

IMPLEMENTASI ALGORITMA LOGIKA FUZZY UNTUK SISTEM PENGATURAN LAMPU LALU LINTAS MENGGUNAKAN MIKROKONTROLER

Implementation of Fuzzy Logic Algorithm for Traffic Lights Control System Using Microcontroller

Heri Prasetyo^{1,*}, Utis Sutisna¹

¹Program Studi Teknik Elektro

Sekolah Tinggi Teknik Wiworotomo Purwokerto

Jl. Semingkir No.1 Purwokerto 53134 Telp. (0281) 632870, 626266

*email : heryqwerty@gmail.com

ABSTRAK

Pada umumnya sistem pengaturan lampu lalu-lintas hanya melakukan pengaturan berdasarkan waktu yang tetap. Pada kenyataannya tingkat kedatangan kendaraan pada persimpangan jalan tidak selalu sama sehingga tentu saja tingkat kemacetan pada persimpangan jalan tidak dapat dikendalikan dengan baik. Pada penelitian ini dibuat simulasi sistem kendali otomatis yang dapat melakukan pengaturan lampu lalu lintas berdasarkan tingkat kedatangan kendaraan dimana sistem pengaturannya tidak konstan tetapi mengikuti tingkat kedatangan kendaraan. Metode yang digunakan untuk mengatur lamanya waktu ini adalah algoritma logika fuzzy dengan penalaran fuzzy metode Mamdani menggunakan software Matlab. Hasil pengujian berdasarkan data hasil simulasi logika fuzzy pada toolbox fuzzy MatLab menunjukkan bahwa algoritma logika fuzzy dapat digunakan untuk memenuhi tujuan pengaturan lalu lintas secara optimal, yaitu durasi waktu yang diberikan didasarkan pada tingkat kedatangan kendaraan. Semakin tinggi tingkat kedatangan kendaraan maka semakin lama durasi waktu yang diberikan, dan semakin rendah tingkat kedatangan maka semakin sebentar durasi waktu yang diberikan.

Kata kunci : Lampu lalu lintas, logika fuzzy, mikrokontroler

ABSTRACT

In general, the control system of traffic lights only perform control based on a fixed time. In fact, the arrival rate of vehicles at the intersection of the road is not always the same so of course congestion at the crossroads can not be controlled. This study presents a simulation control system which can perform automatic traffic light control based on vehicle arrival rate. In this control, the duration of the flame is not constant but follows the arrival rate of vehicles. The method used to control the flame duration is fuzzy logic algorithm with the Mamdani fuzzy reasoning using Matlab software. Test results based on the simulation of fuzzy logic in fuzzy MatLab toolbox show that the fuzzy logic algorithm can be used to meet the goal of optimal traffic control, the duration of a given time based on the arrival rate of vehicles. The higher rate of arrival of the vehicle, the longer the duration of a given time, and the lower rate of the arrival of the vehicle, the more briefly given time duration.

Key-word : Traffic lights, fuzzy logic, microcontroller

PENDAHULUAN

Dengan perkembangan teknologi yang begitu pesat khususnya dibidang kendaraan bermotor maka berbagai macam teknologi kendaraan diciptakan untuk memudahkan manusia dalam menjalankan segala macam bentuk aktivitas. Salah satu masalah yang akan

timbul akibat bertambahnya jumlah kendaraan bermotor adalah kemacetan akibat dari tidak disiplinnya para pengendara bermotor dalam berlalu lintas. Kemacetan yang muncul tersebut dapat disebabkan dari beberapa faktor, salah satunya adalah faktor pengaturan lampu lalu lintas yang kurang baik.

Salah satu solusi untuk masalah tersebut diatas adalah dengan **membuat** sistem pengontrolan lampu lalu lintas secara otomatis yang dapat menyesuaikan dengan jumlah kedatangan kendaraan dari setiap jalur. Sistem pengaturan durasi lamanya nyala lampu akan mengikuti perubahan tingkat kedatangan kendaraan pada setiap jalur persimpangan jalan. Jalur yang volume kendaraannya lebih padat akan mendapatkan nyala lampu lebih lama, sehingga akan mengurangi tingkat kedatangan untuk terjadinya kemacetan.

LOGIKA FUZZY

Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang input kedalam suatu ruang output. *Fuzzy* dinyatakan dalam derajat dari suatu keanggotaan dan derajat dari kebenaran (Kusumadewi, 2004).

