

# Perbandingan Kendali Proporsional dan Kendali Logika *Fuzzy* pada Lampu Lalu-lintas

Twii Mulya, Vecky Poekoel, Jane Litouw, Reynold Robot.

Teknik Elektro, Universitas Sam Ratulangi Manado, Jl. Kampus Bahu-Unsrat Manado, 95115, Indonesia  
[twimulya.as@gmail.com](mailto:twimulya.as@gmail.com), [vecky.poekoel@unsrat.ac.id](mailto:vecky.poekoel@unsrat.ac.id), [jane.litouw@unsrat.ac.id](mailto:jane.litouw@unsrat.ac.id),  
[reynold.robot@unsrat.ac.id](mailto:reynold.robot@unsrat.ac.id)

Diterima: Desember 2019; direvisi: Februari 2020; disetujui : Maret 2020

**Abstract** — *Traffic congestion is a problem that often occurs at a crossroads, generally due to imbalanced vehicle volume and road capacity. To overcome this, an appropriate traffic light control system is needed, therefore through this research try to compare which traffic light control is better between those using proportional controls or fuzzy logic. The results obtained from this study are the duration of the green light in the proportional control system is proportional to the number of controlled vehicles, while in fuzzy logic the duration is not only determined by the number of controlled vehicles but also the number of vehicles in other lanes.*

**Keywords** — *Fuzzy Logic; Proportional Controller; Traffic Lights; Mamdani*

**Abstrak** — *Kemacetan lalu-lintas merupakan masalah yang sering terjadi pada persimpangan jalan, umumnya disebabkan tidak seimbang volume kendaraan dan kapasitas jalan. Untuk mengatasinya dibutuhkan sistem pengendali lampu lalu-lintas yang tepat, oleh karena itu melalui penelitian ini coba dilakukan perbandingan kendali lampu lalu-lintas mana yang lebih baik antara yang menggunakan pengontrol proporsional atau logika *fuzzy*. Hasil yang didapatkan dari penelitian ini adalah durasi lampu hijau pada sistem kendali proporsional bersifat proporsional dengan jumlah kendaraan terkontrol, sementara pada logika *fuzzy* durasinya tidak hanya ditentukan jumlah kendaraan terkontrol semata namun juga jumlah kendaraan pada jalur lainnya.*

**Kata Kunci** — *Kendali Proporsional; Lampu lalu-lintas; Logika *Fuzzy*; Mamdani*

## I. PENDAHULUAN

Kemacetan lalu-lintas merupakan masalah yang sering dirasakan oleh masyarakat perkotaan. Penyebabnya ialah tidak seimbang antara volume kendaraan yang meningkat dengan kurangnya kapasitas jalan, hal ini sering terjadi pada persimpangan jalan. Peneliti membuat metode pengendalian lampu lalu-lintas yang bisa melakukan pengaturan berdasarkan kepadatan kendaraan. Metode pengendalian lampu lalu-lintas ini disebut *Vehicle Actuated (VA) Controller* yaitu metode pengendalian dengan siklus waktu yang tidak konstan tetapi berubah berdasarkan tingkat kepadatan kendaraan. Pada metode ini, setiap jalur di persimpangan jalan memiliki detektor atau sensor untuk mendeteksi kepadatan kendaraan di tiap jalur persimpangan.

Oleh karena itu melalui penelitian ini penulis berupaya untuk membuat perbandingan antara sistem pengontrol lampu lalu-lintas yang menggunakan kendali logika fuzzy dan kendali proporsional pada lampu lalu-lintas. Penelitian ini diharapkan dapat memberikan model pembandingan antara kendali logika fuzzy dan kendali proporsional pada lampu lalu lintas serta memberikan opsi solusi permasalahan lalu

lintas dan juga menjadi modul pembelajaran skala lab.

## A. Penelitian Terkait

- 1) Penelitian oleh Atmaja, Bagus Tris, dan Budiman Putra AR. 2015. Implementasi Sistem *Fuzzy* Untuk Pengaturan Lampu Lalu-Lintas Berdasarkan Kepadatan Arus Kendaraan : Penelitian ini menjelaskan pengimplementasian logika *Fuzzy* untuk mengatur lalu-lintas Simpang 4 dan input kepadatan berupa jumlah arus keadaan jalan tiap jalur.[1]
- 2) Penelitian oleh Prasetyo, Erwan Eko. 2016. Perbandingan Kinerja Pengendali Lampu Lalu Lintas Metode *Fuzzy* Tipe Sugeno dengan Metode Waktu Tetap : Penelitian ini menjelaskan desain pengendali lampu lalu-lintas adaptif dengan tiga masukan, yaitu jumlah antrian kendaraan, waktu tunggu dan laju kendaraan yang menuju persimpangan.[4]
- 3) Rahmat Taufik, Supriyono, Sukarman. 2008. Rancang Bangun Simulator Kendali Lampu Lalu Lintas Dengan Logika *Fuzzy* Berbasis Mikrokontroler : Penelitian ini menggunakan penalaran *Fuzzy* metode Sugeno untuk mengatur lamanya waktu dengan menggunakan perangkat lunak MatLab untuk menyelesaikan persoalan tersebut. Dalam rancang bangun ini menggunakan 3 variabel input fungsi keanggotaan dan 3 variabel output fungsi keanggotaan.[7]

## B. Sistem Lampu Lalu-lintas

Lampu lalu-lintas (menurut UU no. 22/2009 tentang Lalu-lintas dan Angkutan Jalan : alat pemberi isyarat lalu-lintas atau disingkat APILL) adalah lampu yang mengendalikan arus lalu lintas yang terpasang di persimpangan jalan, tempat penyeberangan pejalan kaki, dan tempat arus lalu lintas lainnya. Lampu ini yang menandakan kapan kendaraan harus berjalan dan berhenti secara bergantian dari berbagai arah. Pengaturan lalu lintas di persimpangan jalan dimaksudkan untuk mengatur pergerakan kendaraan pada masing-masing kelompok pergerakan kendaraan agar dapat bergerak secara bergantian sehingga tidak saling mengganggu antar-arus yang ada.

Terdapat dua jenis pengontrolan lampu lalu lintas, yaitu *Preset Cycle Time (PCT)* dan *Vehicle Actuated (VA)*. [2]

### 1) *Preset Cycle Time (PCT) Controller*

Merupakan metode pengontrolan lampu lalu-lintas berdasarkan waktu tetap dimana durasi waktunya telah ditentukan terlebih dahulu, yakni 60 detik untuk persimpangan yang tidak terlalu ramai dan 100 detik untuk persimpangan yang ramai dilalui kendaraan. Kelemahan dari

metode ini yaitu tidak mempertimbangkan kepadatan kendaraan dari simpangan manapun.

## 2) Vehicle Actuated (VA) Controller

Pada metode ini, setiap persimpangan jalan memiliki detektor atau sensor untuk mendeteksi jumlah kendaraan pada setiap persimpangan. Metode ini memiliki tiga parameter, yaitu *initial interval*, *extension unit*, dan *extension limit*.

Pada saat lampu hijau pertama kali menyala, maka waktu yang digunakan yaitu *initial interval* kemudian diteruskan oleh *extension unit*. Jika pada kendaraan masih terdeteksi, maka durasinya diperpanjang oleh *extension unit* yang lain hingga mencapai *extension limit*.

