studi menunjukkan bahwa keterlambatan dalam penanganan korban kecelakaan dapat memperparah korban hingga meregang nyawa, terutama jika insiden besar. Faktor-faktor seperti terhambatnya kendaraan darurat karena kemacetan, antrian di persimpangan akan menghambat penanganan korban ketika membutuhkan penanganan medis dengan cepat. Penelitian tersebut menekankan pentingnya upaya sistematis dalam mempercepat respons darurat, termasuk optimalisasi sistem lampu lalu lintas yang memprioritaskan kendaraan darurat supaya cepat ke lokasi insiden atau ke rumah sakit [19].

Sejumlah studi terdahulu telah membahas topik yang sejalan dengan penelitian ini, penelitian oleh W. Sugeng [4], Sistem ini menggunakan Wemos D1 dan database MySQL untuk menentukan apakah kendaraan darurat berada di dalam wilayah aktivasi, kemudian memberikan lampu hijau secara otomatis. Sistem ini menggunakan koordinat latitude dan longitude dari petugas GPS smartphone, kemudian dikirim secara real-time ke server dan diproses oleh mikrokontroler Wemos D1 dan Algoritma Even-Odd digunakan untuk menentukan apakah kendaraan darurat berada di dalam wilayah aktivasi tertentu. Penelitian oleh P.A. Rosyady *et al* [5], Sebuah sistem lalu lintas adaptif dirancang dengan mempertimbangkan panjang antrian kendaraan di setiap jalur. Perangkat ini dikembangkan menggunakan Arduino Uno sebagai unit mikrokontroler, dilengkapi dengan lampu lalu lintas dan tiga sensor inframerah (IR), yaitu IR Detector Photomodule, yang berfungsi untuk mendeteksi keberadaan kendaraan pada tiap ruas jalan. Hasil pengujian menunjukkan bahwa rata-rata kesalahan pada durasi lampu hijau adalah sebesar 0,0040%, pada lampu kuning sebesar 0,0092%, dan pada lampu merah sebesar 0,0204%.

Penelitian oleh M. Harits Ibrahim *et al* [3], Penelitian ini mengembangkan sensor untuk mendeteksi suara sirine dari kendaraan darurat melalui analisis sinyal audio dengan menerapkan metode *Mel-Frequency Cepstral Coefficients* dalam proses identifikasi suara. Ciri khas suara sirine diambil dari nilai koefisien *MFCC*, kemudian diklasifikasikan menggunakan metode jarak *Euclidean*. Setelah pengenalan suara berhasil, sinyal perintah dikirimkan ke Arduino untuk mengaktifkan sistem kendali lampu lalu lintas sesuai dengan situasi di lapangan. Dari hasil pengujian, sistem mampu mencapai tingkat akurasi sebesar 93% dengan rata-rata waktu komputasi selama 14s, sementara durasi operasional rata-rata setiap siklus lampu lalu lintas tercatat sekitar 23s. Hasil penelitian oleh U. Usman *et al* [6], menghasilkan suatu Sistem lampu lalu lintas ini dirancang dengan memanfaatkan teknologi *IoT* dan sensor suara untuk memantau tingkat kepadatan kendaraan di area persimpangan. Ketika sebuah kendaraan darurat datang dari salah satu arah dan mengeluarkan suara sirine yang terdeteksi oleh sensor suara, maka sistem secara otomatis akan mengubah lampu lalu lintas di arah tersebut menjadi hijau. Waktu jeda atau delay selama 10 detik diberikan, dengan ketentuan bahwa suara sirine memiliki intensitas minimal sebesar 118 desibel. Hasil penelitian oleh A.S. Widagda *et al* [7], Pemodelan dilakukan menggunakan Simulink dan SimEvent pada perangkat lunak MATLAB. Dalam model ini diterapkan logika sinyal lalu lintas pre-emptive yang dikombinasikan dengan sebuah switch sebagai penghubung jalur di area persimpangan, di mana kendali algoritma berada dalam blok logika sinyal lalu lintas untuk mendeteksi kendaraan darurat melalui sensor dan mengubah lampu lalu lintas di jalur kendaraan darurat menjadi hijau. Sistem ini menggunakan model antrian M/M/1 untuk kendaraan biasa dan M/D/1 untuk kendaraan darurat dengan disiplin *FIFO (First In, First Out)*, Hasilnya yaitu waktu tempuh kendaraan darurat berkurang hingga 71,15% atau 3,47 kali.

Meskipun berbagai penelitian telah dikembangkan, mayoritas penelitian tersebut masih memiliki keterbatasan dalam hal fleksibilitas pengambilan keputusan di lingkungan yang penuh gangguan suara, serta keterbatasan sistem manual yang tidak dapat diinterupsi secara langsung oleh petugas di lapangan. Penelitian ini mengembangkan sistem lalu lintas otomatis berbasis sensor suara KY-037 yang terintegrasi dengan platform *IoT* dan menggunakan logika Fuzzy Type-2 untuk pengambilan keputusan yang lebih akurat terhadap ketidakpastian *input* suara dalam perkotaan. Selain itu, penggabungan kontrol otomatis berbasis sensor suara dan kontrol manual melalui aplikasi *IoT* di smartphone memungkinkan petugas dalam kendaraan darurat untuk secara langsung menginterupsi sinyal lampu lalu lintas berubah menjadi hijau. Dengan memanfaatkan teknologi *IoT*, sistem lampu lalu lintas dapat dikendalikan dan dimonitor secara jarak jauh melalui perangkat yang terhubung ke internet. Salah satu keuntungan utamanya adalah kemampuan untuk menginterupsi sinyal lalu lintas secara langsung melalui aplikasi *smartphone*,

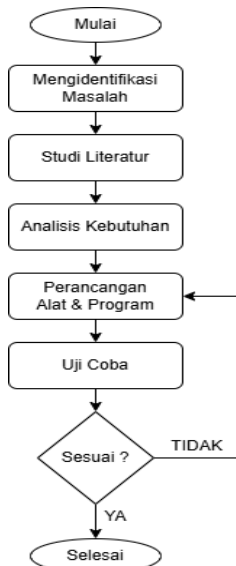
sehingga petugas yang berada di dalam kendaraan darurat, dapat menekan tombol pada aplikasi berbasis IoT untuk mengirim sinyal prioritas ke sistem lalu lintas. Sistem kemudian akan secara otomatis mengubah status lampu menjadi hijau di arah kendaraan darurat, dan merah pada jalur lainnya, sehingga sistem tersebut yang tidak dimiliki oleh sistem konvensional lampu lalu lintas saat ini.