Logika *fuzzy* meniru cara berpikir manusia dengan menggunakan konsep sifat kesamaran suatu nilai. Konsep logika *fuzzy* pertama kali diperkenalkan oleh Professor Lotfi A. Zadeh dari Universitas California, pada bulan Juni 1965 untuk menyatakan kelompok/himpunan yang dapat dibedakan dengan himpunan lain berdasarkan derajat keanggotaan dengan batasan yang tidak begitu jelas (samar), tidak seperti himpunan klasik yang membedakan keanggotaan himpunan menjadi dua, himpunan anggota atau bukan anggota.

Ada beberapa alasan mengapa orang menggunakan logika *fuzzy* (Kusumadewi, 2002), yaitu :

1. Konsep logika *fuzzy* mudah dimengerti karena konsep matematis yang mendasari penalaran *fuzzy* sangat sederhana dan mudah dimengerti.
2. Logika *fuzzy* sangat fleksibel.
3. Logika *fuzzy* memiliki toleransi terhadap data-data yang tidak tepat.
4. Logika *fuzzy* mampu memodelkan fungsi-fungsi non linier yang sangat kompleks.
5. Logika *fuzzy* didasarkan pada bahasa alami.

Sistem Inferensi *Fuzzy* (*Fuzzy Inference System* atau FIS) merupakan suatu kerangka komputasi yang didasarkan pada teori himpunan, aturan *fuzzy* berbentuk IF-THEN dan penalaran *fuzzy* (Kusumadewi, 2006).

Metode inferensi Mamdani menggunakan fungsi keanggotaan *fuzzy* pada bagian keluarannya. Sehingga setelah proses aturan telah diterapkan, terdapat himpunan *fuzzy* yang harus didefuzzifikasi. Umumnya proses defuzzifikasi berlangsung lebih lambat akibat proses komputasi pada keluarannya dan untuk mendapatkan output diperlukan 4 tahapan yaitu :

1. Pembentukan himpunan *fuzzy*.

Pada proses fuzzifikasi, langkah yang pertama adalah menentukan variabel *fuzzy* dan himpunan *fuzzy*-nya.

2. Aplikasi fungsi implikasi pada metode mamdani.

Fungsi implikasi yang digunakan adalah *min*.

3. Komposisi Aturan.

Metode inferensi yang digunakan adalah metode *max*.

4. Penegasan (defuzzifikasi).

MIKROKONTROLER ATMEGA16

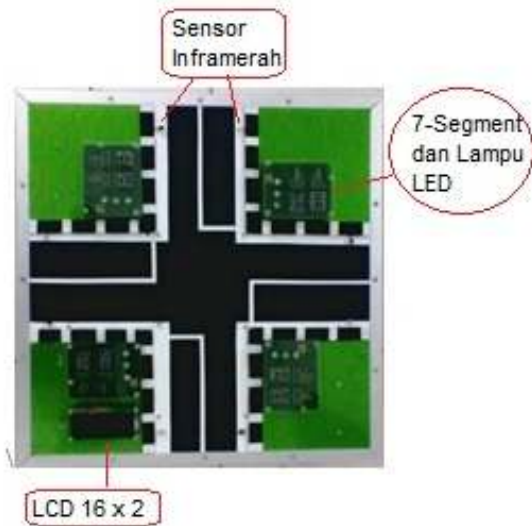
Mikrokontroler ATmega16 merupakan mikrokontroler keluaran dari *Atmel Corporation*. Mikrokontroler tipe ini termasuk dalam jenis AVR (*Alf and Vegard's Risc processor*). Mikrokontroler AVR memiliki arsitektur RISC 8 bit, dimana semua instruksi dikemas dalam kode 16-bit dan sebagian besar instruksi dieksekusi dalam satu siklus *clock*. Perbedaan ini terjadi karena arsitektur yang berbeda, AVR sudah menggunakan teknologi RISC (*Reduced Instruction Set Computing*) sedangkan MCS51 menggunakan tipe CISC (*Complex Instruction Set Computing*) (Atmel Corporation, 2010).

AVR juga mempunyai *In-System Programmable Flash on-chip* yang memungkinkan memori program untuk diprogram ulang dalam sistem menggunakan hubungan serial SPI. Selain itu, ATmega16 mempunyai *throughput* mendekati 1 MIPS per MHz sehingga perancang sistem dapat mengoptimasi konsumsi daya dan kecepatan proses (Datasheet ATmega16).