## C. Kendali Proporsional

Kendali Proporsional merupakan jenis sistem kontrol linier dimana terdapat hubungan yang sebanding atau proporsional antara keluaran dengan nilai yang diinginkan (*set point*, SP).

Umumnya diagram blok kendali proporsional dapat digambarkan seperti gambar 1 dimana, keluarannya dinyatakan dalam persamaan 1.

$$u(t) = k_p \cdot sp \quad (1)$$

dimana  $K_p$  merupakan nilai koefisien untuk proporsional dan  $SP$  adalah nilai *set point*.

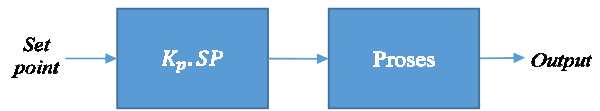
Jika nilai *set point* besar dan positif, maka keluaran kontrol juga besar dan positif.

## D. Logika Fuzzy

Logika *Fuzzy* adalah teknik/metode yang dipakai untuk mengatasi hal yang tidak pasti pada masalah-masalah yang mempunyai banyak jawaban. Pada dasarnya logika *fuzzy* merupakan logika bernilai banyak atau *multivalued logic* yang mampu mendefinisikan nilai di antara keadaan yang konvensional seperti benar atau salah, ya atau tidak, hitam atau putih dan lain-lain. [1]

Tahapan dalam logika *fuzzy* dapat digambarkan seperti gambar 2 yang terdiri dari empat tahapan, yaitu; [4]

- 1) *Fuzzifikasi* adalah suatu proses perubahan nilai tegas yang ada ke dalam fungsi keanggotaan.
- 2) Penalaran adalah proses implikasi dalam menalar nilai masukan guna penentuan nilai keluaran sebagai bentuk pengambilan keputusan.
- 3) Aturan dasar (*rule based*) pada kontrol logika *fuzzy* merupakan suatu bentuk aturan relasi “jika-maka” atau “if-then” seperti berikut ini: *if x is A then y is B* dimana  $A$  dan  $B$  adalah *linguistic values* yang didefinisikan dalam rentang variabel  $X$  dan  $Y$ .
- 4) Defuzzifikasi adalah suatu himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan *output* yang dihasilkan merupakan suatu bilangan pada domain himpunan *fuzzy* tersebut. Sehingga jika diberikan suatu himpunan *fuzzy* dalam *range* tertentu, maka harus dapat diambil suatu nilai *crisp* tertentu.



Gambar 1. Blok kendali proporsional

Sistem inferensi *fuzzy* atau *Fuzzy Inference System* (FIS) merupakan penarikan sebuah kesimpulan dari kumpulan kaidah *fuzzy*. Jadi pada sistem inferensi *fuzzy* penalaran dilakukan menggunakan masukan *fuzzy* (*Crisp Value*) dan *fuzzy rules* yang telah di tentukan sehingga menghasilkan sebuah keluaran *fuzzy* (*Crisp Value*). [3]

Dalam sistem inferensi *fuzzy* terdapat banyak metode yang dapat digunakan, salah satunya adalah metode Mamdani yang diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. Metode Mamdani sering sekali disebut sebagai metode *min-max* yang terdiri dari 4 tahapan:

- 1) Pembentukan Himpunan Fuzzy dengan mengubah variabel masukan (nilai *crisp*) menjadi variabel-variabel linguistik dalam bentuk himpunan *fuzzy*.
- 2) Aplikasi fungsi implikasi yang berupa fungsi min seperti persamaan 2.

$$\mu_{(A \wedge B)}(x) = \min[\mu_{(A)}(x), \mu_{(B)}(x)] \quad (2)$$

- 3) Komposisi aturan dengan menggunakan metode *max* atau metode *maximum* mengambil nilai maksimum aturan, kemudian menggunakannya untuk memodifikasi daerah-daerah *fuzzy* dan mengaplikasikannya ke keluaran menggunakan operator OR (*Union*).
- 4) *Defuzzifikasi*, Masukan dari *defuzzifikasi* berasal dari himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan keluaran yang dihasilkan merupakan suatu bilangan dominan himpunan *fuzzy* tersebut. Variabel keluaran yang dihasilkan oleh *defuzzifikasi* merupakan variabel *crisp*. Ada beberapa metode dalam *defuzzifikasi* pada metode mamdani salah satunya yaitu metode *centroid* yang akan menghasilkan keluaran *crisp* dengan cara mengambil titik pusat daerah *fuzzy*. Secara umum metode *centroid* dapat dirumuskan sebagai berikut:

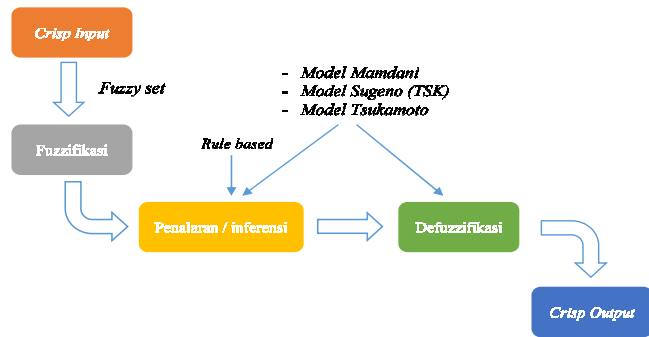
$$\mu(x) = \frac{\int_a^b x\mu(x) dx}{\int_a^b \mu(x) dx} \quad (3)$$

atau

$$\mu(x) = \frac{\sum_{i=1}^n x_i \mu(x_i)}{\sum_{i=1}^n \mu(x_i)} \quad (4)$$

Ada dua keuntungan menggunakan metode *centroid*, yaitu :

- a. Nilai *defuzzifikasi* akan bergerak secara halus sehingga perubahan dari suatu himpunan *fuzzy* juga akan berjalan dengan halus.
- b. Lebih mudah dalam perhitungan.



Gambar 2. Diagram Blok Tahapan Logika Fuzzy

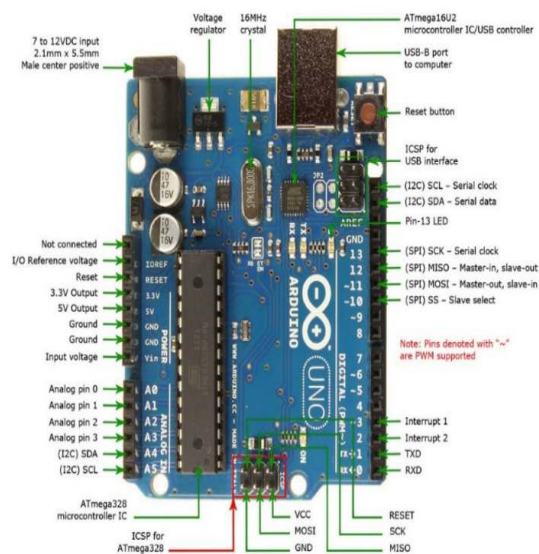
### E. Perangkat Lunak (Software) Arduino IDE

Arduino IDE (*Integrated Development Environment*) adalah aplikasi yang digunakan dalam menulis program dan memasukkan program ke dalam pengontrol Arduino. Lingkungan pengembangannya *open source* Arduino IDE memudahkan untuk menulis kode dan meng-*upload* ke pengontrol Arduino. Aplikasi ini dapat digunakan pada Windows, Mac-OS X, dan Linux. Bahasa pemrogramannya sendiri merupakan penggabungan antara bahasa C++ dan Java dikarenakan struktur bahasa pemrograman dan penggunaan *library* yang mirip dengan C++ dan Java.

