

# PENGATURAN KECEPATAN MOTOR DALAM MEMPERTAHANKAN BATAS TEPI BADAN ROBOT *LINE FOLLOWER* TERHADAP *LINE* MENGGUNAKAN *FUZZY LOGIC CONTROLLER*

Imam Fauzi<sup>1</sup>, Ir. Erni Y, MT., Dr.<sup>2</sup>, Ir. Purwanto, MT.<sup>3</sup>

<sup>1</sup>Mahasiswa Teknik Elektro, <sup>2,3</sup>Dosen Teknik Elektro, Universitas Brawijaya

E-mail: [Imamfauji04@gmail.com](mailto:Imamfauji04@gmail.com)

## Abstrak

Skripsi ini membahas tentang perancangan robot *line follower* dan pengaturan kecepatan motor penggerak robot menggunakan *Fuzzy Logic Controller* (FLC). *Line follower* yang dibuat diaplikasikan sebagai robot manajemen sampah dengan daya penggerak masing-masing roda berupa motor DC. Data keluaran sensor photodiode berupa logika 0 atau 1 dikonversi dalam bentuk sudut ( $0^\circ$ ) menggunakan metode *reasoning fuzzy* untuk selanjutnya digunakan sebagai masukan kontroler. Sistem kontrol yang digunakan pada alat ini adalah *fuzzy logic control* (FLC) dengan *set point* berupa simpangan antara badan robot dengan *line* sebesar ( $0^\circ$ ). Fungsi FLC pada sistem ini adalah sebagai pengambil keputusan menggunakan metode fuzzifikasi MAX-MIN. Sedangkan metode defuzzifikasi yang digunakan adalah *Center of Gravity* (COG). Hasil dari pengaplikasian FLC dapat memperkecil *error steady state* hingga 0% dengan *delay time* 1,96s hingga 2,56s.

**Kata kunci** : *line follower*, kontroler, *reasoning*, *fuzzy logic control*

## 1 Pendahuluan

Dewasa ini hampir semua orang mengenal robot. Robot dapat diklasifikasikan dalam berbagai jenis, antara lain: *mobile robot* yang dapat berpindah posisi dari satu titik ke titik yang lain, *robot manipulator* atau robot lengan dapat bekerja menyerupai lengan manusia, *robot humanoid* yang menyerupai manusia secara utuh, *flying robot* yang dapat terbang, robot berkaki, robot jaringan yang dikendalikan melalui internet dengan protocol TCP/IP, robot animalia, dan robot *cyborg*. [1]

*Mobile robot* merupakan jenis robot yang paling populer. *Mobile robot* dapat dibuat sebagai pengikut garis (*line follower*), pengikut dinding (*wall follower*), pengikut cahaya serta *Unmanned Aerial Vehicle* (UAV).

*Line follower* merupakan jenis robot beroda yang bergerak mengikuti lintasan berupa garis (*line*) dengan daya penggerak berupa motor. Robot *line follower* biasanya digunakan dalam ajang perlombaan robot. Bahkan dapat di aplikasikan dalam membantu pekerjaan manusia menangan tugas "search and rescue". [1] Contohnya Robot Manajemen Sampah yang

digunakan dalam pengambilan dan pemilahan sampah berdasarkan jenis sampahnya.

Kestabilan *line follower* dengan menerapkan metode osilasi Ziegler-Nichols untuk tuning parameter PID memiliki rata-rata *error* 5%. [4]

Namun kontroler PID tidak sesuai jika digunakan sebagai pengambil keputusan. Oleh karena itu pada laporan skripsi ini menggunakan *Fuzzy Logic Control* (FLC) berbasis mikrokontroler ATmega8 dan ATmega16. [2] [3]

FLC dinilai lebih efektif dibandingkan dengan kontroler PID dalam hal pengambilan keputusan, karena FLC masih dapat bertahan dengan adanya perubahan nilai dari parameter kontrol. Dengan penerapan FLC diharapkan dapat memperkecil nilai *error* dari penelitian sebelumnya, sehingga pergerakan robot lebih halus (*smooth*).