Adapun fitur yang dimiliki ATmega16 adalah : 8 KByte *Flash Program*, 512 Bytes EEPROM, 512 Bytes Internal SRAM, 2 Timer 8 bit dan 1 Timer 16 bit, *Analog to Digital Converter* (ADC) dan USART.

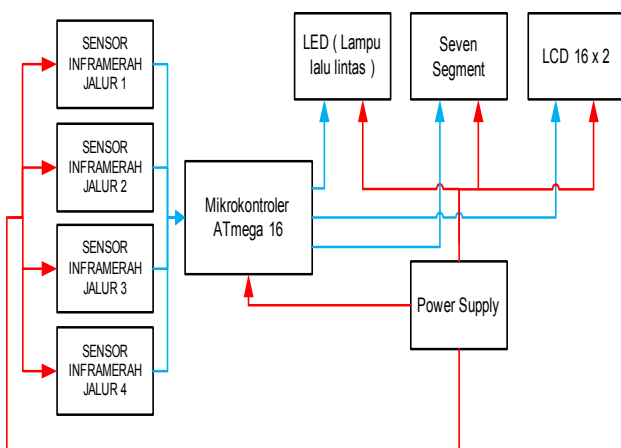
PERANCANGAN DAN PEMBUATAN SISTEM

Perancangan dan pembuatan alat simulasi ini terdiri dari dua bagian, yaitu perangkat keras dan perangkat lunak.

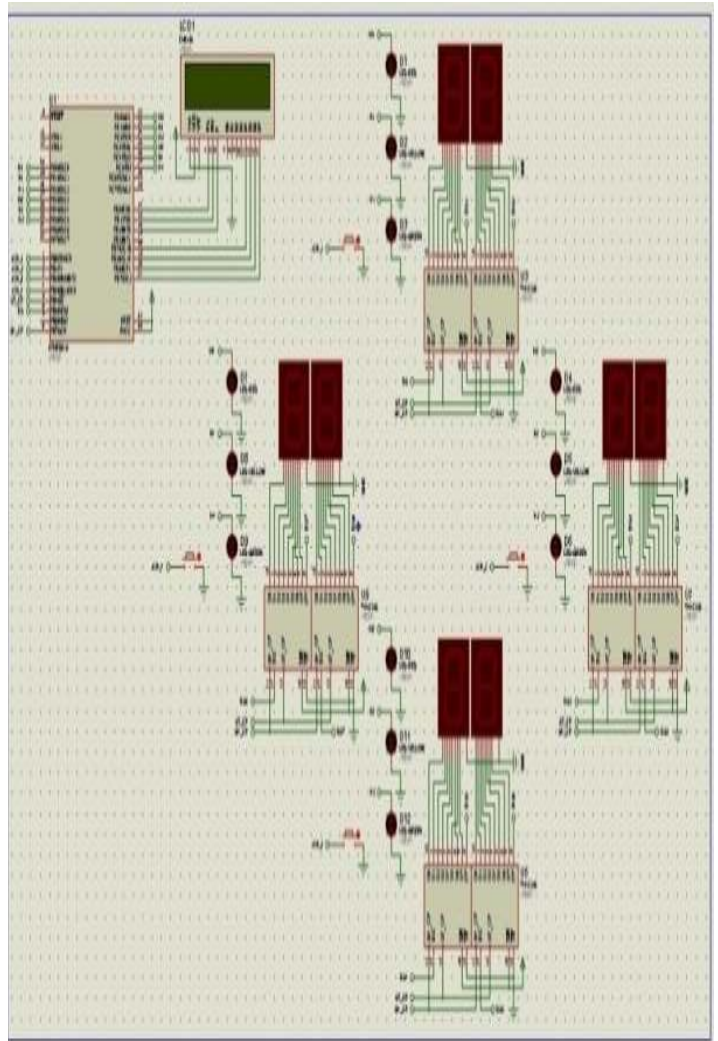


Gambar 1. Bentuk miniatur sistem

Perancangan perangkat keras terdiri dari mekanik miniatur untuk simulasi lalu lintas persimpangan jalan (Gambar 1), mikrokontroler ATmega16, rangkaian BCD (*Biner Code Desimal*) 7-segment, rangkaian komparator dengan sensor inframerah dan komponen pendukung lainnya. Komponen yang digunakan ditunjukkan pada diagram blok sistem pada Gambar 2. Sedangkan rangkaian skematik sistem ditunjukkan pada Gambar 3.