### F. Mikrokontroler Arduino UNO

Arduino Uno adalah papan mikrokontroler berbasis ATmega328. Memiliki 14 pin *input / output* digital dimana enam pin di antaranya dapat digunakan sebagai *output* PWM dan enam lainnya sebagai *input analog*. Untuk menghidupkan papan mikrokontroler ini, cukup hubungkan papan ini ke komputer melalui kabel USB atau ke adaptor DC atau baterai. Arduino Uno menggunakan ATmega16U2 yang diprogram sebagai *converter USB-to-serial* berbeda dengan papan sebelumnya yang menggunakan *chip FTDI driver USB- to-serial*. Secara fisik Arduino Uno memiliki bentuk seperti gambar 3 yang terdiri atas berbagai bagian. Nama “Uno” berarti satu dalam bahasa Italia, untuk menandai peluncuran Arduino 1.0. Uno dan versi 1.0 akan menjadi versi referensi dari Arduino. Uno adalah yang terbaru dalam serangkaian *board* USB Arduino, dan sebagai model referensi untuk platform Arduino.

- 1) Power, yang diperoleh dari USB atau catu daya eksternal berupa adaptor atau baterai. Kisaran kebutuhan daya yang disarankan untuk papan Arduino Uno adalah 7 sampai dengan 12 volt.
- 2) Memori program 32 KB (0,5 KB digunakan untuk *bootloader*), 2 KB dari SRAM dan 1 KB EEPROM.
- 3) Input dan Output yang terdiri dari 14 pin di mana beberapa pin tersebut memiliki fungsi khusus:
  - a) Serial : 0 (RX) dan 1 (TX), untuk menerima (RX) dan mengirimkan (TX) data serial TTL. Pin ini dihubungkan ke pin yang berkaitan dengan *chip* Serial ATmega8U2 USB-to-TTL.



Gambar 3. Arduino Uno

- b) *Interrupt External*: di pin 2 dan 3. Pin ini dapat dikonfigurasi untuk memicu *interrupt* pada nilai yang rendah, dengan batasan tepi naik atau turun, atau perubahan nilai.
- c) PWM : 3, 5, 6, 9, 10, dan 11. Menyediakan *output* PWM 8-bit.
- d) SPI : 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Pin ini mendukung komunikasi SPI.
- e) I2C : A4 (SDA) dan A5 (SCL). Dukungan I2C (TWI)
- f) Analog input pada pin A0 sampai A5 sebagai masukan ADC.

### G. Microcontroller Programming and Application Trainer

*Microcontroller programming and application trainer* adalah sebuah perangkat alat peraga yang digunakan sebagai sarana pengujian dan sarana pembelajaran serta pengembangan aplikasi dan program berbasis mikrokontroler Arduino UNO.

### H. Modul Simulasi Traffic Light

Modul Simulasi *Traffic Light* adalah sebuah rangkaian lampu LED yang digunakan untuk mensimulasikan sistem lampu lalu lintas.

### I. Pendeteksi Volume Kendaraan

Untuk mendeteksi volume kendaraan pada persimpangan jalan dapat menggunakan sensor *infrared*, kamera, dan lain-lain. Namun pada penelitian ini penulis menggunakan sensor LDR (*Light Dependent Resistor*). Sensor LDR adalah Resistor peka cahaya atau *photoresistor* yang merupakan komponen elektronik yang resistansinya akan menurun jika ada penambahan intensitas cahaya yang mengenainya.

## II. METODELOGI PENELITIAN

### A. Tempat dan Waktu

Penelitian ini dilaksanakan dimulai dari Seminar Proposal pada bulan Januari 2019 sampai Seminar Hasil pada bulan November 2019, di Laboratorium Teknik Kendali Jurusan

Elektro Fakultas Teknik Universitas Sam Ratulangi.

### B. Prosedur Penelitian

Prosedur penelitian perbandingan kendali logika fuzzy dan kendali proporsional pada lampu lalu lintas terdiri dari:



Gambar 4. *Microcontroller Programming and Application Trainer*

#### 1) Studi Literatur

Berupa kegiatan mencari dan mempelajari berbagai referensi dan informasi yang berkaitan dengan sistem pengendalian proporsional dan fuzzy.

#### 2) Perancangan konsep bentuk modul simulasi

Perancangan konsep dasar sistem yang dimulai dengan menentukan bentuk modul yang akan dipakai.

#### 3) Merancang desain *hardware* modul simulasi *traffic light*.

Perancangan komponen harus sesuai dengan konsep dan dapat digunakan yang kompatibel dengan mikrokontroler Arduino Uno dan perangkat lainnya.

#### 4) Menyiapkan alat dan bahan dalam perancangan modul simulasi *traffic light*.

Menyiapkan alat dan bahan modul simulasi *traffic light* beserta komponen penunjang lainnya yang diperlukan.

#### 5) Implementasi

Dari hasil rancangan dan desain pada proses sebelumnya program (*software*) dan men-download program ke mikrokontroler. Program yang dibuat untuk mengendalikan menggunakan bahasa C++ yang dibuat pada *software* Arduino IDE.

#### 6) Membuat laporan penelitian

Laporan dibuat dan disusun sesuai dengan hasil penelitian.

### C. Diagram Blok Perancangan

Diagram blok perancangan dalam tugas akhir ini digambarkan seperti gambar 4 yang terdiri dari;

1) *Input*, berupa jumlah kendaraan yang datanya diambil dari sensor cahaya LDR. Proses pendeteksian jumlah kendaraan ini menggunakan prinsip, jika ada cahaya yang menuju LDR terhalangi oleh suatu kendaraan, maka nilai tahanan LDR akan membesar sedangkan jika sebaliknya maka nilai tahanan LDR akan mengecil. Besar kecilnya nilai tahanan LDR ini akan menjadi acuan input jumlah kendaraan yang dibaca oleh pengontrol.

2) *Pengontrol*, berupa mikrokontroler Arduino Uno yang berfungsi untuk memproses data jumlah kendaraan dari LDR menjadi *timing* durasi lampu

lalu lintas. Pada penerapannya nanti pengontrol ini akan diprogram dengan menggunakan sistem pengontrol proporsional dan sistem logika fuzzy untuk membandingkan hasil pengontrolannya.

3) *Plant*, berupa lampu LED merah, kuning dan hijau untuk mensimulasikan lampu lalu lintas.

Gambar 5. Diagram Blok Perancangan

4) *Output*, berupa durasi lampu lalu lintas yang ditandai dengan perubahan lampu LED yang berkorelasi dengan *input* jumlah kendaraan dan jenis sistem pengontrol yang digunakan.

### D. Rancangan Hardware

Sesuai dengan desain konsep dasar perancangan maka dirancangnya sistem *hardware* seperti berikut:

#### 1) Pendeteksi Volume Kendaraan

Pendeteksi volume kendaraan menggunakan sensor LDR yang bila sensor tersebut tertutupi oleh kendaraan maka terjadi perubahan resistansi yang nanti diolah oleh Arduino sebagai indikasi adanya sejumlah kendaraan pada jalur tersebut sesuai jumlah sensor yang tertutupi. Sensor LDR yang digunakan untuk mendeteksi jumlah kendaraan keseluruhannya berjumlah sembilan buah dengan penempatan seperti gambar 6.