Penelitian ini merancang sebuah prototipe sistem lampu lalu lintas berbasis mikrokontroler Arduino dan sensor suara KY-037 yang mampu mendeteksi suara sirine kendaraan darurat. Ketika suara sirine terdeteksi, sistem akan menginterupsi sinyal lalu lintas dan mengubah lampu menjadi hijau pada arah kendaraan tersebut untuk memberikan prioritas lampu hijau pada kendaraan darurat. Keputusan untuk mengubah status lampu lalu lintas dikendalikan menggunakan Fuzzy tipe2. Fuzzy Type-2 lebih unggul dalam studi kasus ini karena mampu menangani ketidakpastian pada tingkat keanggotaan dengan lebih fleksibel. Artinya, metode ini tidak hanya mengakomodasi variasi nilai *input* dari sensor, tetapi juga ketidakpastian dalam penentuan seberapa kuat suatu sinyal dianggap sebagai sirine, sehingga akan lebih akurat dalam mengambil keputusan meskipun data input tidak bersih atau ambigu dan menjadikannya ideal untuk digunakan dalam sistem *real-time* yang bergantung pada input sensor suara. Dengan adanya sistem ini, diharapkan kendaraan darurat dapat melintasi persimpangan tanpa hambatan, sehingga dapat merespons insiden secara lebih cepat.

II. METODE PENELITIAN

Terdapat tahapan sistem [8] yang harus dilalui untuk mencapai keberhasilan dalam pelaksanaan penelitian ini agar sesuai dengan hasil akhir prototipe yang diharapkan. *Flowchart* [15] dari tahap penelitian dapat dilihat pada Gambar.1.

Langkah selanjutnya melakukan analisis kebutuhan untuk menentukan jenis perangkat keras dan perangkat



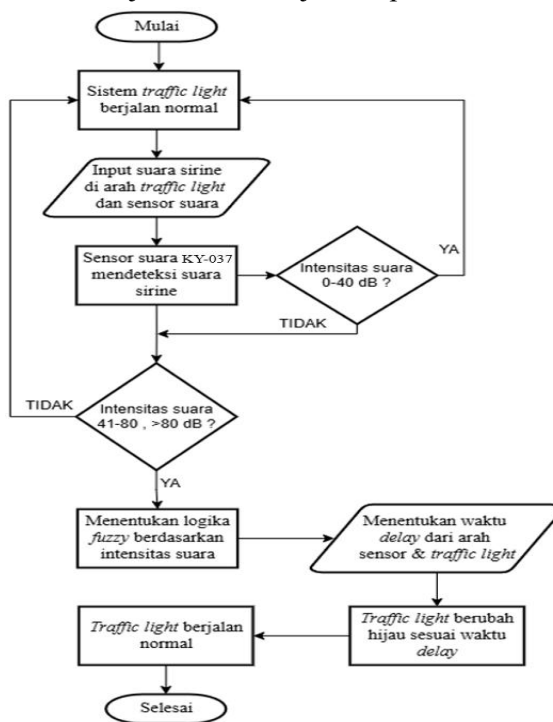
Gambar. 1. Tahapan Penelitian

lunak yang dibutuhkan untuk perancangan *traffic light* untuk memprioritaskan kendaraan darurat ini. Sehingga dapat memastikan bahwa semua bagian perangkat keras dan perangkat lunak yang telah dipilih dapat berfungsi sesuai dengan tujuan penelitian yang diharapkan. Kebutuhan tersebut dapat dilihat pada Gambar.2.

<i>Hardware</i>	Komponen	<i>Software</i>	Keterangan
1	Mikrokontroler Arduino Uno R3 + <i>WiFi</i> ES8266	1	Arduino IDE Aplikasi pemrograman arduino
2	Modul Sensor Suara KY-037	2	<i>Matlab</i> Aplikasi visualisasi <i>fuzzy type-2</i>
3	Modul <i>Traffic Light</i> Traffic Light 5V 8MM R-Y-G Arduino	3	Blynk Aplikasi pendukung <i>IoT</i>
4	Kabel <i>Jumper Dupont 30cm (female to female)</i>		
5	Suara Sirine Kendaraan Darurat Aplikasi <i>Smart Siren 2000 SignalMaster</i>		

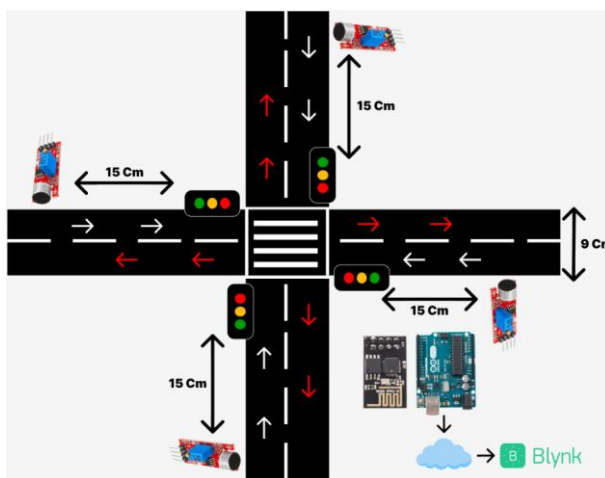
Gambar. 2. Kebutuhan *Hardware* dan *Software*

Selanjutnya membuat *Flowchart* sistem yaitu tahapan dari alur kerja proses sistem secara berurutan [15], pada penelitian ini *flowchart* menggambarkan alur kerja sistem ditunjukkan pada Gambar.3.



Gambar. 3. *Flowchart* Sistem

Selanjutnya melakukan perancangan sistem prototipe yang akan dikembangkan dan bertujuan untuk menghasilkan model perangkat percobaan yang dapat dilakukan proses pengujian. Perancangan sistem prototipe



Gambar. 4. Rancangan Sistem Prototipe

dilakukan sesuai dengan hasil dari analisis kebutuhan. Rancangan sistem prototipe ditunjukkan pada Gambar.4.

Perancangan sistem yang dilakukan dalam penelitian memerlukan serangkaian pengujian atau uji coba untuk memastikan bahwa semua bagian komponen dapat berfungsi normal sesuai dengan yang diharapkan. Skenario ujicoba diperlihatkan pada Gambar.5.

1. Menguji sensor suara 1,2,3,4 tanpa ada sirine (<40 dB)
2. Menguji sensor suara 1,2,3,4 dengan suara sirine (41-80 dB)
3. Menguji sensor suara 1,2,3,4 dengan suara sirine (>80 dB)
4. Menguji modul *traffic light* setelah mendeteksi suara sirine
5. Menguji prototype ketika ada dua kendaraan darurat dari arah yang berbeda
6. Menguji suara sirine kendaraan darurat ketika melewati sensor dari arah jalan sebaliknya
7. Menguji ketika ada lebih dari satu suara sirine kendaraan darurat dari satu arah
8. Menguji waktu respon dari deteksi suara ke perubahan lampu hijau
9. Menguji interupsi manual melalui aplikasi Blynk (tombol *Traffic Light* manual)
10. Menguji modul *traffic light* setelah diaktifkan mode manual melalui aplikasi Blynk
11. Menguji waktu respon pada aplikasi Blynk ke perubahan lampu hijau

Gambar. 5. Skenario Ujicoba Sistem

A. Implementasi Fuzzy Type-2

Logika *fuzzy* merupakan salah satu komponen *soft computing*, dasar logika *fuzzy* adalah teori himpunan *fuzzy*. [18]. Logika *fuzzy type-2* yang diterapkan merupakan IT2FL Sugeno yang digunakan dalam pengambilan keputusan pada sistem *traffic light* berbasis *IoT* untuk memprioritaskan kendaraan darurat menggunakan *fuzzy type-2*. Pengambilan keputusan dirancang untuk menentukan *delay* dari lampu hijau pada *traffic light* atau lampu lalu lintas yang proses pengambilan keputusan dilakukan berdasarkan rentang nilai analog yang dideteksi oleh sensor.