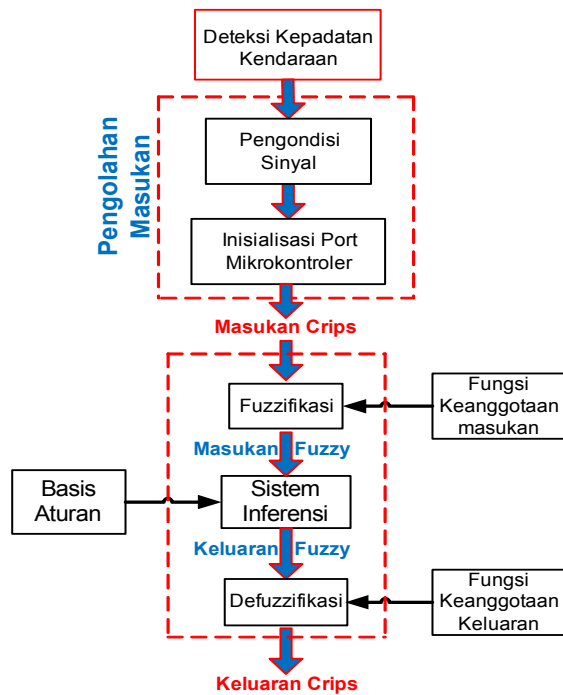


Gambar 2. Diagram blok sistem



Gambar 3. Rangkaian *schematic* keseluruhan sistem

Perancangan perangkat lunak terdiri dari perancangan algoritma kontrol logika *fuzzy* dan implementasi pemrograman dengan bahasa C.



Gambar 4. Diagram jalannya penelitian

Perancangan pengaturan sistem lampu lalu lintas dengan logika fuzz dalam penelitian ini ditunjukkan pada Gambar 4.

A. Deteksi Kedatangan

Pada langkah ini digunakan sensor inframerah yang berfungsi sebagai pendeteksi kedatangan kendaraan yang memasuki persimpangan melalui jalur tertentu. Sensor yang digunakan berjumlah 1 buah disetiap jalur.

B. Pengolahan Masukan

Tahap pengolahan masukan meliputi pengondisi sinyal dan inisialisasi port mikrokontroler.

1. Pengondisi Sinyal

Proses ini menggunakan rangkaian komparator sebagai rangkaian pengondisi sinyal dengan IC LM324. LM324 merupakan IC Op-Amp (*Operational Amplifier*) yang digunakan untuk melakukan penguatan terhadap sinyal dari sensor deteksi.

2. Inisialisasi Port Mikrokontroler

Hasil dari setiap proses pengondisi sinyal menjadi masukan bagi ATmega16. Oleh karena itu, dalam pengoperasiannya perlu di-set untuk menentukan *port* dan *pin* mana saja dari

port yang tersedia untuk difungsikan sebagai masukan dan sebagai keluaran.

C. Fuzzifikasi

Fuzzifikasi adalah proses yang dilakukan untuk mengubah variabel nyata (*crisp*) menjadi bentuk variabel *fuzzy*. Hal ini ditujukan agar masukan kontroler *fuzzy* bisa dipetakan menuju jenis yang sesuai dengan himpunan *fuzzy*. Dalam penelitian ini digunakan fungsi keanggotaan tipe *triangular* dan *trapezoidal* dikarenakan komputerisasi yang sederhana serta efisien. Pada Tabel 1 ditunjukkan variabel dan himpunan *fuzzy* yang digunakan.

Tabel 1. Variabel dan himpunan fuzzy

Fungsi	Variabel	Himpunan	Domain	Fungsi Keanggotaan
Input	Cacahan	Sangat sedikit	[0, 3]	Bahu kiri
		Sedikit	[1, 5]	Segitiga
		Normal	[3, 7]	Segitiga
		Banyak	[5, 9]	Segitiga
		Sangat banyak	[7, 10]	Bahu kanan
	Perubahan Cacahan	Sangat berkurang	[-10, -3]	Bahu kiri
		Berkurang	[-6, 0]	Segitiga
		Tetap	[-3, 3]	Segitiga
		Bertambah	[0, 6]	Segitiga
		Sangat bertambah	[3, 10]	Bahu kanan
Output	Waktu	Sangat sebentar	[5, 12]	Bahu kiri
		Sebentar	[7, 17]	Segitiga
		Sedang	[12, 22]	Segitiga
		Lama	[17, 27]	Segitiga
		Sangat lama	[22, 30]	Bahu kanan