#### 2) Alat Pemberi Isyarat Lalu Lintas

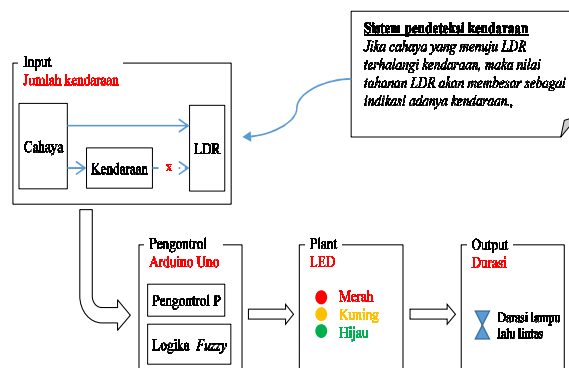
Alat pemberi isyarat lalu lintas yang sering digunakan pada umumnya adalah lampu LED. Lampu LED yang digunakan terdiri dari warna merah, kuning dan hijau yang telah terpasang pada modul kit *trainer traffic light*. Adapun penempatan posisi LED dapat digambarkan pada gambar 6.

#### 3) Pengontrol Arduino

Pengontrol Arduino yang digunakan memiliki beberapa pin yang dihubungkan ke beberapa komponen seperti LDR dan LED.

### E. Rancangan Sistem Pengendalian Lampu Lalu-lintas

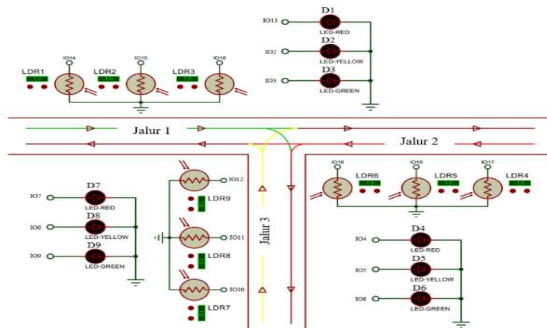
Penulis menggunakan sistem pengontrol *vehicle actuated* (VA) yakni suatu sistem yang menggunakan sensor kendaraan (dalam hal ini adalah LDR) untuk mendeteksi jumlah kendaraan pada setiap jalur. Hasil deteksi jumlah kendaraan dari sensor ini akan mempengaruhi seberapa lama durasi lampu hijau menyala. Terdapat dua sistem pengontrol lampu lalu-lintas yang dirancang yakni; pengontrol proporsional dan logika *fuzzy*.



TABEL I. SUSUNAN PIN ANTARA ARDUINO DENGAN PERANGKAT LAINNYA



Jalur 1				Jalur 2				Jalur 3			
LDR	Pin <sup>+</sup>	LED	Pin <sup>-</sup>	LDR	Pin <sup>+</sup>	LED	Pin <sup>-</sup>	LDR	Pin <sup>+</sup>	LED	Pin <sup>-</sup>
LDR 1	IO1 4	LED 1 3		LDR 4	IO1 7	LED 4 4	IO6	LDR 7	IO1 0	LED 7 7	IO9
LDR 2	IO1 5	LED 2 2		LDR 5	IO1 8	LED 5 5	IO7	LDR 8	IO1 1	LED 8 8	IO1 0
LDR 3	IO1 6	LED 3 3	IO3	LDR 6	IO1 9	LED 6 6	IO8	LDR 9	IO1 2	LED 9 9	IO1 1



Gambar 6. Sambungan dan susunan LDR sebagai Pendeteksi Volume

### 1) Pengontrolan Proporsional

Terdapat tiga parameter yang penulis gunakan dalam sistem pengontrolan proporsional ini ketiga parameter tersebut, yaitu:

- 1) *Initial interval*, yakni durasi lampu hijau saat pertama kali menyala sebesar 10 detik.
- 2) *Extension unit*, yakni durasi lampu hijau yang nilainya proporsional dengan jumlah kendaraan.
- 3) *Extension limit*, dalam hal apabila jumlah kendaraan telah bertambah banyak melewati batas 20 kendaraan, maka durasi lampu hijau akan tetap konstan sebesar 100 detik.

Dari ketiga parameter tersebut maka dapat diasumsikan;

- 1) Pada saat jumlah kendaraan 0 durasi lampu hijau 10 detik.
- 2) Pada saat jumlah kendaraan diatas 20 durasi lampu hijau konstan 100 detik.
- 3) Sedangkan pada saat jumlah kendaraan 0 sampai 20 ditentukan dengan menggunakan pengontrol  $P$  yang secara matematis ditentukan pada persamaan 1. Dimana,  $K_p$  adalah nilai konstanta proporsional yang ditentukan dengan menggunakan persamaan kemiringan garis lurus;

$$K_p = \frac{t_2 - t_1}{x_2 - x_1} = \frac{100 - 10}{20 - 0} = 4,5$$

Karena *initial value* dimulai pada titik (0;10) maka pada persamaan 5 perlu ditambahkan *offset* 10. Sehingga persamaan pengontrol  $P$  untuk jumlah kendaraan 0 sampai 20 menjadi;

$$4,5x + 10 \quad (5)$$

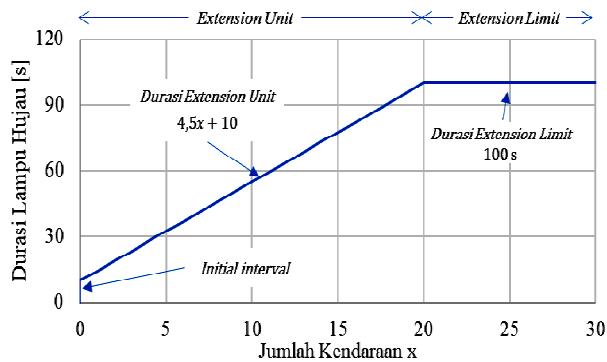
Dengan demikian persamaan durasi lampu hijau secara keseluruhan dapat dinyatakan dalam persamaan 6.

$$t_{\text{lampu hijau}} = \begin{cases} 10 & ; \quad x = 0 \\ 4,5x + 10 & ; \quad 0 < x \leq 20 \\ 100 & ; \quad x > 20 \end{cases} \quad (6)$$

dimana:

$$t_{\text{lampu hijau}} = \text{Durasi lampu hijau [s]} \\ x = \text{jumlah kendaraan}$$

Dapat diamati dari persamaan 4 dan gambar 6 bahwa pengontrol proporsional ini didesain agar durasinya linear terhadap jumlah kendaraan pada rentang nol sampai 20, dan konstan sebesar 100 detik pada saat jumlah kendaraannya melampaui 20.



Gambar 7. Grafik hubungan antara jumlah kendaraan (x) dengan durasi lampu hijau.

### 2) Pengontrol dengan Logika Fuzzy

Pada pengontrol dengan logika *fuzzy* ini penulis menggunakan parameter yang sama dengan pengontrol proporsional namun jika pada sistem pengontrol proporsional hanya menggunakan satu input saja yakni; jumlah kendaraan pada jalur yang dikontrol, maka pada sistem pengontrol dengan logika fuzzy ini penulis menggunakan dua input yaitu; jumlah kendaraan pada jalur yang dikontrol dan jumlah kendaraan pada jalur lainnya. Sementara untuk *output* yakni; durasi lampu hijau.