## 2 Tinjauan Pustaka

### 2.1. Robot *Line Follower*

Robot *line follower* atau robot pengikut garis adalah suatu jenis robot yang pergerakannya dengan mendeteksi garis sehingga robot tersebut dapat melakukan perpindahan posisi dari satu titik ke titik yang lain. Pada skripsi ini robot *line follower* yang di gunakan adalah robot beroda dengan sistem *differensial drive*. Sistem *diferensial* merupakan sistem penggerak menggunakan dua buah roda yang dikopel motor yang dapat digerakkan secara terpisah. [5]

Sensor yang digunakan adalah sensor photodiode. Robot tersebut dirancang untuk di aplikasikan sebagai robot manajemen sampah.

### 2.2 *Fuzzy Logic Control*

*Fuzzy* secara harfiah berarti samar, sedangkan kebalikannya dalam hal ini adalah *Crisp* yang secara harfiah berarti tegas [8]. Tahun 1965 L.A. Zadeh memodifikasi teori himpunan yang disebut himpunan kabur (*fuzzy Set*). Himpunan *fuzzy* di dasarkan pada gagasan untuk memperluas jangkauan fungsi karakteristik sehingga fungsi tersebut akan mencakup bilangan real pada interval [0,1]. Nilai keanggotaannya menunjukkan bahwa suatu nilai dalam semesta pembicaraan tidak hanya berada pada 0 atau 1, namun juga nilai yang terletak diantaranya. Komponen utama penyusun kontrol logika *fuzzy* adalah unit fuzzifikasi, *fuzzy inference*, dan unit defuzzifikasi.

fuzzifikasi merupakan proses untuk mengubah variabel non *fuzzy* (variabel numerik) menjadi variabel *fuzzy* (variabel linguistik). Nilai masukan-masukan yang masih dalam bentuk variabel

numerik yang telah dikuantisasi sebelum diolah oleh pengendali logika *fuzzy* harus diubah terlebih dahulu ke dalam variabel *fuzzy*. Proses fuzzifikasi diekspresikan sebagai berikut:

$$x = \text{fuzzifier}(x_0) \quad (1)$$

dengan:

- $x_0$  = nilai *crisp* variabel masukan
- $x$  = himpunan *fuzzy* variabel yang terdefinisi
- fuzzifier* = operator fuzzifikasi yang memetakan himpunan *crisp* ke himpunan *fuzzy*

Pedoman memilih fungsi keanggotaan untuk proses fuzzifikasi, menurut Jun Yan, menggunakan:

1. Himpunan *fuzzy* dengan distribusi simetris.
2. Gunakan himpunan *fuzzy* dengan jumlah ganjil, berkaitan erat dengan jumlah kaidah (*rules*).
3. Mengatur himpunan *fuzzy* agar saling menumpuk.
4. Menggunakan fungsi keanggotaan bentuk segitiga atau trapesium.

Metode inferensi merupakan proses untuk mendapatkan keluaran dari suatu kondisi masukan dengan mengikuti aturan-aturan yang telah ditetapkan.

Pada metode Max–Min aturan operasi minimum Mamdani digunakan untuk implikasi *fuzzy*. Persamaan aturan minimum adalah

$$\mu_{C_i} = \bigcup_1^n \alpha_i \wedge \mu_{C_i} \quad (2)$$

dengan  $\alpha_i = \mu_{A_i}(x_0) \wedge \mu_{B_i}(y_0)$

Sebagai contoh, terdapat dua basis kaidah aturan *fuzzy*, yaitu:

$R_1$  : Jika  $x$  adalah  $A_1$  dan  $y$  adalah  $B_1$  maka  $z$  adalah  $C_1$

$R_2$  : Jika  $x$  adalah  $A_2$  dan  $y$  adalah  $B_2$  maka  $z$  adalah  $C_2$