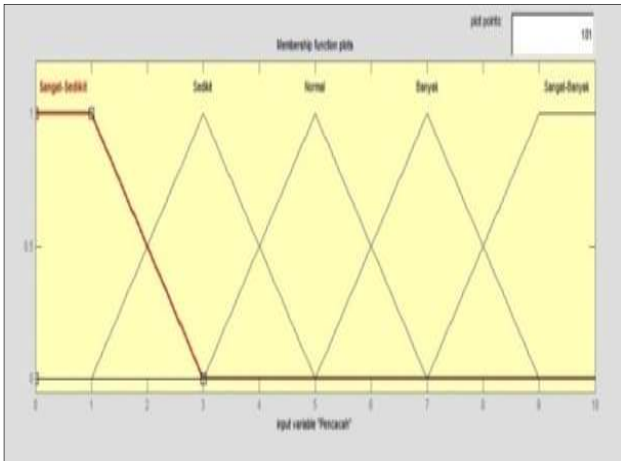
Proses ini memanfaatkan pembacaan data dari sensor inframerah yang kemudian diolah dengan perhitungan sebagai berikut.

$$Cacahan = jumlah\ deteksi$$

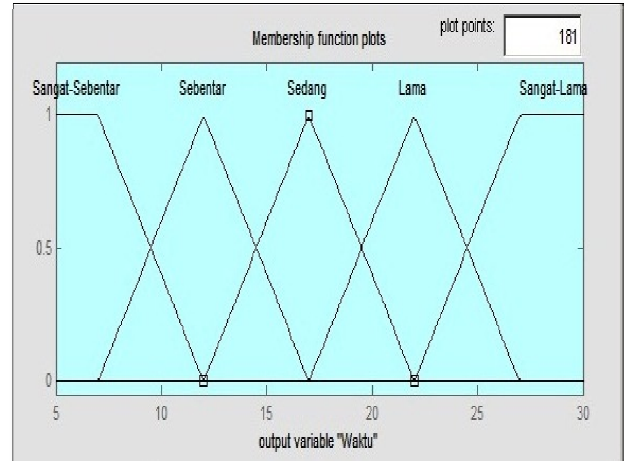
$$Perubahan\ cacahan = cacahan - cacahan\ sebelumnya$$

$$Waktu = Output\ fuzzy\ hasil\ simulasi\ MatLab$$

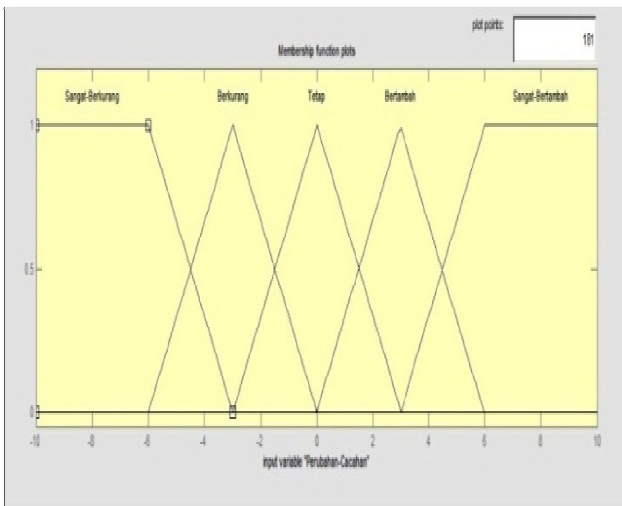
Himpunan masukan logika *fuzzy* ditunjukkan pada Gambar 5 dan Gambar 6.



Gambar 5. Himpunan fuzzy masukan cacahan



Gambar 7. Himpunan fuzzy keluaran waktu



Gambar 6. Himpunan fuzzy masukan perubahan cacahan

Himpunan keluaran logika fuzzy merupakan representasi hasil dari keputusan dalam bentuk variasi nilai waktu untuk masing-masing jalur. Himpunan untuk keluaran logika fuzzy ditunjukkan pada Gambar 7.

D. Sistem Inferensi

Inferensi disini dapat disebut pula sebagai aplikasi fungsi implikasi. Dalam penelitian ini digunakan metode Mamdani. Penentuan basis aturan logika fuzzy didasarkan pada pengalaman dan disusun dalam bentuk penalaran Jika-Maka (*If-Then*). Metode inferensi yang dipergunakan ialah metode min-max. Sistem penalaran ditulis dalam sebuah tabel FAM (*Fuzzy Associative Memory*) yang ditunjukkan pada Tabel 2.