1. Fuzzifikasi (*Fuzzifier*)

Pada penelitian ini digunakan model *Fuzzy Interface System (FIS)* yaitu *Single Input Single Output* yang merupakan implementasi dari satu input yaitu intensitas suara sirine dari kendaraan darurat dan menghasilkan satu *output* yaitu *delayTime*. Pada variabel *input* intensitas suara sirine terbagi menjadi 3 himpunan *fuzzy*, Pada fungsi keanggotaan (MF) sebelah kiri terdapat himpunan “rendah” yaitu untuk rentang intensitas suara kurang dari 40 desibel (<40dB), pada fungsi keanggotaan (MF) tengah terdapat himpunan “sedang” untuk rentang intensitas suara 40-80 desibel, dan fungsi keanggotaan (MF) sebelah kanan terdapat himpunan “tinggi” untuk rentang intensitas suara lebih dari 80 desibel (>80dB). Sedangkan, *Fuzzy Interface System (FIS)* dari variabel *output* utama yaitu *delayTime* terdiri dari 3 himpunan *fuzzy* yaitu “Lama”, “Sedang”, dan “Singkat”.

2. Inferensi (*Inference*)

Inferensi / *inference* merupakan tahap setelah proses fuzzifikasi / *fuzzifier* dimana proses ini menerapkan aturan-aturan *fuzzy* yang telah ditetapkan untuk menghasilkan *output* berdasarkan *input* yang telah diberikan sebelumnya. Pada penelitian ini terdapat tiga himpunan *fuzzy* untuk variabel intensitas suara, sehingga dapat ditentukan aturan dasar dari *fuzzy type-2* untuk proses inferensi / *inference*. Aturan dasar *fuzzy type-2* terdiri dari (R0) **IF** IntensitasSuara is rendah **THEN** *delayTime* is Singkat (0ms), (R1) : **IF** IntensitasSuara is sedang **THEN** *delayTime* is Sedang, (R2) **IF** IntensitasSuara is tinggi **THEN** *delayTime* is Tinggi.

3. Reduksi Tipe (*Type Reducer*)

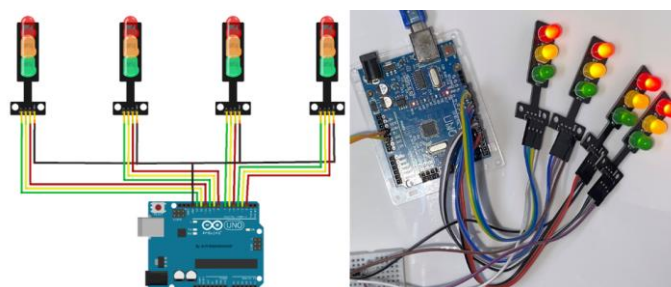
Reduksi tipe / *type reducer* merupakan proses dalam *fuzzy type-2* untuk mengubah *output fuzzy* yang berbentuk interval yaitu *fuzzy type-2* menjadi *output fuzzy* berbentuk nilai tunggal seperti *fuzzy type-1*, sehingga dapat memungkinkan sistem untuk menghasilkan *output* yang lebih mudah diinterpretasikan dan digunakan dalam pengambilan keputusan. Proses Reduksi Tipe dapat dilakukan dengan nilai fungsi keanggotaan Intensitas Suara Sirine berikut: MF (*Upper*) yaitu μ rendah (0-0,35), μ sedang (0,35-0,65), μ tinggi (0,65-1). Dan MF (*Lower*) yaitu μ rendah (0-0,3), μ sedang (0,35-0,6), μ tinggi (0,6-1).

4. Defuzzifikasi (*Defuzzifier*)

Defuzzifikasi / *defuzzifier* merupakan proses terakhir dalam sistem pengambilan keputusan menggunakan *fuzzy type-2* yaitu dengan mengubah *output fuzzy* berbentuk nilai tunggal yaitu hasil dari proses reduksi tipe / *type reducer* menjadi nilai numerik yang akan digunakan oleh sistem untuk mengontrol sistem *traffic light* atau lampu lalu lintas seperti defuzzifikasi / *defuzzifier* ini digunakan untuk menentukan berapa lama *delay* lampu hijau dari *traffic light* akan menyala untuk memberikan lampu hijau untuk kendaraan darurat. Sehingga implementasinya adalah nilai batas bawah *centroid* + nilai batas atas *centroid* kemudian dibagi 2.

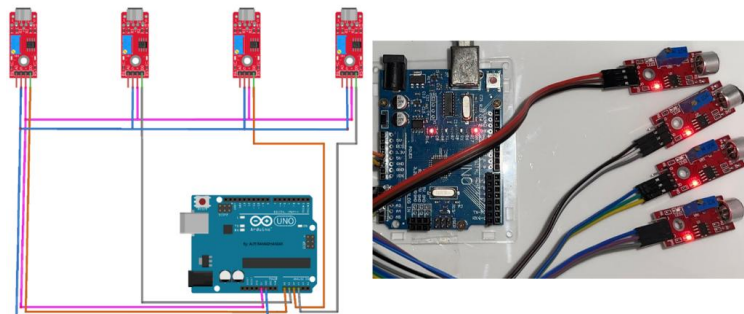
B. Rancangan Hardware dan Software Prototipe

Pada rancangan *hardware*, setiap komponen akan dikombinasikan dengan mikrokontroler utama yaitu arduino uno R3, komponen tersebut terdiri dari modul *traffic light* arduino, modul sensor suara KY-037, dan modul *wifi* ESP8266-01 [11]. Rangkaian awal merupakan rangkaian *hardware* modul *traffic light* yang akan mensimulasikan sistem lampu lalu lintas pada *prototype* yaitu perempatan jalan atau 4 arah simpang, sistem akan mengontrol status lampu merah, kuning, hijau sesuai dengan kode program yang telah diunggah. *Schematic* dan hasil rangkaian ditunjukkan pada Gambar.6.



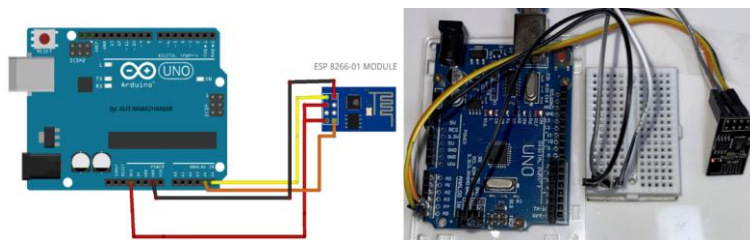
Gambar. 6. Rangkaian *Traffic Light*

Selanjutnya yaitu sensor suara KY-037, masing-masing sensor suara KY-037 tersebut akan dirangkai ke arduino uno R3 dengan *Output Digital* dari setiap sensor suara KY-037 [20] dihubungkan ke pin *Analog In* pada arduino uno R3 [12] untuk memberikan akses sistem dalam mendeteksi suara yang masuk yaitu mensimulasikan suara dari kendaraan darurat. *Schematic* dan rangkaian ditunjukkan pada Gambar.7.