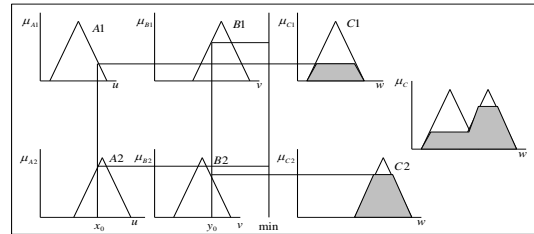
Pada metode penalaran MAX-MIN fungsi keanggotaan konsekuen dinyatakan dengan

$$\mu_{C_i}(W) = \mu_{C_i1} \vee \mu_{C_i2} = [\alpha_1 \wedge \mu_{C_i1}(w)] \vee [\alpha_2 \wedge \mu_{C_i2}(w)] \quad (3)$$

$$\text{Dimana } \alpha_1 = \mu_{A_1}(x_0) \wedge \mu_{B_1}(y_0) \quad (4)$$

$$\alpha_2 = \mu_{A_2}(x_0) \wedge \mu_{B_2}(y_0) \quad (5)$$

Lebih jelas metode ini dideskripsikan dalam Gambar 1



**Gambar 1** Inferensi *Fuzzy* dengan Metode MAX-MIN

(Yan, 1994)

Defuzzifikasi adalah proses untuk mendapatkan nilai numerik dari data *fuzzy* yang dihasilkan dari proses inferensi (Yan, 1994). Proses defuzzifikasi dinyatakan sebagai berikut:

$$y_0 = \text{defuzzifier}(y) \quad (6)$$

dengan:

- $y$  : aksi kontrol *fuzzy*
- $y_0$  : aksi kontrol *crisp*
- defuzzifier* : operator defuzzifikasi

Metode *Center Of Gravity* (COG) didefinisikan sebagai berikut:

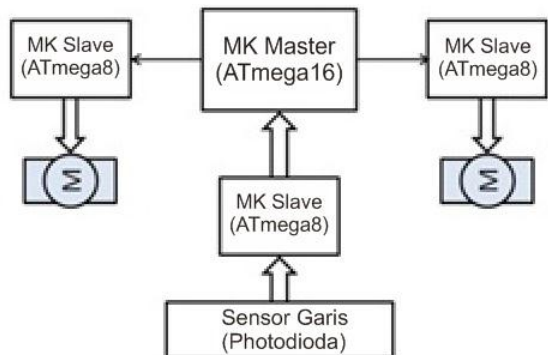
$$U = \frac{\sum_{i=1}^n w_i u_i}{\sum_{i=1}^n w_i} \quad (7)$$

dengan:

- $U$  = Keluaran
- $w_i$  = Bobot nilai benar  $w_i$
- $u_i$  = Nilai linguistik pada fungsi keanggotaan keluaran
- $n$  = Banyak derajat keanggotaan

### 3 Perancangan Alat

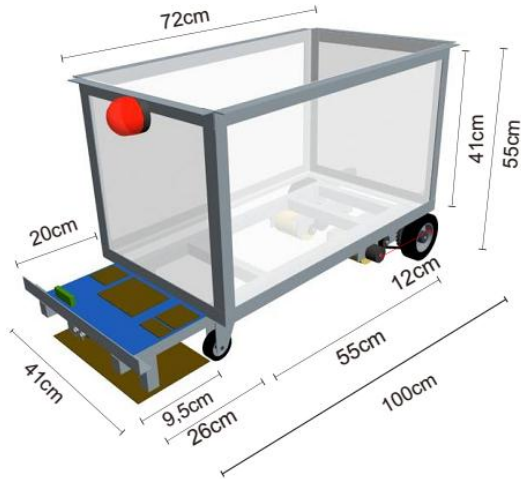
Pada perancangan alat diperlukan blok diagram sistem yang dapat menjelaskan sistem secara garis besar. Blok diagram sistem alat ditunjukkan dalam Gambar 2.



**Gambar 2** Blok Diagram Sistem

Mekanik robot *line follower* dirancang menyerupai miniatur truk pengangkut dengan

panjang 100cm, lebar 40cm dan tinggi 45cm. Robot memiliki box dengan ukuran 70cm x 40cm x 40cm, dua roda depan merupakan roda bebas dan dua roda belakang yang dikopel motor DC sebagai penggerak dengan sistem differensial (terpisah). [5] [6]

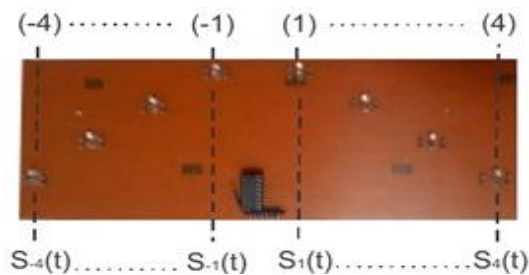


**Gambar 3** Desain Mekanik Robot Tampak Perspektif Depan Samping

Photodiode yang digunakan sebagai sensor berjumlah delapan buah dan jarak antara sensor yang satu dengan yang lain ditentukan sebesar 3cm.