Tabel 2. FAM (*Fuzzy Associative Memory*)

		Perubahan Cacahan				
		Sangat Berkurang	Berkurang	Tetap	Bertambah	Sangat Bertambah
Cacahan	Input					
	Sangat Sedikit	Sangat Berkurang	Sangat Berkurang	Sangat Tetap	Sangat Bertambah	Sangat Sangat Bertambah
	Sedikit	Sangat Sebentar	Sebentar	Sebentar	Sebentar	Sebentar
	Normal	Sangat Sebentar	Sedang	Sedang	Sedang	Lama
	Banyak	Lama	Lama	Lama	Lama	Sangat Lama
	Sangat Banyak	Sangat Lama	Sangat Lama	Sangat Lama	Sangat Lama	Sangat Lama

Dari Tabel 2. dapat dibuat 25 rule atau aturan fuzzy yang digunakan dalam proses inferensi. Basis aturan tersebut ditunjukkan pada Tabel 3.

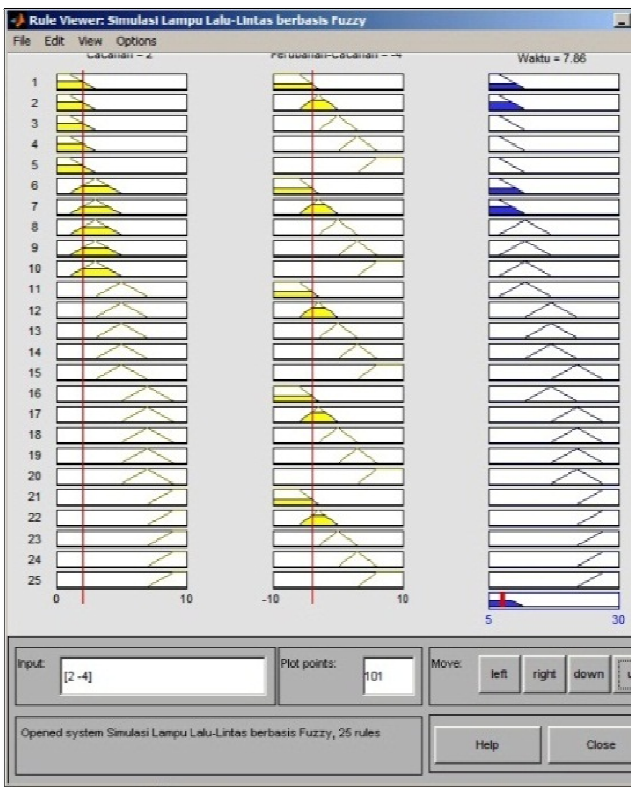
Tabel 3. Basis aturan fuzzy

No	Aturan
1	If (Cacahan is Sangat-Sedikit) and (Perubahan-Cacahan is Sangat-Berkurang) then (Waktu is Sangat-Sebentar) (1)
2	If (Cacahan is Sangat-Sedikit) and (Perubahan-Cacahan is Berkurang) then (Waktu is Sangat-Sebentar) (1)
.....
25	If (Cacahan is Sangat-Banyak) and (Perubahan-Cacahan is Sangat-Bertambah) then (Waktu is Sangat-Lama) (1)

E. Defuzzifikasi

Defuzzifikasi adalah proses untuk mengubah keluaran fuzzy menjadi keluaran crisp. Hasil defuzzifikasi digunakan untuk mengatur lamanya waktu untuk masing-masing jalur. Metode defuzzifikasi yang digunakan adalah *Center of Gravity* (COG).

F. Simulasi Fuzzy Inference System (FIS)



Gambar 8. Contoh tampilan uji coba simulasi FIS

Pada Gambar 8 diperlihatkan contoh tampilan uji coba simulasi FIS.

Keterangan:

- Interval [0,10] menunjukkan semesta pembicaraan untuk cacahan.
- Interval [-10,10] menunjukkan semesta pembicaraan untuk perubahan cacahan.
- Interval [5,30] menunjukkan semesta pembicaraan untuk variabel waktu.

Hasil untuk contoh kasus yang ditunjukkan pada Gambar 8 adalah sebagai berikut.