#### a) Pembentukan Himpunan Fuzzy

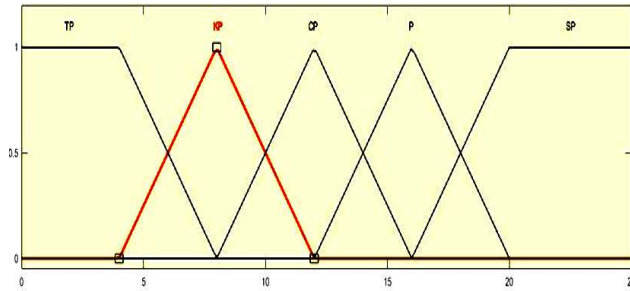
Pada tahapan ini dilakukan pembentukan Himpunan variabel-variabel linguistik dalam bentuk himpunan *fuzzy*, baik untuk masukan (*input*) dan keluaran (*output*) yang akan digunakan, yaitu;

- 1) Himpunan variabel-variabel *input* jumlah kendaraan pada jalur yang dikontrol yang terdiri dari lima variabel linguistik yaitu;
  - a. Tidak padat / TP (0, 4, 8)
  - b. Kurang padat / KP (4, 8, 12)
  - c. Cukup padat / CP (8, 12, 16)
  - d. Padat / P (12, 16, 20)
  - e. Sangat Padat / SP (16, 20, ∞)

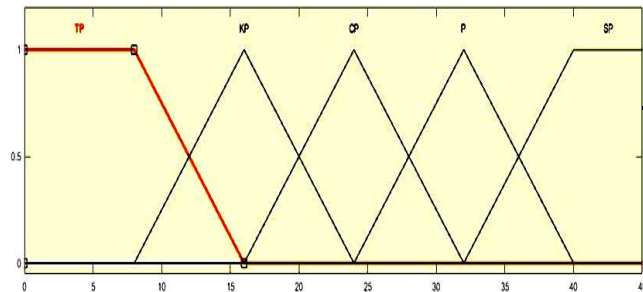
Himpunan ini dapat digambarkan dalam bentuk grafik gambar 8.

- 2) Himpunan variabel-variabel *input* jumlah kendaraan pada jalur lainnya yang terdiri dari lima variabel linguistik yaitu;
  - a. Tidak padat / TP (0, 8, 16)
  - b. Kurang padat / KP (8, 16, 24)
  - c. Cukup padat / CP (16, 24, 32)
  - d. Padat / P (24, 32, 40)
  - e. Sangat Padat / SP (32, 40, ∞)

Himpunan ini dapat digambarkan dalam bentuk grafik gambar 9.

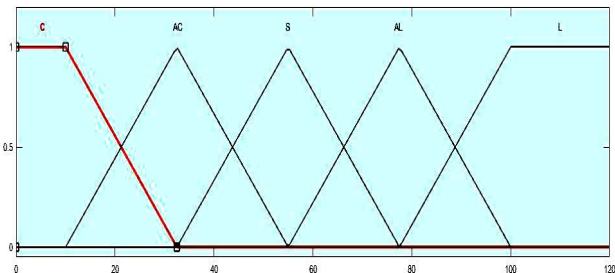


Gambar 8. Grafik fungsi keanggotaan input jumlah kendaraan pada jalur yang dikontrol



Gambar 9. Grafik fungsi keanggotaan input jumlah kendaraan pada jalur lainnya

Dimana  $x_i$  adalah titik-titik sembarang pada daerah area dari bidang hasil komposisi dan  $\mu(x_i)$  adalah derajat keanggotaannya.



Gambar 10. Grafik fungsi keanggotaan output durasi lampu hijau

TABEL II. ATURAN DASAR (RULE BASE)

Jalur lain	Jalur yang dikontrol				
	TP	KP	CP	P	SP
TP	C	AC	S	AL	L
KP	C	AC	S	AL	L
CP	C	AC	S	AL	AL
P	C	AC	S	S	AL
SP	C	AC	AC	S	S

3) Himpunan variabel-variabel *output* durasi lampu hijau yang terdiri dari lima variabel linguistik yaitu;

- Cepat / C (0, 10, 32,5)
- Agak cepat / AC (10, 32,5, 55)
- Sedang / S (32,5, 55, 77,5)
- Agak lama / AL (55, 77,5, 100)
- Lama / L (77,5, 100,  $\infty$ )

Himpunan ini dapat digambarkan dalam bentuk grafik gambar 10.

b) *Aplikasi fungsi implikasi yang membentuk aturan dasar (rule based)*

Dari dua *input* dan satu *output* yang digunakan penulis kemudian membentuk aturan dasar (*rule based*) yang dibentuk dari implikasi antara kedua input terhadap satu *output* tersebut.

Hasilnya berupa 25 aturan dengan 25 *antecedent* atau premis dan lima *consequent* atau kesimpulan. Aturan dasar yang digunakan dirangkum ke dalam tabel II.

c) *Komposisi Aturan dengan Metode Max*

Sebelum dilakukan defuzzifikasi, harus dilakukan proses komposisi, yaitu agregasi hasil *clipping* dari semua aturan fuzzy sehingga didapatkan satu fuzzy set tunggal. Dalam proses komposisi ini digunakan Metode Max yang akan mengambil nilai maksimum aturan.

Sebagai contoh misalkan  $\mu_L(0,4)$  dan  $\mu_L(0,6)$  maka nilai proses komposisi ini akan menghasilkan  $\mu_L(0,6)$ .

d) *Defuzzifikasi*

Untuk menghasilkan keluaran *crisp* maka diperlukan tahap terakhir yaitu defuzzifikasi, dimana dalam tugas akhir ini digunakan metode *centroid* dengan cara mengambil titik pusat daerah fuzzy. Nilai *crisp* keluaran didapat dari perhitungan dengan menggunakan persamaan 3.

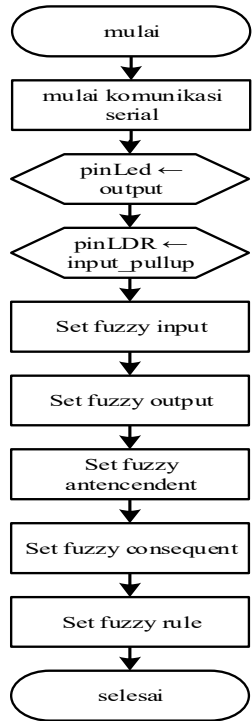
#### F. Rancangan Kode Program

Kode program dalam penelitian ini dibagi menjadi dua fungsi yaitu; *setup* dan *loop*.

Fungsi *setup* adalah fungsi yang hanya akan dieksekusi sekali saja ketika Arduino dihidupkan atau di-*reset*, sehingga dengan demikian dalam fungsi ini penulis memasukkan beberapa kode program yang bertujuan untuk inisialisasi dan pengaturan yang tidak memerlukan eksekusi berulang. Instruksi kode pada fungsi *setup* ini dapat digambarkan pada gambar 11 yang terdiri dari; pengaktifan komunikasi serial antara Arduino dengan PC melalui USB, inisialisasi pin yang berhubungan dengan I/O dan inisialisasi hal-hal yang berkaitan dengan proses logika *fuzzy*. Seperti, deklarasi fungsi keanggotaan *fuzzy input* dan *output* serta aturan atau *base rule* yang berupa *antecedent* atau premis dan *consequent* atau kesimpulan.