Sistem ini menggunakan metode penalaran (*reasoning*). Untuk mendapatkan fungsi keanggotaan masukan posisi ( $P$ ) menggunakan fungsi keanggotaan segitiga himpunan *fuzzy*. [7]

Fungsi keanggotaan masukan terbagi menjadi lima label yaitu, *Left High* (LH), *Left Middle* (LM), *Zero* (Z), *Right Middle* (RM) dan *Right High* (RH). Identifikasi sensor dapat dilihat dalam Gambar 5. Keadaan logika sensor saat ( $t$ ) di asumsikan  $S_a(t)$ . Subscript  $a$  merepresentasikan posisi setiap sensor berdasarkan rancangan sensor dalam Gambar 4



**Gambar 4** Identifikasi Sensor

Sensor akan berlogika 1 jika mendeteksi warna hitam dan berlogika 0 saat mendeteksi warna putih.

$$S_a(t) = \begin{cases} 1 & \text{sensor ON} \\ 0 & \text{sensor OFF} \end{cases}$$

Sedangkan nilai dari masing-masing sensor yang mendeteksi warna hitam di asumsikan sebagai  $N_a(t)$ . Hubungan antara  $S_a(t)$  dan  $N_a(t)$  dapat dilihat pada persamaan 5

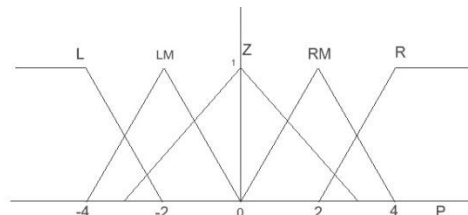
$$\begin{aligned} S_a(t) = 1 & \leftrightarrow N_a(t) = a \\ S_a(t) = 0 & \leftrightarrow N_a(t) = 0 \end{aligned} \quad (8)$$

Dari persamaan 5 dapat dihitung nilai  $P$  :

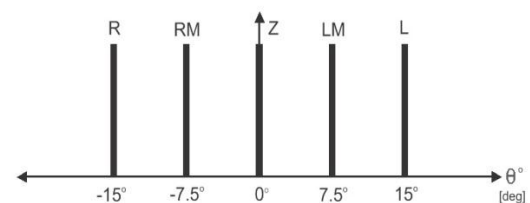
$$P = \frac{\sum_{a=-4}^4 N_a(t)}{\sum_{a=-4}^4 S_a(t)} \quad (9)$$

dimana  $a = \{-4, -3, -2, -1, 1, 2, 3, 4\}$

Rentang nilai  $P$  diambil dari batas  $a$ , yaitu dari -4 sampai 4. Sedangkan keluaran menggunakan *singleton* yang bukan fungsi keanggotaan tetapi nilai numerik yang nyata. Fungsi keanggotaan keluaran sensor merupakan representasi dari sudut simpangan robot yang dikeluarkan oleh MK *slave* sensor. Konstanta *singleton* ditentukan  $-15^\circ$ ,  $-7,5^\circ$ ,  $0^\circ$ ,  $7,5^\circ$  dan  $15^\circ$ . Fungsi keanggotaan masukan posisi ditunjukkan dalam Gambar 5 sedangkan konstanta singleton ditunjukkan dalam Gambar 6



**Gambar 5** Fungsi Keanggotaan Masukan P



**Gambar 6** Konstanta Singleton Sensor

Berdasarkan diagram blok sistem dalam Gambar 2, dibutuhkan dua buah FLC. Satu FLC sebagai kontroler untuk motor kiri dan satu FLC sebagai kontroler motor kanan. Perbedaan dari keduanya terletak pada penentuan fungsi keanggotaan keluaran.