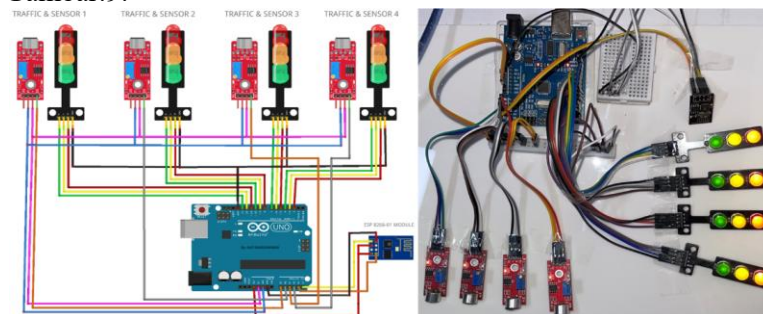
Gambar. 7. Rangkaian Sensor Suara

Selanjutnya merupakan modul *WiFi* ESP8266-01 dengan melakukan proses perakitan dengan menghubungkan modul *WiFi* ESP8266-01 ke arduino uno R3 [13], melalui komunikasi serial menggunakan *library SoftwareSerial* dengan konfigurasi pin RX dan TX terhubung pada pin arduino uno R3. Hasil *Schematic* dan rangkaian ditunjukkan pada Gambar.8.



Gambar. 8. Rangkaian *WiFi*

Rancangan *hardware* terakhir yaitu perancangan hardware keseluruhan yang terdiri dari tiga komponen utama yang telah dihubungkan yaitu modul *traffic light*, modul sensor suara KY-037, Modul *WiFi* ESP8266-01 dan seluruhnya dikendalikan menggunakan mikrokontroler utama yaitu arduino uno R3. Hasil dari *Schematic* dan rancangan ditunjukkan pada Gambar.9.



Gambar. 9. Rangkaian Semua Komponen

Semua bagian tersebut dirakit kedalam sebuah *box* berbentuk persegi berbahan dasar akrilik berukuran 35cm x 35cm. *Box* ini berfungsi sebagai wadah utama sekaligus sebagai area simulasi area persimpangan lalu lintas dengan *traffic light* dan masing-masing arah memiliki 1 *traffic light* dan juga sensor suara. Rancangan akhir semua komponen ditunjukkan pada Gambar.10.



Gambar. 10. Rangkaian Akhir

Pada rancangan *Software* / Program, setiap komponen harus diprogram melalui arduino *IDE* agar masing-masing komponen dapat berfungsi dan berjalan normal sesuai dengan desain sistem yang telah dirancang. Rancangan program dimulai dari komponen sensor suara KY-037, pada implementasi sistem deteksi suara menggunakan sensor suara KY-037 dilakukan dengan membaca sinyal digital berupa desibel (Db) dari 4 sensor yang sudah terhubung di pin analog arduino uno yaitu pin A0, A1, A2, dan A3 kemudian nilai desibel tersebut diproses menggunakan *fuzzifysoundintensity* untuk menentukan berapa lama *delay* durasi prioritas lampu hijau pada jalur sensor suara yang mendeteksi suara sirine, Kode program sensor suara KY-037 ditunjukkan pada Gambar.11.

```
1 | suara1 = digitalRead(A0);
2 | suara2 = digitalRead(A1);
3 | suara3 = digitalRead(A2);
4 | suara4 = digitalRead(A3);
5 | if (suara1 == 1) {
6 |   int soundIntensity = analogRead(A0);
7 |   int delayTime = fuzzifySoundIntensity(soundIntensity);
8 |   timer.stop();
9 |   digitalWrite(r1, LOW); digitalWrite(y1, LOW); digitalWrite(g1, HIGH);
10 |  digitalWrite(r2, HIGH); digitalWrite(y2, LOW); digitalWrite(g2, LOW);
11 |  digitalWrite(r3, HIGH); digitalWrite(y3, LOW); digitalWrite(g3, LOW);
12 |  digitalWrite(r4, HIGH); digitalWrite(y4, LOW); digitalWrite(g4, LOW);
13 |  delay(delayTime);
14 |  timer.start(perempatanTime);
15 | }
16 | //same for suara2, suara3, suara4
17 |
18 | int Tsuara1 = analogRead(A0);
19 | int Tsuara2 = analogRead(A1);
20 | int Tsuara3 = analogRead(A2);
21 | int Tsuara4 = analogRead(A3);
22 | Serial.print("Sensor 1: ");
23 | Serial.print(Tsuara1);
24 | Serial.print(" | Sensor 2: ");
25 | Serial.print(Tsuara2);
26 | Serial.print(" | Sensor 3: ");
27 | Serial.print(Tsuara3);
28 | Serial.print(" | Sensor 4: ");
29 | Serial.println(Tsuara4);
30 | delay(100);
```

Gambar. 11. Kode Program Sensor Suara KY-037

Selanjutnya merupakan implementasi program untuk logika *fuzzy type-2*, Proses fuzzifikasi dilakukan dengan mengubah *input* nilai *crisp* yaitu intensitas suara dari sensor suara menjadi derajat keanggotaan dalam himpunan *fuzzy* sesuai dengan *fuzzysset* [14], *input* intensitas suara yang dibaca sensor suara ky-037 akan dicocokkan dengan fungsi keanggotaan [17] tersebut untuk menghasilkan nilai derajat keanggotaan. terdapat tiga himpunan *fuzzy* untuk *input* intensitas suara yaitu rendah, sedang, dan tinggi. Inferensi *fuzzy* pada program rancangan *prototype* ini dilakukan dengan menerapkan aturan berbasis logika *IF-THEN* terhadap *input fuzzy* yaitu *fuzzytype1estimate* yang akan diproses berdasarkan aturan *fuzzy* (*fuzzy rules*), aturan inferensi ini akan menghubungkan nilai intensitas suara dengan durasi waktu hijau pada lampu lalu lintas atau *traffic light*. [16] Reduksi Tipe diterapkan dengan pendekatan alternatif yaitu menghitung nilai estimasi *output* berdasarkan dua hasil *fuzzy type-1* dari *lower* dan *upper* dari nilai intensitas suara. Hal ini disebabkan karena *library Fuzzy.h* dalam arduino *IDE* belum mendukung *fuzzy type-2*. Defuzzifikasi dilakukan dengan mengubah hasil *fuzzy* berupa dua nilai *delay* hasil dari proses inferensi menjadi satu nilai *crisp* [14] yaitu pada *fuzzifysoundintensity*, yang digunakan untuk mengatur *delaytime* durasi lampu hijau pada lampu lalu lintas. Kode program logika *Fuzzy Type-2* dapat dilihat pada Gambar.12.

```
1 | #include <Fuzzy.h>
2 |
3 | Fuzzy* fuzzy = new Fuzzy();
4 | FuzzyInput* suaraInput = new FuzzyInput(1);
5 | FuzzyOutput* lampuOutput = new FuzzyOutput(1);
6 | suaraInput->addFuzzySet(new FuzzySet(0, 0, 30, 35));
7 | suaraInput->addFuzzySet(new FuzzySet(30, 35, 60, 65));
8 | suaraInput->addFuzzySet(new FuzzySet(60, 65, 100, 100));
9 | lampuOutput->addFuzzySet(new FuzzySet(0, 0, 0, 0));
10 | lampuOutput->addFuzzySet(new FuzzySet(4000, 5000, 5000, 6000));
11 | lampuOutput->addFuzzySet(new FuzzySet(9000, 10000, 10000, 11000));
12 |
13 | int fuzzifySoundIntensity(int intensity) {
14 |   int lower = intensity - 10;
15 |   int upper = intensity + 10;
16 |   int delayLower = fuzzyType1Estimate(lower);
17 |   int delayUpper = fuzzyType1Estimate(upper);
18 |   return (delayLower + delayUpper) / 2;
19 | }
20 | int fuzzyType1Estimate(int intensity) {
21 |   if (intensity < 30) {
22 |     return 1000;
23 |   } else if (intensity < 60) {
24 |     return 10000;
25 |   } else {
26 |     return 15000;
27 |   }
28 | }
```

