

IMPLEMENTASI FUZZY LOGIC CONTROL PADA SISTEM KONTROL KECEPATAN MOTOR DC UNTUK MOBIL ANAK BERBASIS ANDROID

IMPLEMENTATION OF FUZZY LOGIC CONTROL ON DC CAR MOTOR SPEED CONTROL SYSTEM FOR ANDROID BASE CHILD CAR

Lukas Christian, Porman Pangaribuan², Agung Surya Wibowo³

^{1,2,3}Prodi S1 Teknik Elektro, Fakultas Teknik Elektro, Universitas Telkom

¹obhlukas@yahoo.co.id, ²porman@telkomuniversity.ac.id, ³agungsw@telkomuniversity.ac.id

Abstrak

Mobil anak adalah mobil mainan yang dimodifikasi sehingga dapat dinaiki dan dikemudikan anak-anak dibawah 5 tahun. Pada awalnya mobil mainan anak diciptakan meniru konsep sepeda roda tiga, 1 roda di depan dan 2 roda di belakang. Penjualan dan pembaharuan pada mobil mainan anak perlahan mulai menjamur di toko-toko mainan anak, mulai dari semakin baiknya detail body mobil mainan anak, penambahan 1 roda didepan hingga dilengkapi dengan motor DC.

Pada tugas akhir kali ini, penulis menambahkan metode *fuzzy logic control* pada mobil anak yang dikendalikan dengan menggunakan Android. Sensor *line tracking* yang dipasang pada motor DC digunakan untuk monitoring data dari kecepatan putar. *Fuzzy logic control* digunakan untuk meminimalisir *error* dan mempersingkat waktu respon yang dibutuhkan motor DC agar kecepatan putar konstan saat mobil anak dinaiki dan dikemudikan anak-anak yang memiliki berat yang berbeda-beda sesuai dengan kecepatan yang diinginkan terutama pada saat jalan lurus.

Pada tahap pengujian tanpa beban menggunakan sampling 0.5 detik didapatkan hasil, *error* $\pm 11.7\%$, *settling time* ± 24 detik. Hasil yang didapat dari pengujian mobil anak saat diberi beban menggunakan sampling 0.5 detik, *error* $\pm 11.74\%$, *settling time* ± 18 detik. Pada tahap pengujian tanpa beban menggunakan sampling 1.5 detik didapatkan hasil, *error* $\pm 7.88\%$, *settling time* ± 21 detik. Hasil pengujian mobil anak diberi beban menggunakan sampling 1.5 detik, *error* $\pm 3.68\%$, *settling time* ± 22.5 detik. Pada hasil pengujian ini dapat disimpulkan *fuzzy logic control* yang dirancang pada mobil anak mampu memperbaiki kecepatan putar sehingga mendekati atau sama dengan *setpoint*.

Kata Kunci : *fuzzy logic control, line tracking, Android, motor DC*

Abstract

Car toy is a toy car that can ride and steered children under 5 years old. At first the car toys were created to imitate the concept tricycle, one wheel in front and two wheels in the rear. Sales and updates on the car slowly started mushrooming toys in the shops of toys, ranging from improvements in the detail of the car body toys, the addition of one front wheel to be equipped with DC motors.

In this final project, the author adds fuzzy logic control method on a toy car controlled by using android. Line tracking sensors mounted on the dc motor is used to monitor the data of rotational speed. Fuzzy logic control is used to minimize errors and shorten the response time required DC motor so that the rotating speed constant current child car ride and nttrolled children, weight varies according to the speed like setpoint, especially when the road is straight.

At this stage of trying without load using 0.5 second sampling results obtained, error $\pm 11.7\%$, settling time ± 24 seconds. The results of the testing of child car when weighted using sampling 0.5 seconds, error $\pm 11.74\%$, settling time ± 18 seconds. At this stage of pengujian without load using 1.5 second sampling results obtained, error $\pm 7.88\%$, ± 21 seconds settling time. Results pengujian child car is loaded using 1.5 second sampling, the error is $\pm 3.68\%$, ± 22.5 seconds settling time. On the results of this test can be concluded fuzzy logic control designed in a child's car is able to fix the rotational speed so close to or equal to the setpoint.