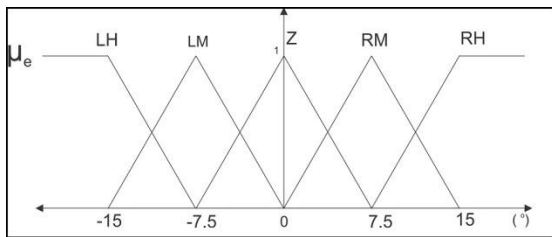
Variabel masukan yang digunakan dalam perancangan kontroler ada dua, yaitu *error* ( $e$ ) dan *delta error* ( $\Delta e$ ) posisi robot terhadap *line* ( $\theta^\circ$ ), sedangkan variabel keluaran kontroler berupa *Pulse Width Modulation* (PWM) motor.

Error dan *delta error* di representasikan dalam perumusan sebagai berikut :

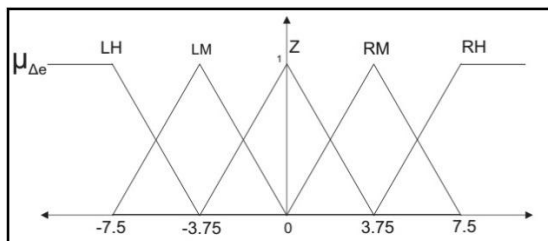
$$e(t) = \text{set poin} - \text{present value}(t) \quad (10)$$

$$\Delta e(t) = e(t) - e(t - 1) \quad (11)$$

Masing-masing variabel *error* dan *delta error* mempunyai fungsi keanggotaan yang terbagi lagi menjadi lima label yaitu, *Left High* (LH), *Left Middle* (LM), *Zero* (Z), *Right Middle* (RM) dan *Right High* (RH). Fungsi keanggotaan masukan *error* dan *delta error* ditunjukkan dalam Gambar 7 dan Gambar 8.

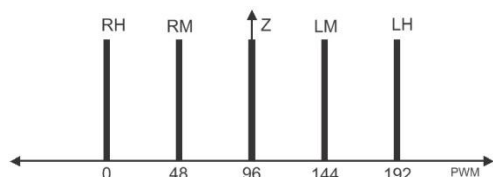


Gambar 7 Fungsi Keanggotaan Masukan *Error*.

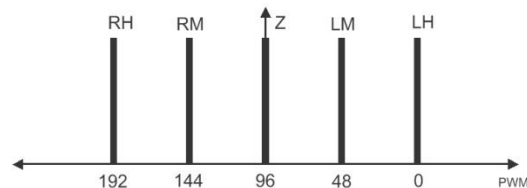


Gambar 8 Fungsi Keanggotaan *Delta Error*

Gambar 9 dan Gambar 10 menunjukkan fungsi keanggotaan keluaran motor DC kiri dan kanan yang merupakan representasi dari besarnya nilai PWM yang dikeluarkan oleh mikroprosesor berupa hasil perhitungan metode defuzzifikasi *Center Of Gravity (COG)* yang telah dibulatkan oleh integer di dalam pemrograman.



Gambar 9 Fungsi Keanggotaan Keluaran Motor Kiri



Gambar 10 Fungsi Keanggotaan Keluaran Motor Kanan

*Rule* (aturan) *fuzzy* digunakan sebagai penentu keluaran dari fuzzifikasi yang akan diolah dalam proses defuzzifikasi, dengan jumlah *rule* 25 macam. *Rule* tersebut didapat dari 5 *membership function error* dan 5 *membership function delta error*. (Tabel 1)

Tabel 1 Aturan *Fuzzy*

$\begin{matrix} e \\ \Delta e \end{matrix}$	LH	LM	Z	RM	RH
LH	RH	RH	RH	RM	Z
LM	RH	RH	RM	Z	LM
Z	RH	RM	Z	LM	LH
RM	RM	Z	LM	LH	LH
RH	Z	LM	LH	LH	LH

#### 4 Pengujian Alat

Bab ini berisi proses pengujian dan analisis terhadap alat sistem yang telah dirancang.

##### 4.1. Pengujian Sensor Photodioda

Peralatan yang digunakan :

- Sensor photodioda
- Mikrokontroler ATmega8 beserta rangkaian minimum sistemnya.
- LCD
- Laptop dengan perangkat lunak *AVR Studio* untuk memberikan perintah pengolahan data logika sensor dan *Proteus* untuk merekam hasil pengolahan mikrokontroler *ATmega8*.