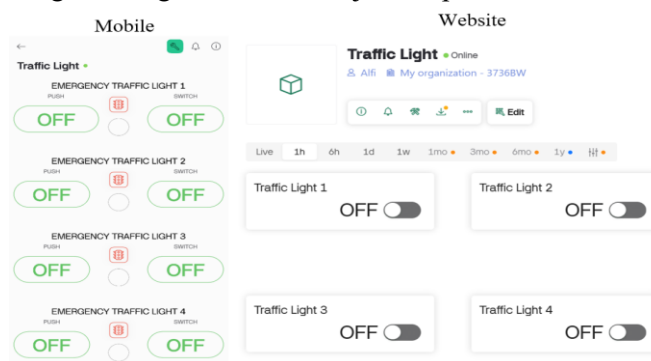
Gambar. 12. Kode Program Implementasi Fuzzy Type-2

```
1 | #define BLYNK_TEMPLATE_ID "TMPL6qqVJ9Z-q"
2 | #define BLYNK_TEMPLATE_NAME "Traffic Light"
3 | #define BLYNK_AUTH_TOKEN "TpymQllieuJUKCfp5b50UgxKvc9XRnQnW"
4 | #include <ESP8266_Lib.h>
5 | #include <BlynkSimpleShieldEsp8266.h>
6 |
7 | char ssid[] = "Ciee gak punya kuota";
8 | char pass[] = "mmmmmmmm";
9 |
10 | #include <SoftwareSerial.h>
11 | SoftwareSerial EspSerial(A5, A4);
12 | #define ESP8266_BAUD 9600
13 | ESP8266 wifi(&EspSerial);
14 |
15 | void setup()
16 | {
17 |   Serial.begin(9600);
18 |   EspSerial.begin(ESP8266_BAUD);
19 |   void loop()
20 | {
21 |   Blynk.run();
22 | }
```

Gambar. 13. Kode Program Implementasi *IoT* Blynk

Selanjutnya yaitu implementasi *Internet of Things* dilakukan dengan memprogram *prototype* dengan menggunakan arduino *IDE* sehingga sistem dapat melakukan konfigurasi *Internet of Things (IoT)* melalui koneksi *Wifi* internet yang menghubungkan sistem rancangan *prototype* dengan platform yang akan digunakan yaitu Blynk. Program untuk konfigurasi *IoT* pada Blynk ditunjukkan pada Gambar.13.

Setelah koneksi sistem *prototype* dengan Blynk berhasil, maka dapat dilakukan pembuatan *templates* pada platform Blynk untuk membuat desain *web* (melalui *website* Blynk) dan juga desain untuk *mobile* (melalui aplikasi Blynk di *playstore* atau *Appstore*). Proses pembuatan desain dapat diubah atau ditambahkan sesuai dengan variasi user, dan ketika proses pembuatan desain ini dilakukan juga proses penambahan *Datastream* untuk konfigurasi antara alat atau *widget* yang ditambahkan dapat melakukan perintah sesuai dengan yang akan dilakukan. Tampilan dari desain *website* yang juga terintegrasi dengan *mobile* ditunjukkan pada Gambar.14.



Gambar. 14. Website dan Mobile pada Blynk

III. HASIL DAN PEMBAHASAN

A. Hasil Pengujian

Hasil pengujian akan membahas hasil dari pengujian dan ujicoba sistem *traffic light* dan sensor suara sesuai dengan skenario ujicoba sistem *traffic light* dan sensor suara. Pengujian pertama yaitu Pengujian Sensor Suara 1,2,3,4 Tanpa Ada Sirine (<40 dB), Pengujian dilakukan dengan mengaktifkan *prototype* dan masing-masing sensor tidak diberi suara sirine, sehingga masing-masing sensor tidak mendeteksi suara sirine dari kendaraan darurat ditandai dengan lampu *led* pada masing-masing sensor tidak menyala. Hasil menunjukkan bahwa masing-masing sensor mendeteksi intensitas suara dalam *serial monitor* kurang dari 40 desibel (<40dB) yang berarti sensor suara tidak mengirim sinyal untuk mengaktifkan prioritas, sehingga sistem lampu lalu lintas atau *traffic light* berjalan normal seperti *traffic light* pada umumnya yaitu berurutan mulai dari *traffic light* 1 hingga *traffic light* 4, dan masing-masing sensor suara menunjukkan bahwa *led input* suara dalam keadaan mati semua, hanya *led power* saja yang aktif. Dokumentasi ditunjukkan pada Gambar.15.