Input

Cacahan : 2

Perubahan Cacahan : -4

Output

Waktu : 7.86 detik

Kolom pertama pada Gambar 8 menunjukkan tingkat keanggotaan cacahan pada variabel **cacahan**, kolom kedua menunjukkan tingkat keanggotaan **perubahan cacahan** pada variabel perubahan cacahan, dan kolom ketiga menunjukkan konsekuensi dari fungsi implikasi aturan yang sesuai dengan kondisi tersebut.

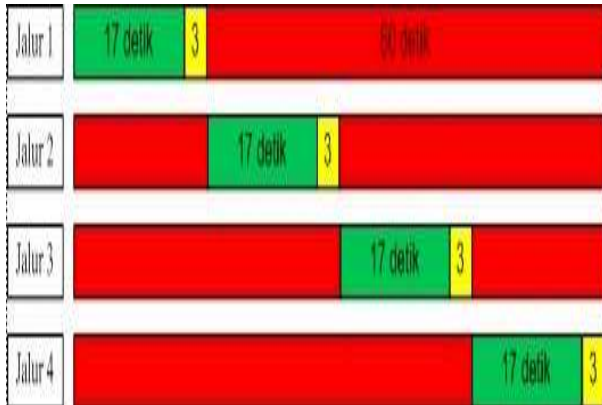
Baris terakhir dan kolom terakhir, menunjukkan gabungan daerah fuzzy dari masing-masing aturan. Dari gambar tersebut, garis vertikal merah tebal pada variabel **waktu** menunjukkan nilai waktu yaitu sebesar 7.86 detik. Nilai waktu tersebut termasuk ke dalam keanggotaan himpunan fuzzy **sangat sebentar** sekaligus **sebentar**, tetapi jika melihat dari tingkat keanggotannya maka dapat dikatakan nilai waktu tersebut masuk ke dalam keanggotaan himpunan fuzzy **sangat sebentar**.

HASIL PENGUJIAN DAN PEMBAHASAN

Pada penelitian ini proses pergantian nyala lampu lalu lintas dimulai dengan kondisi awal lampu hijau menyala pada jalur 1 dan lampu merah pada jalur 2, 3 dan 4. Pergantian lampu lalu lintas berjalan secara terus menerus dan bekerja layaknya lampu lalu lintas dengan keadaan yang sebenarnya.

Langkah pengujian dapat dijelaskan sebagai berikut :

1. Kondisi awal yaitu menghidupkan lampu hijau pada jalur 1 dan lampu merah pada jalur yang lain dengan 7-segment menunjukkan angka 60 detik untuk lampu merah menyala, 17 detik untuk lampu hijau menyala dan 3 detik untuk lampu kuning menyala. Pada saat lampu kuning menyala, 7-segment tidak diaktifkan hanya lampu yang menyala, untuk tampilan *countdown* 7-segment dimatikan.



Gambar 9. Siklus waktu lampu lalu lintas

2. Kondisi pengaturan otomatis yaitu disaat lampu merah menyala di salah satu jalur dan mendeteksi kedatangan kendaraan pada siklus pertama (cacahan) dan siklus kedua (perubahan cacahan). Sehingga nyala lampu hijau pada siklus kedua tergantung dari jumlah deteksi kedatangan pada siklus pertama dan siklus kedua. Untuk lamanya nyala lampu merah pada setiap jalur akan menyesuaikan terhadap nyala lampu hijau.
3. Sensor inframerah akan memberikan logika 0 pada saat mendeteksi obyek (*active low*) dan menyimpannya di register masing-masing jalur yang telah ditentukan. Dengan demikian pada saat jalur akan mendapat giliran lampu hijau menyala, mikrokontroler sudah dapat memproses berapa lamanya waktu lampu hijau menyala yang akan dikeluarkan pada jalur tersebut.

Tabel 4. Hasil pengujian sistem

No	Jalur yang dilintasi	Input		Output
		Cacahan	Perubahan Cacahan	Waktu (detik)
1	Jalur 1	7	3	22
		3	-4	8
		3	3	12
		5	-1	17
2	Jalur 2	6	-5	16
		3	-3	7
		8	5	24
		4	-4	13
3	Jalur 3	2	0	10
		5	3	17
		4	-1	13
		9	5	27
4	Jalur 4	2	-5	10
		3	-1	11
		7	4	22
		10	3	27

Dari hasil pengujian pada Tabel 4, terlihat bahwa sistem mampu menghasilkan keluaran berupa lamanya waktu hijau yang bervariasi tergantung pada besarnya masukan yang ada. Hal ini menunjukkan bahwa sistem kendali fuzzy bisa menyesuaikan keadaan keluaran berupa lama lampu hijau menyala sesuai dengan jumlah kedatangan kendaraan pada suatu jalur.