##### 4.1.1 Langkah Pengujian

Peralatan dirangkai seperti pada Gambar 11



Gambar 11 Rangkaian Pengujian Sensor Photodioda

Catu daya rangkaian berasal dari port *USB* laptop. Kemudian memberikan perintah pengolahan data ke mikrokontroler *ATmega8* menggunakan

AVR studio. Sensor photodiode di arahkan pada objek berwarna putih dengan garis hitam dan data keluaran sensor direkam menggunakan *software proteus*.

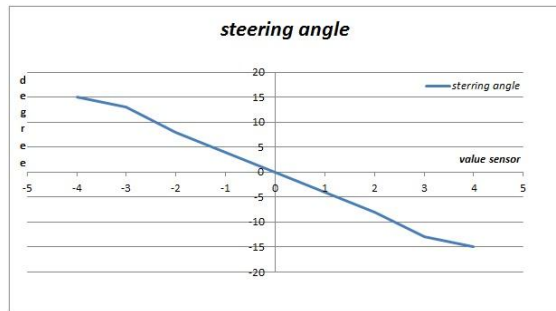
#### 4.1.2 Hasil Pengujian

Hasil pengujian berupa data keluaran *steering angle* berdasarkan nilai sensor dapat dilihat dalam tabel 2

**Tabel 2**

value sensor	-4	-3	-2	-1	0	1	2	3	4
steering angle ( $\theta^\circ$ )	15	13	8	4	0	-4	-8	-13	-15

Hasil pengujian sensor dalam grafik dapat dilihat dalam Gambar 12



**Gambar 12** Grafik Keluaran Sensor Berdasarkan Nilai Sensor

Gambar 12 menunjukkan bahwa *steering angle* berdasarkan nilai sensor photodiode terbatas antara range  $-15^\circ$  dan  $15^\circ$  dari *set point*  $0^\circ$ .

#### 4.2 Pengujian FLC

Pengujian ini bertujuan untuk mengetahui performa FLC yang diaplikasikan pada alat. Pengujian dilakukan dengan cara memberikan masukan FLC berupa sudut simpangan ( $\theta^\circ$ ) antara badan robot dengan *line* yang berbeda-beda berdasarkan *steering angle* sensor. Hasil dari pengujian FLC dapat dilihat pada Tabel 3.

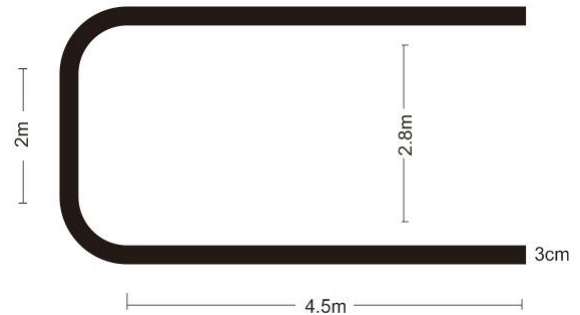
**Tabel 3** Hasil Pengujian FLC

Steering angle ( $\theta^\circ$ )	PWM Motor Kanan	PWM Motor Kiri
15	0	192
13	12	179
8	44	147
4	70	121
0	96	96
-4	121	70
-8	147	44
-13	179	12
-15	192	0

Tabel 3 menunjukkan nilai keluaran PWM motor dari hasil perhitungan FLC berdasarkan *steering angle* ( $\theta^\circ$ ) yang dihasilkan sensor photodiode.

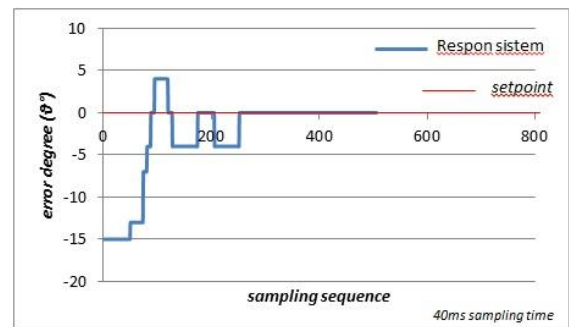
#### 4.3 Pengujian Sistem Keseluruhan

Pengujian pergerakan robot dalam mengikuti *line* dilakukan dengan menggunakan lintasan berbentuk huruf U seperti pada Gambar 13. Lebar garis adalah 3cm. Ukuran garis disesuaikan dengan konstruksi robot yang cukup besar.