Sensor 1: 37 | Sensor 2: 37 | Sensor 3: 36 | Sensor 4: 36



Gambar. 15. Ujicoba Tanpa Ada Sirine

Pengujian selanjutnya yaitu Pengujian Sensor Suara 1,2,3,4 Dengan Suara Sirine (41-80 dB), Pada pengujian ini, masing-masing sensor suara diberikan *input* suara sirine kendaraan darurat yang berasal dari aplikasi *smartphone* yaitu *Smart Siren 2000* dengan volume yang sama, Pada pengujian tersebut dapat terlihat bahwa *led2* pada sensor suara yang menunjukkan besar kecilnya suara yang diterima oleh sensor yaitu menyala redup, dan pada *serial monitor* arduino *IDE* menunjukkan bahwa sensor suara 1 menerima sinyal suara sirine dari kendaraan darurat antara 41-80 desibel, Hasil ujicoba dengan sirine 41-80 desibel menunjukkan sistem mendeteksi bahwa terdapat suara

sirine dari kendaraan darurat dan akan memproses *input* tersebut, sehingga *traffic light* 1 yang merupakan integrasi dari sensor suara 1 akan memberikan prioritas “sedang” untuk lampu hijau selama 5000ms dan respon yang diberikan oleh sistem tergolong cepat yaitu kurang dari 1 detik sehingga hasil ujicoba dikatakan berhasil, hasil ujicoba pada sensor suara 2,3,4 juga menunjukkan hal yang sama. Dokumentasi ditunjukkan pada Gambar.16.

```
-> Sensor 1: 44  
-> Sensor 1: 47
```



Gambar. 16. Ujicoba Dengan Sirine 41-80dB

Pengujian selanjutnya yaitu Pengujian Sensor Suara 1,2,3,4 Dengan Suara Sirine (>80 dB), pada pengujian dengan suara sirine lebih dari 80 desibel, dilakukan dengan cara yang sama dengan sebelumnya yaitu menggunakan aplikasi *smartphone* yaitu *Smart Siren 2000* dengan volume yang sama untuk setiap sensor yaitu volume tertinggi. Pada pengujian tersebut dapat terlihat bahwa *led2* pada sensor suara yang menunjukkan besar kecilnya suara yang diterima oleh sensor yaitu menyala terang, dan pada *serial monitor* arduino *IDE* menunjukkan bahwa sensor suara 1 menerima sinyal suara sirine dari kendaraan darurat sebesar 985 yang merupakan >80 desibel. Hasil ujicoba dengan sirine >80 desibel menunjukkan sistem mendeteksi bahwa terdapat suara sirine dari kendaraan darurat dan respon yang diberikan oleh sistem tergolong cepat yaitu kurang dari 1 detik sehingga hasil ujicoba dikatakan berhasil, selanjutnya sistem akan memproses *input* tersebut, sehingga *traffic light* 1 yang merupakan integrasi dari sensor suara 1 akan memberikan prioritas “Tinggi” untuk lampu hijau selama 10000ms. Ujicoba yang sama juga dilakukan pada sensor suara 2, sensor suara 3, dan sensor suara 4, hasil ujicoba pada sensor suara 2,3,4 menunjukkan hal yang sama. Dokumentasi ditunjukkan pada Gambar.17.

```
Sensor 1: 985
```



Gambar. 17. Ujicoba Dengan Sirine >80dB

Selanjutnya yaitu melakukan 2 ujicoba, yaitu jika terdapat 2 kendaraan darurat dari arah sama yaitu pada jalur *traffic light* 4, dan 2 kendaraan darurat dari arah yang berbeda yaitu dari jalur *traffic light* 1 dan jalur *traffic light* 4. Pengujian untuk simulasi 2 kendaraan darurat dari arah yang sama yaitu dengan menggunakan suara sirine yang diberikan di sensor suara 4 sebanyak dua kali, sehingga dapat mensimulasikan terdapat 2 kendaraan darurat yang lewat dengan jarak yang tidak terlalu jauh. Pada pengujian 2 kendaraan darurat dari arah jalan yang sama yaitu pada jalur *traffic light* 4, Sensor suara 4 berhasil mendeteksi adanya 2 kendaraan darurat dengan suara sirine yang dapat dilihat pada *serial monitor* terdapat dua kali sensor mendeteksi suara sirine dari kendaraan darurat. Hasil ujicoba menunjukkan bahwa sistem akan memberikan prioritas lampu hijau pada jalur *traffic light* 4 sebanyak dua kali dari *delaytime* yang sudah ditentukan. Sehingga pengujian ini berhasil sesuai dengan yang diharapkan pada skenario uji coba. Dokumentasi ditunjukkan pada Gambar.18.



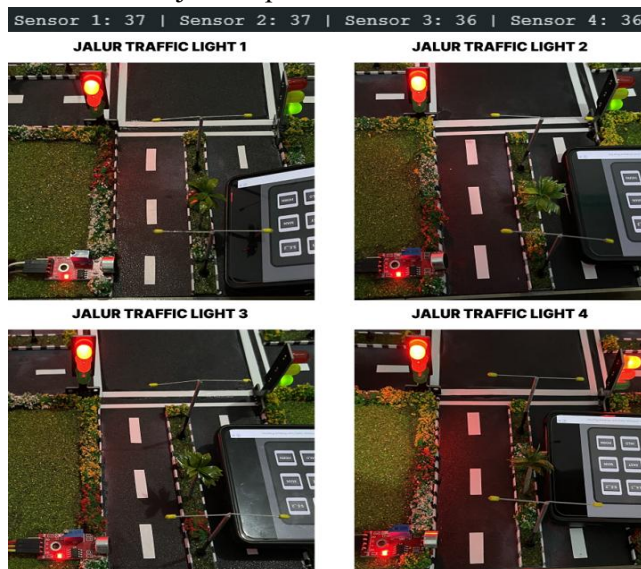
Gambar. 18. Ujicoba 2 kendaraan darurat dari arah sama

Pada pengujian suara sirine dari 2 arah jalur yang berbeda yaitu pada jalur *traffic light* 1 dan jalur *traffic light* 4 dengan mengaktifkan suara sirine pada sensor suara 1 dan sensor suara 4, *serial monitor* pada arduino *IDE* juga akan menunjukkan bahwa sensor suara tersebut mendeteksi adanya suara sirine dari kendaraan darurat secara bersamaan. Hasil dari pengujian menunjukkan bahwa sistem akan memberikan prioritas lampu hijau untuk jalur sensor yang mendeteksi suara sirine terlebih dahulu, pada ujicoba ini jalur sensor 1 merupakan yang terlebih dahulu diberikan suara sirine, selanjutnya sirine diberikan di jalur sensor 4. Dokumentasi ditunjukkan pada Gambar.19.



Gambar. 19. Ujicoba 2 kendaraan darurat dari arah berbeda

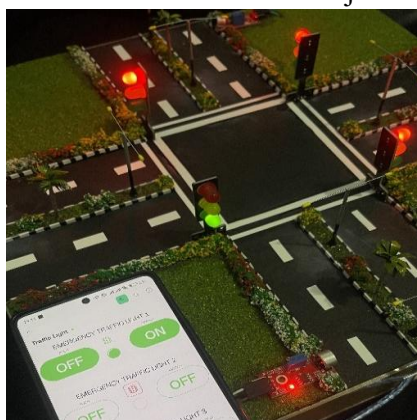
Pengujian selanjutnya yaitu pengujian Sensor Tidak Mendeteksi Suara Sirine dari Jalur Lain, Pada pengujian ini dilakukan simulasi ketika kendaraan darurat telah melewati *traffic light* yang berasal dari jalurnya, kemudian melewati jalur lainnya maka sensor suara dari jalur yang akan dilewati tersebut harus tidak mendeteksi suara sirine dari kendaraan darurat tersebut, karena suara sirine kendaraan tersebut berasal dari arah sebaliknya atau arah berlawanan dari jalur sensor, Dokumentasi ditunjukkan pada Gambar.20.