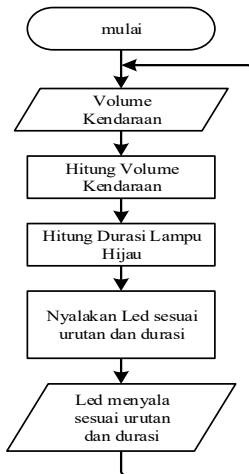
Sementara untuk fungsi *loop* penulis memasukkan beberapa instruksi yang tujuan utamanya adalah untuk mengatur lampu lalu lintas secara berulang. Rancangan alur program pada fungsi *loop* yang dibuat dapat digambarkan seperti gambar 12. Instruksi-instruksi tersebut terdiri beberapa proses seperti; Perhitungan volume kendaraan (dari *input* sensor LDR). Perhitungan durasi lampu hijau dengan menggunakan metode pengontrol proporsional atau logika *fuzzy*. Hasil perhitungan ini kemudian dicetak ke serial monitor PC melalui komunikasi serial. Pengaturan penyalan lampu lalu lintas sesuai dengan urutan dan durasi hasil perhitungan sebelumnya. *Output* yang diharapkan dari proses ini adalah lampu lalu lintas dapat menyala sesuai dengan urutan dan durasi berdasarkan jumlah volume kendaraan dan metode perhitungan yang digunakan.

Hasil perhitungan volume kendaraan ini disimpan ke dalam *array* `volJalur[]` dengan indeks 0 sampai 2 yang mewakili jalur 0, 1 dan 2.

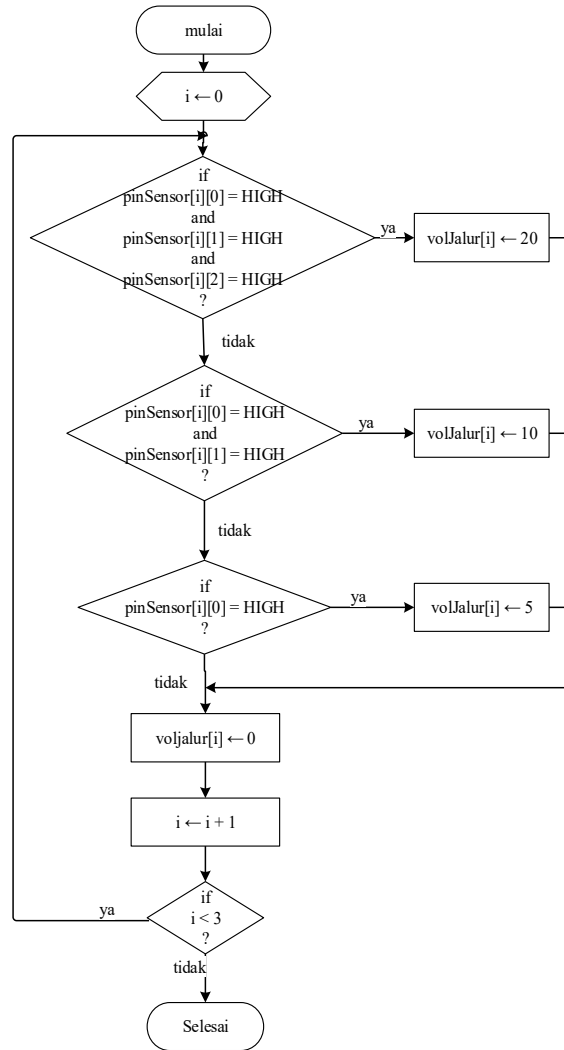


Gambar 11. Diagram alur instruksi pada fungsi setup

Perhitungan durasi lampu hijau dengan kendali proporsional ini menggunakan persamaan 4 dan menggunakan iterasi yang hasilnya disimpan ke dalam *array* *durasiLampuHijau[]*. Secara garis besar instruksinya dapat digambarkan dalam diagram alur pada gambar 15.



Gambar 12. Diagram alur instruksi pada fungsi loop

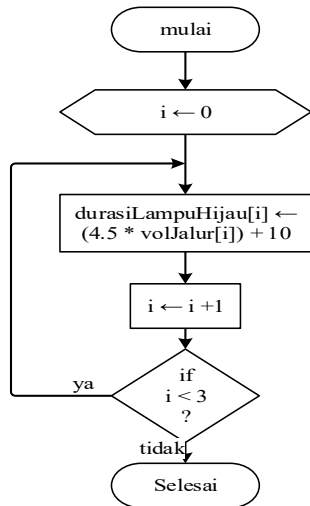


Gambar 13. Diagram alur perhitungan volume kendaraan pada tiap jalur.

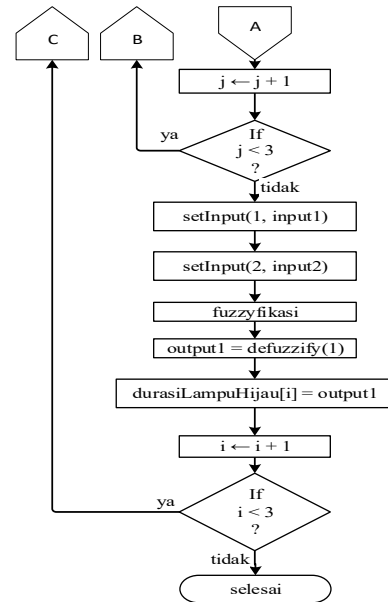
Perhitungan durasi lampu hijau dengan menggunakan logika *fuzzy* ini didesain dimana terdapat dua *input* yang digunakan yaitu; nilai volume kendaraan pada jalur yang dikontrol dan nilai volume kendaraan pada jalur lainnya dan satu *output* yaitu durasi lampu hijau, sementara untuk proses inferensinya menggunakan model Mamdani.

Alur program perhitungan durasi lampu hijau dengan menggunakan logika *fuzzy* dapat digambarkan pada gambar berikut.

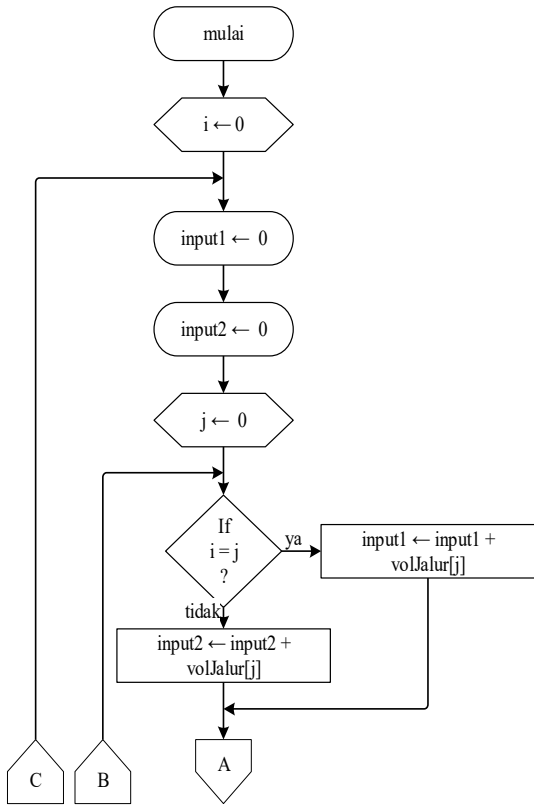
Instruksi penyalakan lampu ini dibuat untuk mengatur lampu lalu lintas agar dapat menyala sesuai dengan urutan seperti pada gambar 17.