Keywords: *fuzzy logic control, Rotary Encoder, Android, DC motor*

1. Pendahuluan

Mobil mainan anak pada umumnya mobil mainan yang dapat dinaiki dan dikemudikan anak-anak dibawah 4 tahun. Bermula dari perancangan sederhana mobil mainan anak pada abad ke - 20 yang meniru konsep sepeda roda tiga, mobil mainan anak menggunakan roda berjari-jari dan ban mati, dimana pada waktu itu 1 roda dipasang dibagian depan dan 2 roda dipasang dibagian belakang, chasis dan body-nya pun masih menggunakan kayu dan logam. Kemudian mobil anak dimodifikasi dengan menambahkan 1 roda ekstra di bagian depan, sehingga mobil mainan anak memiliki 4 roda.

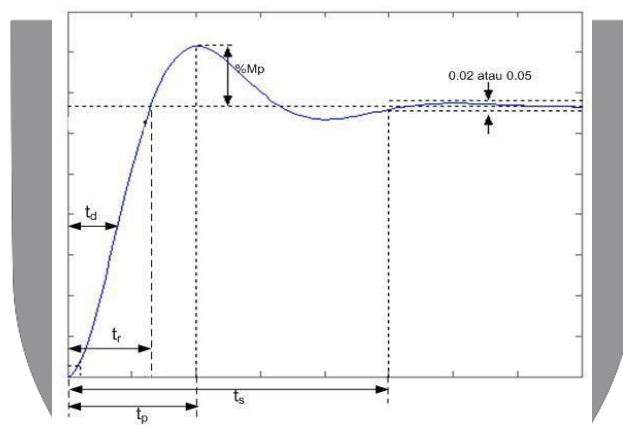
Penjualan mobil mainan anak yang semakin laris di pasaran membuat industri mobil mainan berlomba-lomba mengembangkan mobil mainan. Perancangan mobil anak berkembang dari tampilan mobil mainan anak yang dirancang lebih detail hingga mobil mainan anak yang dilengkapi dengan motor DC. Dibalik perkembangan itu ada beberapa masalah yang berkaitan erat dalam hal keamanan dan kenyamanan anak. Tidak semua anak pada umur 1 sampai 4 tahun mengerti dengan baik cara mengendarai mobil mainan anak ini, oleh karena itu peran penting dari orang tua tetap diperlukan untuk menjaga dan memberi perhatian bagi perkembangan anak.

Mobil mainan anak akan dikomunikasikan langsung dengan android via *bluetooth* sehingga orang tua dapat mengendalikan mobil mainan anak secara langsung dari android. Mobil mainan anak yang dirancang penulis akan dilengkapi dengan motor DC dan sensor *line tracking*. Sensor *line tracking* akan memonitoring data kecepatan putar motor DC, yang digunakan sebagai umpan balik sehingga sistem akan memperbaiki kecepatan putar bila kecepatan putar belum sesuai.

2. Dasar Teori dan Perancangan

Karakteristik Respon Sistem Orde 2

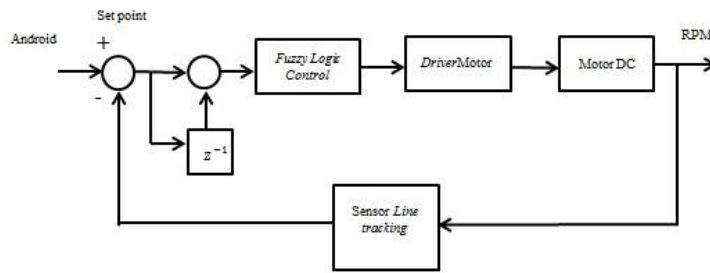
Karakteristik respon sistem orde 2 dapat dilihat dari bentuk umum kurva respon orde 2 untuk masukan sinyal unit step



Gambar Kurva Respon Orde 2

- Waktu naik (*rise time*), t_r
Waktu naik adalah waktu yang di ukur mulai dari respon = 0 sampai dengan sistem memotong sumbu *steady state* yang pertama
- *Overshoot* maksimum, M_p
Overshoot adalah nilai yang menyatakan perbandingan antara nilai maksimum respon yang melampaui nilai *steady state* sistem
- Waktu tunak (*settling time*), t_s
Waktu tunak adalah ukuran waktu yang menyatakan respon telah masuk 5% atau 2% atau 0.5% dari keadaan *steady state*