Gambar. 20. Ujicoba Sensor Tidak Mendeteksi Yang Bukan Jalurnya

Hasil pengujian menunjukkan bahwa sensor suara 1,2,3,4 pada masing-masing arah jalur 1, jalur 2, jalur 3, dan jalur 4 tidak mendeteksi keberadaan suara sirine dari kendaraan darurat, hal ini ditunjukkan dengan tidak aktifnya LED2 pada masing-masing sensor suara, sistem juga tidak menginterupsi sinyal prioritas untuk kendaraan darurat sehingga sistem *traffic light* tetap berjalan normal, pembacaan intensitas suara yang diterima sensor suara pada *serial monitor arduino IDE* juga menunjukkan bahwa masing-masing sensor tidak mendeteksi adanya suara sirine kendaraan darurat yaitu tetap pada rentang kurang dari 40 desibel (<40 dB). Dengan demikian, pengujian Sensor Suara Tidak Mendeteksi Suara Sirine dari Jalur Lain dapat dikatakan berhasil karena sesuai dengan harapan dalam skenario ujicoba, dan membuktikan bahwa sistem dan sensor suara tidak akan mendeteksi suara sirine dari kendaraan darurat yang datang dari jalur arah sebaliknya atau jalur sebrang dari jalur *traffic light* nya.

Pengujian selanjutnya yaitu pengujian Interupsi Manual melalui Aplikasi Blynk, pada pengujian ini dilakukan dengan mengimplementasikan *Internet of Things* [9][10] yaitu melakukan interupsi lampu lalu lintas secara manual menggunakan platform Blynk *mobile*. Hasil ujicoba menunjukkan bahwa sistem dapat merespon dengan cepat (kurang dari 1000ms) ketika tombol dari Blynk untuk *Traffic Light 1* ditekan atau diaktifkan, sistem *traffic light* yang pada awalnya berjalan normal akan langsung memberikan prioritas lampu hijau pada jalur 1 atau *traffic light* 1. Ujicoba selanjutnya yaitu setelah sistem pada jalur 1 atau *traffic light* 1 terinterupsi manual, sistem menunjukkan siklus lampu lalu lintas atau *traffic light* kembali berjalan normal dan sensor suara kembali berjalan normal untuk mendeteksi suara sirine dari kendaraan darurat, yang berarti mode interupsi manual tidak aktif. Ujicoba dilakukan pada jalur 2,3,4 juga, Sehingga hasil ujicoba dikatakan berhasil sesuai dengan skenario ujicoba yaitu *Traffic Light* langsung berubah hijau sesuai dengan arah yang dipilih oleh user melalui aplikasi, waktu respon ketika tombol di klik sampai lampu berubah hijau harus kurang dari 1 detik, dan sistem lampu lalu lintas akan kembali berjalan normal seperti sebelum diaktifkan mode Blynk manual. Dokumentasi ditunjukkan pada Gambar.21.



Gambar. 21. Ujicoba Interupsi Manual melalui Blynk

Berdasarkan hasil pengujian, sistem *smart traffic light* yang dikembangkan mampu mendeteksi suara sirine kendaraan darurat dengan baik menggunakan sensor suara KY-037. Namun, perlu dicatat bahwa sensor suara KY-037 ini memiliki karakteristik sebagai sensor analog pasif yang hanya mampu mengukur tingkat intensitas suara tanpa kemampuan untuk mengenali atau mengklasifikasikan jenis suara secara spesifik, sehingga sensor suara KY-037 ini tidak dapat membedakan apakah suara tersebut berasal dari sirine, klakson, suara mesin, atau sumber suara lain yang tidak relevan, hal ini menjadi keterbatasan utama dari rancangan sistem ini. Karena sensor KY-037 yang digunakan dalam sistem ini hanya mendeteksi tingkat intensitas suara (volume), tanpa kemampuan untuk mengenali atau mengklasifikasikan jenis suara tersebut, sensor ini tidak dapat membedakan apakah suara yang diterima berasal dari sirine kendaraan darurat atau dari sumber suara lain yang tidak relevan. Akibatnya, sensor KY-037 berpotensi menghasilkan *false positive*, yaitu mendeteksi keberadaan kendaraan darurat padahal suara berasal dari sumber lain. Sedangkan dalam skenario *false negative*, kendaraan darurat mungkin memang ada, tetapi tidak terdeteksi oleh sensor karena beberapa kemungkinan, seperti jaraknya terlalu jauh dari sensor dan karena tidak mampu mengklasifikasikan suara, sensor akan mengabaikan suara sirine sehingga sistem gagal memberikan prioritas lampu hijau. Hal ini tentu dapat memperlambat laju kendaraan darurat dan mengurangi efektivitas sistem secara keseluruhan.

Dalam sistem yang diuji, prioritas diberikan kepada kendaraan darurat berdasarkan intensitas suara yang terdeteksi oleh sensor suara KY-037. Jika suara berada dalam rentang 41–80 dB, maka sistem memberikan lampu hijau selama 5000 ms (5 detik), dan jika melebihi 80 dB, maka durasi lampu hijau diberikan selama 10000 ms (10 detik). Pengaturan waktu ini merupakan simulasi dari kondisi nyata, namun telah disesuaikan dengan skala sistem prototipe. Oleh karena itu, waktu 5000 ms dan 10000 ms dalam prototipe ini tidak dimaksudkan sebagai angka tetap atau absolut, melainkan sebagai perumpamaan waktu nyata dalam skala prototipe, yang lebih kecil, lebih cepat, dan tidak mencerminkan kondisi lalu lintas sesungguhnya.

Meskipun hasil pengujian menunjukkan bahwa sistem *smart traffic light* berbasis sensor suara dan kontrol manual melalui aplikasi Blynk bekerja sesuai skenario yang diharapkan, terdapat beberapa tantangan teknis dan potensi keterbatasan sistem yang perlu diperhatikan, seperti Sensor KY-037 belum memiliki kemampuan untuk mengenali atau memverifikasi suara sirine secara spesifik, jika sensor tidak terkalibrasi dengan baik maka waktu deteksi bisa sedikit lebih lama dari yang diharapkan, apabila jaringan internet lemah atau terputus maka tombol kontrol di aplikasi tidak memberikan respon ke sistem.

IV. KESIMPULAN

Sistem pada *prototype* yang telah dirancang berhasil menciptakan simulasi untuk memprioritaskan kendaraan darurat dengan mendeteksi suara sirine dari kendaraan darurat berbasis arduino uno, modul *traffic light*, dan sensor suara KY-037 dan diproses dengan *fuzzy tipe-2*. Metode *Fuzzy tipe-2* dapat diterapkan untuk menentukan prioritas dan durasi lampu hijau menggunakan *delaytime* dengan kategori “Singkat”, “Sedang”, dan “Lama” yang diproses melalui tahapan fuzzifikasi, inferensi, reduksi tipe, dan defuzzifikasi berdasarkan intensitas suara sirine dari kendaraan darurat yang diterima oleh sensor suara KY-037 dengan kategori <40 dB (*Traffic light* normal), 41-80 dB (Prioritas sedang (5000ms)), dan >80 dB (Prioritas tinggi (10000ms)). Implementasi *Internet of Things* pada sistem kontrol lampu lalu lintas berhasil diterapkan pada *prototype* ini dengan menggunakan platform Blynk sebagai kontrol manual dari lampu lalu lintas berbasis *IoT*, sehingga pengguna (Petugas atau pengemudi dari kendaraan darurat) dapat menginterupsi lampu lalu lintas berubah hijau secara manual dengan menggunakan tombol digital yang ada pada aplikasi Blynk di *smartphone*.