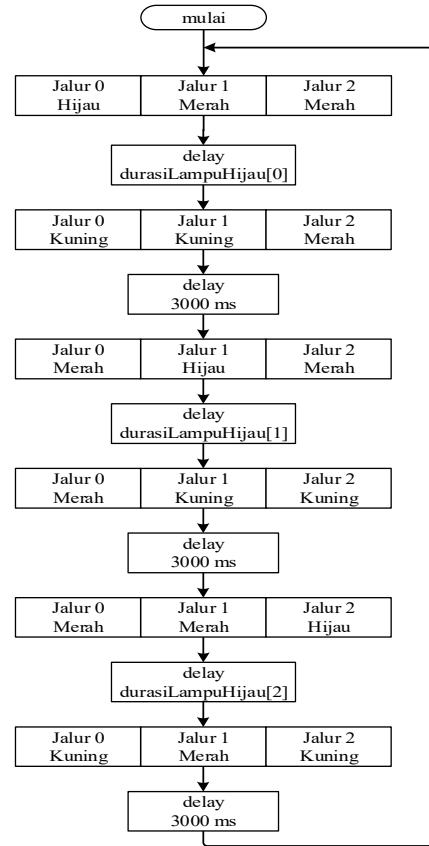
Gambar 14. Diagram alur perhitungan durasi lampu hijau dengan menggunakan kendali proporsional.



Gambar 16. Diagram alur perhitungan durasi lampu hijau dengan menggunakan logika *fuzzy* (lanjutan).



Gambar 15. Diagram alur perhitungan durasi lampu hijau dengan menggunakan logika *fuzzy*.



Gambar 17. Diagram alur urutan penyalan lampu lalu-lintas.

III. HASIL DAN PEMBAHASAN

Dari hasil rancangan *hardware* dan program yang telah dibuat maka direalisasikan alat pengujian seperti pada gambar 18. Selanjutnya untuk mengetahui perbandingan kendali lampu lalu-lintas yang menggunakan kendali proporsional dan logika *fuzzy* dilakukanlah serangkaian pengujian yang dilakukan secara langsung pada *microcontroller programming and application trainer* dan modul simulasi *traffic light*.

Selain pengujian langsung tersebut, penulis juga melakukan perhitungan teoritis untuk masing-masing kendali lampu lalu-lintas agar didapat suatu perbandingan komprehensif antara teoritis dan realisasinya.

A. Hasil Pengujian Kontrol Lampu Lalu-lintas dengan Menggunakan Kendali Proporsional

Pengujian kontrol lampu lalu-lintas dengan menggunakan kendali proporsional ini dilakukan dengan mengamati



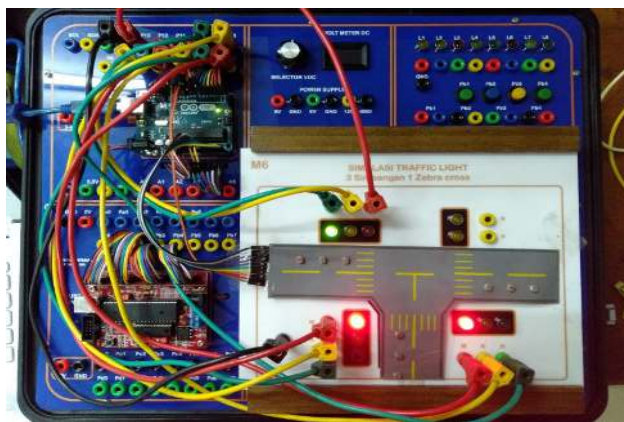
berapa jumlah kendaraan pada jalur yang dikontrol dan durasi lampu hijaunya. Hasil pengujian ini dirangkum ke dalam tabel III.

Sebagai pembanding durasi lampu hijau dari Arduino maka, dilakukan perhitungan teoritis dilakukan dengan menggunakan persamaan 4. Secara keseluruhan selisih rata-rata durasi lampu hijau antara Arduino dan perhitungan teoritis adalah sebesar 0,125 s dengan selisih terbesar 0,5 s dan terkecil 0 s.

**B. Hasil Pengujian Kontrol Lampu Lalu-lintas dengan Menggunakan Logika Fuzzy**

Pengujian kontrol lampu lalu-lintas dengan menggunakan logika fuzzy ini dilakukan dengan mengamati berapa jumlah kendaraan pada jalur yang dikontrol serta jumlah kendaraan pada jalur lainnya dan pengaruhnya pada durasi lampu hijau pada jalur yang dikontrol. Hasil pengujian ini dirangkum ke dalam tabel IV.

Sebagai pembanding durasi lampu hijau dari Arduino maka, dilakukan perhitungan dengan menggunakan aplikasi MATLAB®. Secara keseluruhan selisih rata-rata durasi lampu hijau antara Arduino dan teoritis adalah sebesar 0,5 s dengan selisih terbesar 1 s dan terkecil 0 s.



Gambar 18. Hasil rancangan hardware dan pengujian simulasi

TABEL III. HASIL PENGUJIAN DURASI LAMPU HIJAU DENGAN MENGGUNAKAN PENGONTROL PROPORSIONAL

Volume Kendaraan	Durasi Lampu Hijau [s]		Selisih [s]
	Arduino	Teoritis	
0	10	10	0
5	32	32,5	0,5
10	55	55	0
20	100	100	0
Selisih rata-rata			0,125

**C. Pembahasan**

**1) Pengujian Kontrol Lampu Lalu Lintas Dengan Menggunakan Pengontrol Proporsional**

Dari hasil pengujian kontrol lampu lalu lintas dengan menggunakan kendali proporsional (tabel II) dapat dilihat bahwa durasi lampu hijau pada jalur yang dikontrol nilainya berbanding lurus (proporsional) dengan jumlah kendaraan pada jalur yang dikontrol.

TABEL IV. HASIL PENGUJIAN DURASI LAMPU HIJAU MENGGUNAKAN LOGIKA FUZZY

Volume Kendaraan	Durasi Lampu Hijau		Selisih [s]	
	Jalur kontrol	Jalur lainnya		
0	0	11	11,3	0,3
0	5	11	11,3	0,3
0	10	12	12,2	0,2
0	15	11	11,7	0,7
0	20	13	13,3	0,3
0	25	11	11,7	0,7
0	30	12	12,2	0,2
0	40	11	11,3	0,3
5	0	19	18,9	0,1
5	5	19	18,9	0,1
5	10	19	18,9	0,1
5	15	19	18,9	0,1
5	20	21	21,4	0,4
5	25	19	18,9	0,1
5	30	19	18,9	0,1
5	40	19	18,9	0,1
10	0	43	43,8	0,8
10	5	43	43,8	0,8
10	10	43	43,8	0,8
10	15	43	43,8	0,8
10	20	43	43,8	0,8
10	25	43	43,8	0,8
10	30	43	43,8	0,8
10	40	32	32,5	0,5
20	0	103	104	1
20	5	103	104	1
20	10	102	103	1
20	15	102	103	1
20	20	90	90,5	0,5
20	25	77	77,5	0,5
20	30	77	77,5	0,5
20	40	55	55	0
Selisih rata-rata				0,5

Hasil pengujian ini telah sesuai dengan rancangan yang dibuktikan dari selisih rata-rata antara nilai durasi dari Arduino dan teoritis yang hanya sebesar 0,125 s.