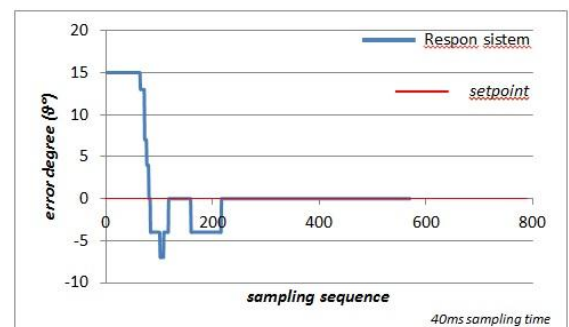


**Gambar 13** Line Pengujian

Untuk mendapatkan respon sistem, digunakan *software proteus* untuk merekam data. Data tersebut kemudian di dampilkan dalam bentuk grafik menggunakan *microsoft excel*. Gambar 14 dan Gambar 15 menunjukkan respon sistem terhadap simpangan sudut ( $\theta^\circ$ ).



**Gambar 14** Grafik Respon Sistem terhadap Error Posisi Negatif



**Gambar 15** Grafik Respon Sistem terhadap Error Posisi Positif

Gambar 14 menunjukkan respon sistem terhadap simpangan atau *error* sudut ( $\theta^\circ$ ) negatif dari *set point*. Pada kondisi tersebut, sistem

menunjukkan adanya dua kali osilasi dan *overshoot* sebesar 26,67% serta *settling time* sebesar 10,08s dengan *delay* sebesar 1,96s. Sedangkan dalam Gambar 15 respon sistem mengalami *overshoot* sebesar 46,67% dan mencapai *settling time* pada 8,68s dengan *delay* sebesar 2,56s.

## 5 Kesimpulan

Dari hasil skripsi ini dapat disimpulkan bahwa penggunaan FLC pada *line follower* dapat memperkecil *error steady state* sistem hingga 0%. Akan tetapi FLC memiliki respon yang lebih lambat jika dibandingkan dengan *PID controller* dikarenakan adanya waktu *delay* yang cukup besar pada sistem.

## 6 Saran

Pada sistem *differensial drive* perlu diperhatikan tingkat kepresisian perakitan mekanik *gearbox* dari aktuator robot. Untuk plan yang memiliki ukuran lebih besar dan membutuhkan respon yang cepat sebaiknya digunakan motor dengan torsi lebih besar.

## Daftar Pustaka

- [1] Adeck A. K., 2013. *Sisem Pengendalian Kecepatan Motor Pendorong Robot Hovercraft Line Follower Menggunakan Kontroler PID Berbasis Mikrokontroler ATmega8535*. Skripsi. Teknik Elektro Universitas Brawijaya. Malang: Indonesia
- [2] Atmel. 2010. *8-bit Microcontroller with 16K Bytes In-System Programmable Flash*. <http://www.atmel.com/images/2466s.pdf>
- [3] Atmel. 2013. *8-bit Microcontroller with 8K Bytes In-System Programmable Flash*. [http://www.atmel.com/images/atmel-2486-8-bit-avr-microcontroller-atmega8\\_1\\_summary.pdf](http://www.atmel.com/images/atmel-2486-8-bit-avr-microcontroller-atmega8_1_summary.pdf)
- [4] Bagus, I.S., 2013. *Perancangan Robot Auto Line Follower yang Menerapkan Metode Osilasi Ziegler-Nichols Untuk Tuning Parameter PID pada Kontes Robot Indonesia*. Skripsi. Teknik Elektro Universitas Brawijaya. Malang: Indonesia
- [5] Cook, Gerald. 2011. *Mobile robots : navigation, control and remote sensing*. New Jersey. John Wiley & Sons, Inc.,
- [6] Fitzgerald, A.E. 2003. *Electric Machinery(6<sup>th</sup> edition)*. New York: McGraw-Hill Companies, Inc.
- [7] Takimoto, H. & Hoshino, Y. 2009. *Development of fuzzy controller for the steering angle and the motor power*

control application to the line trace car. *FUZZ-IEEE 2009. IEEE International. Korea*

- [8] Yan, J. 1994. *Using Fuzzy Logic*. London: Prentice Hall