Terdapat beberapa kekurangan pada penelitian ini, sehingga diperlukan saran apabila penelitian ini akan dikembangkan lebih lanjut. Seperti, Penambahan bidang kecerdasan buatan (*AI*) pada sistem dengan metode *Machine Learning* seperti *Convolutional Neural Network (CNN)* atau *Support Vector Machine (SVM)* untuk mendeteksi suara sirine dari kendaraan darurat secara lebih akurat,

DAFTAR PUSTAKA

- [1] A. R. Salim, T. Wahyudi, and M. Rahman, "Dampak ekonomi dari kemacetan lalu lintas di Kota Makassar," *Iqtisaduna: Jurnal Ekonomi dan Bisnis Islam*, vol. 13, no. 1, pp. 23–32, 2021.
- [2] Pemerintah Republik Indonesia, Undang-Undang Nomor 22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan. Jakarta: Kementerian Hukum dan Hak Asasi Manusia Republik Indonesia, 2009.
- [3] M. Harits Ibrahim, I. Iwut Tritoasmoro, and L. Novamizanti, "Implementasi sistem kontrol lampu lalu lintas berdasarkan suara sirene pada Arduino," *Agustus*, vol. 7, no. 2, p. 3667, 2020.
- [4] W. Sugeng, "Pengaturan lampu lalu lintas untuk prioritas jalan pada kendaraan darurat menggunakan metoda algoritma Even-Odd," *Jurnal Pekommas*, vol. 2023, pp. 17-28.
- [5] P. A. Rosyady and M. R. Feter, "Prototype lampu lalu lintas adaptif berdasarkan panjang antrian kendaraan berbasis Arduino Uno," *Circuit: Jurnal Ilmiah Pendidikan Teknik Elektro*, vol. 6, no. 2, pp. 173-186, 2022.
- [6] U. Usman, P. Albert, I. Sari, I. Idris, and R. Khair, "Rancang bangun traffic light system tanggap darurat berbasis IoT," *Jurnal Sistem Komputer dan Informatika (JSON)*, 2020.
- [7] A. S. Widagda, S. Sumaryo, and E. Susanto, "Desain dan permodelan pengaturan lampu lalu lintas untuk mengakomodasi kendaraan darurat yang lewat perlintasan jalan," *e-Proceeding of Engineering*, vol. 5, no. 3, pp. 3879–3886, 2018.
- [8] D. Anjeli, S. T. Faulina, and A. Fakhri, "Sistem Informasi Perpustakaan Sekolah Dasar Negeri 49 OKU Menggunakan Embarcadero XE2 Berbasis Client Server," *Jurnal Informatika dan Komputer (JIK)*, vol. 13, no. 2, pp. 57–66, 2022.
- [9] A. Zainudin, I. Anisah, and M. M. Gulo, "Implementasi Fog Computing Pada Aplikasi Smart Home Berbasis Internet of Things," *CESS (Journal of Computer Engineering System and Science)*, vol. 6, no. 1, pp. 127–132, 2021.
- [10] A. Selay, G. D. Andgha, M. A. Alfarizi, M. I. B. Wahyudi, M. N. Falah, M. Khaira, and M. Encep, "Internet of Things," *Karimah Tauhid*, vol. 1, no. 6, p. 860, 2022.
- [11] M. Mustaziri, Y. Mirza, and H. Deviana, "Sistem Monitoring Parkir Mobil Berbasis Mikrokontroler Arduino Uno," *Jurnal JUPITER*, vol. 12, no. 2, pp. 12–25, 2020.
- [12] H. Hartini, S. Primaini, N. Nurhayani, and D. D. Hartanto, "Aplikasi Mikrokontroler Arduino Uno dalam Rancang Bangun Kunci Pintu Menggunakan E-KTP," *Jusikom: Jurnal Sistem Komputer Musirawas*, vol. 7, no. 1, pp. 74–82, 2022.
- [13] T. Ridwan and M. Fajar, "Rancang Bangun Otomasi Rumah Berbasis Arduino Uno dan Smartphone Android," *JTE Jurnal Teknik Elektro*, vol. 5, no. 2, 2021.
- [14] D. C. M. Wijaya, B. Rahmat, and E. Y. Puspaningrum, "Sistem Pendukung Keputusan Berbasis Interval Type-2 Fuzzy Sugeno pada Kendali pH Air," *InComTech: Jurnal Telekomunikasi dan Komputer*, vol. 12, no. 3, pp. 226–239, Dec. 2022.
- [15] J. R. Fauzi, "Algoritma dan Flowchart dalam Menyelesaikan Suatu Masalah," *J. Tek. Inform.*, vol. 3, no. 2, pp. 12, 2020.
- [16] W. Pratama, *Perhitungan Score pada Game Pembelajaran Bahasa Arab Menggunakan Metode Fuzzy Tipe-2*, Skripsi, Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang, 2022.
- [17] S. N. I. Wahyuni, *Penerapan Inferensi Fuzzy Tsukamoto dalam Pengambilan Keputusan Pengiriman Barang (Studi Kasus Jasa Ekspedisi SiCepat Cabang Mojoroto Kota Kediri)*, Skripsi, Program Studi Tadris Matematika, Fakultas Tarbiyah, Institut Agama Islam Negeri Kediri, 2022.
- [18] M. S. Muttaqin, Y. Widiastiti, and C. Nugrahaeni, "Implementasi Logika Fuzzy untuk Setiap Pendukung Keputusan Calon Penerimaan Beasiswa Kartu Jakarta Pintar (Studi Kasus: SMA N 76 Jakarta)" *Seminar Nasional Mahasiswa Ilmu Komputer dan Aplikasinya (SENAMIKA)*, Jakarta-Indonesia, p. 455. Aug. 2020.
- [19] A. Melani, V. Nawangsih, N. Devi, F. Laseda, M. Rizal, and A. Hoirullah, "Analisis Keterlambatan Penanganan Korban Kecelakaan Tanpa Identitas di RSUD Waluyo Jati Kraksaan," *Jurnal Publik*, vol. 18, pp. 119–128, 2024.
- [20] S. Chairil, T. Radillah, and B. Satria, "Sistem Kontrol Menghidupkan Lampu Otomatis Menggunakan Sensor Suara KY-037 Berbasis Arduino Uno," *Indonesian Journal of Computer Science*, vol. 12, no. 1, p. 216, 2023.