Pada hasil pengujian ini didapati bahwa durasi lampu hijau akan bertambah dengan proporsional saat jumlah kendaraan pada jalur yang dikontrol berada dalam *extension unit* yakni pada saat jumlah kendarannya nol sampai 20. Hal lainnya yang didapat dari pengujian ini adalah; berapa pun jumlah kendaraan pada jalur lainnya tidak akan

mempengaruhi durasi lampu hijau pada jalur yang dikontrol.

## 2) Pengujian Kontrol Lampu Lalu Lintas Dengan Menggunakan Logika Fuzzy

Dari hasil pengujian kontrol lampu lalu lintas dengan menggunakan logika *fuzzy* (tabel IV) dapat dilihat bahwa durasi lampu hijau pada jalur yang dikontrol nilainya tergantung pada jumlah kendaraan pada jalur yang dikontrol dan jalur lainnya.

Jika diperhatikan durasi lampu hijau pada jalur yang dikontrol akan bertambah dengan seiring bertambahnya jumlah kendaraan pada jalur yang dikontrol, namun jika jumlah kendaraan pada jalur yang dikontrol meningkat sementara jumlah kendaraan pada jalur lainnya bertambah maka durasi lampu hijau pada jalur yang dikontrol akan berkurang. Hal ini bersesuaian dengan rancangan aturan dasar (*rule based*) pada tabel II.

Dengan demikian dari hasil pengujian ini didapatkan durasi lampu hijau akan bergantung pada jumlah kendaraan baik di jalur yang dikontrol maupun di jalur lainnya.

Perhitungan secara teoritis durasi lampu hijau pada logika *fuzzy* dengan menggunakan bantuan aplikasi MATLAB memperlihatkan hasil yang tidak terlalu berbeda dengan hasil implementasi pada Arduino. Hal ini dibuktikan dengan selisih durasi rata-ratanya hanya sekitar 0,5 s.

## IV. KESIMPULAN DAN SARAN

### A. Kesimpulan

Penggunaan sistem kendali proporsional pada lampu lalu lintas akan memberikan pengaturan durasi lampu hijau yang linear terhadap banyaknya jumlah kendaraan. Sistem kendali proporsional tidak memperhitungkan jumlah kendaraan pada jalur lainnya, melainkan hanya pada jalur yang dikontrol, sehingga kurang mampu menangani masalah penumpukan kendaraan di jalur lainnya. Sistem kendali lampu lalu-lintas menggunakan logika *fuzzy* akan memberikan variasi lampu hijau yang dapat ditentukan *rule-based*-nya, sehingga durasinya dapat bervariasi dengan berbagai kemungkinan nilai *input*. Sistem kendali lampu lalu-lintas dengan menggunakan logika *fuzzy* bisa diprogram agar dapat memperhitungkan nilai jumlah kendaraan pada jalur yang dikontrol dan jalur lainnya, sehingga meskipun lebih rumit, permasalahan penumpukan kendaraan pada jalur lainnya dapat ditangani.

### B. Saran

Pengaturan lampu lalu lintas dengan menggunakan logika *fuzzy* perlu penentuan *rule-based* yang tepat. Penulis menyadari *rule-based* yang dibuat ini masih sangat sederhana sehingga ke depannya perlu ditambahkan *rule-based* yang bisa meng-*cover* permasalahan lainnya yang mungkin timbul seperti, peningkatan jumlah kendaraan yang sangat besar.

## V. KUTIPAN

- [1] Atmaja, Bagus Tris, dan Budiman Putra AR. 2015. Implementasi Sistem Fuzzy Untuk Pengaturan Lampu Lalu Lintas Berdasarkan Kepadatan Arus Kendaraan. Surabaya : Jurusan Teknik Fisika, Fakultas Teknologi, Industri Institut Teknologi Sepuluh Noverber.
- [2] Munir, Rinaldi. 2018. Sistem Inferensi Fuzzy. Bandung: Teknik Informatika–STEI ITB.

- [3] Prawira, Bobby Yoga. 2018. Efisiensi Pencahayaan Ruangan Perkuliahan dengan Logika Fuzzy. Manado: Jurusan Teknik Elektro, Fakultas Teknik, Universitas Sam Ratulangi.
- [4] Prasetyo, Erwan Eko. 2016. Perbandingan Kinerja Pengendali Lampu Lalu Lintas Metode *Fuzzy* Tipe Sugeno dengan Metode Waktu Tetap. Yogyakarta. Teknik Kedirgantaraan, STTKD.
- [5] M. G. H. Kulkarni and M. P. G. Waingankar, Fuzzy Logic Based Traffic Light Controller, no. August, pp. 8 11, 2007.
- [6] Muhammad Ikhwan, Said Munzir, Nurmaulidar. 2016. Sistem Kendali Proporsional, Integral, Dan Derivatif (PID) Pada Persamaan Panas.
- [7] Rahmat Taufik, Supriyono, Sukarman. 2008. Rancang Bangun Simulator Kendali Lampu Lalu Lintas Dengan Logika *Fuzzy* Berbasis Mikrokontroler. Sekolah Tinggi Teknologi Nuklir – BATAN.
- [8] Kusumadewi, Sri, 2002, Analisis & Desain Sistem *Fuzzy* Menggunakan Toolbox MATLAB. Yogyakarta, GRAHA ILMU.
- [9] Muchammad Abrori & Amrul Hinung Prihamayu. 2015. Aplikasi Logika *Fuzzy* Metode Mamdani Dalam Pengambilan Keputusan Penentuan Jumlah Produksi.
- [10] Indahdin, Jemala, Simulasi Pengaturan Lampu Lalulintas Berdasarkan Logika *Fuzzy* Menggunakan Borland Delphi 7.0, Universitas Mercu Buana, Jakarta: 2008.



Penulis bernama lengkap Twi Rahmat Mulya A.S, anak kedua dari empat bersaudara. Anak dari Abdul Sakkar (ayah) dan Sukmawati (ibu). Lahir di Kabupaten Soppeng Sulawesi Selatan pada tanggal 5 Mei 1992. Yang pada saat ini beralamat Mahakeret Timur Lingkungan IV Kecamatan Wenang Manado. Sekolah pertama TK Biringkaya Makassar (1997-1998), SD Inpres Daya Makassar (1998-2004), SMP Negeri 25 Makassar (2004-2006), SMP Negeri 1 Tagulandang (2006-2007), SMK Negeri 2 Manado.

Tahun 2012, melanjutkan studi di Fakultas Teknik, Jurusan Teknik Elektro Universitas Sam Ratulangi Manado dengan konsentrasi minat Teknik Kendali. Melaksanakan Kerja Praktek di BMKG Manado lalu Kuliah Kerja Terpada Angkatan 119 tahun 2018 di Kelurahan Pampusungan, Kecamatan Lembeh Selatan Kota Bitung.

Penulis aktif dalam organisasi kampus, Anggota Himpunan Mahasiswa Elektro, Anggota Badan Tadzkir Fakultas Teknik, Pengurus Control Engineering Community, Pengurus Senat Mahasiswa Fakultas Teknik, Wakil Ketua UKM Seni Fakultas Teknik “Pabrik Aesthetic”, Ketua UKM Korps Sukarela Palang Merah Indonesia Unit UNSRAT, Koordinator Wilayah Forum Komunikasi Himpunan Mahasiswa Elektro Indonesia Wilayah XVI. Penulis pernah meraih peringkat 1 Lomba Pertolongan Pertama se-Sulawesi Utara, mengikuti pelatihan Search And Rescue Nasional dan menjadi Anggota Satuan Tanggap Darurat Bencana Provinsi Sulawesi Utara.