Nilai masukan adalah jumlah deteksi kedatangan pada jalur dengan nilai sembarang yaitu antara 0 sampai 10. Proses pendeteksian (*scan*) masukan kedatangan pada cacahan dilakukan pada keadaan lampu merah menyala, setelah lampu merah padam proses pendeteksian akan reset dan nilai pendeteksian akan tersimpan dalam mikrokontroler. Proses kerja ini juga berlaku untuk proses pendeteksian untuk menghasilkan perubahan cacahan.

Hasil dari proses pendeteksian pada cacahan dan perubahan cacahan adalah pada waktu nyala lampu hijau.

Lamanya waktu lampu hijau menyala di suatu jalur tergantung dari jumlah kepadatan pada cacahan dan perubahan cacahan. Semakin besar jumlah kepadatan di suatu jalur maka semakin lama lampu hijau di jalur tersebut.

KESIMPULAN

Sistem pengaturan lampu lalu lintas dengan algoritma logika *fuzzy* berdasarkan metode Mamdani memberikan nilai lamanya waktu lampu hijau menyala tergantung dari jumlah kedatangan suatu jalur pada siklus pertama dan siklus kedua lampu. Semakin besar jumlah kedatangan di suatu jalur maka semakin lama lampu hijau di jalur tersebut, begitu juga sebaliknya. Sistem lampu lalu lintas dengan algoritma logika *fuzzy* lebih efektif dibandingkan dengan sistem lalu lintas konvensional. Hal ini dikarenakan sistem pengaturan lalu lintas dengan algoritma logika *fuzzy* dapat menyesuaikan dengan kedatangan yang sedang terjadi pada suatu persimpangan jalan. Hal ini berbeda dengan sistem pengaturan lampu lalu lintas konvensional yang merupakan sistem dengan waktu siklus yang telah ditetapkan (*preset cycle time*).

DAFTAR PUSTAKA

- Atmel Corporation, 2010, *8-bit Microcontroller with 16K Bytes In-System Programmable Flash (ATmega16, ATmega16L)*, San Jose: Atmel Corporation.
- Danuri, Muhamad, 2008, *Traffic Management Center dengan Logika Fuzzy dan Sensor Kamera*, Infokam Nomor II Th.IV.
- Kusumadewi, Sri dan Hartati, Sri, 2006, *Neuro fuzzy Integrasi sistem fuzzy dan jaringan syaraf*, Graha Ilmu, Yogyakarta.
- Kusumadewi, Sri dan Purnomo, Hari, 2004, *Aplikasi Logika Fuzzy untuk Pendukung Keputusan*, Graha Ilmu, Yogyakarta.
- Kusumadewi, Sri, 2002, *Analisis dan Desain Sistem Fuzzy Menggunakan Toolbox Matlab*, Graha Ilmu, Yogyakarta.
- Kusumadewi, Sri, 2003, *Artificial Intelligence*, Graha Ilmu, Yogyakarta.
- Irawanto, Bambang dan Kurniawan, Desfri, 2010, *Penerapan Sistem Inferensi Metode Min-Max dalam Logika Fuzzy untuk Pengaturan Traffic Light*, Jurnal Sains & Matematika (JSM) Vol. 18, No. 1.
- Parmonangan S, Novan, 2012, *Aplikasi Fuzzy Logic Controller pada Pengontrolan Lampu Lalu Lintas*, Makalah IF4058 Topik Khusus Informatika I – Sem. II, Institut Teknologi Bandung.

Taufik, Rahmat, dkk., 2008, *Rancang Bangun Simulator Kendali Lampu Lalu lintas dengan Logika Fuzzy Berbasis Mikrokontroler*, Seminar Nasional IV SDM Teknologi Nuklir Yogyakarta.

Winoto, Ardi, 2008, *Mikrokontroler AVR ATmega8/32/16/8535 dan Pemrogramannya dengan Bahasa C pada WinAVR*, Informatika, Bandung.

Zulfikar, dkk., 2011, *Perancangan Pengontrolan Traffic Light Otomatis*, Jurnal Rekayasa Elektrika Vol. 9, No. 3.