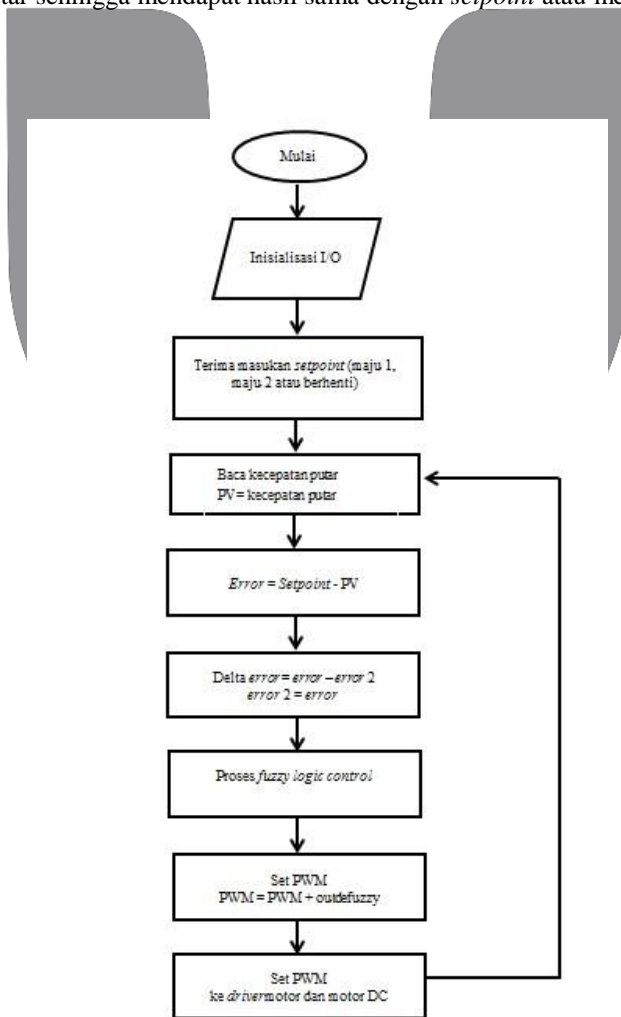
Prinsip kerja



Gambar Prinsip Kerja

Gambaran umum sistem dapat dilihat dari gambar 3.1. *Setpoint* dan *fuzzy logic control* terdapat di dalam mikrokontroler. Nilai *setpoint* merupakan nilai kecepatan putar yang diinginkan pada sistem mobil mainan anak. Pada gambar diagram blok di atas perancangan mobil mainan anak akan dihubungkan dengan Android dan sensor *line tracking*. Android akan memberi perintah arah kepada mikrokontroler lewat *bluetooth* sehingga Android bisa mengendalikan sistem pada mikrokontroler secara langsung. Pada saat tombol kecepatan ditekan data dari Android akan terhubung dengan Arduino Uno dan menjalankan perintah sesuai dengan *setpoint* yang sudah diatur terlebih dahulu. Perintah tersebut kemudian menjadi masukan bagi *fuzzy logic control* yang akan menggerakkan motor DC. *Sensor line tracking* kemudian akan membaca kecepatan putar dari motor DC. Hasil baca dari sensor *line tracking* tersebut akan dikirim kembali ke mikrokontroler, hasil baca sensor akan menjadi masukan di dalam *fuzzy logic control*. Kini *fuzzy logic control* memiliki 2 masukan di dalamnya, masukan dari Android yang terhubung dengan *setpoint* awal dan masukan dari hasil pembacaan kecepatan putar dari motor DC. *Fuzzy logic control* akan membandingkan kecepatan putar hasil pembacaan sensor dengan nilai kecepatan putar *setpoint*. Apabila hasil baca sensor *line tracking* belum sesuai dengan *setpoint* maka sistem akan terus memperbaiki kecepatan putar sehingga mendapat hasil sama dengan *setpoint* atau mendapat hasil yang mendekati dengan *setpoint*.

Flowchart



3 Pengujian dan Analisis

Tujuan pengujian dan analisis sistem adalah untuk memperoleh kecepatan konstan, dimana nilai RPM sesuai atau mendekati nilai set point. Pilihan Maju 1 pada android merupakan interface dari set point 1 dan pilihan maju 2 pada android merupakan interface set point 2. Pada pengujian kali ini, diharapkan mobil anak dapat melaju dengan kecepatan yang sama atau mendekati nilai set point yang diberikan. Adapun penambahan beban pada mobil anak saat berhenti dan kembali berjalan untuk menguji kelayakan mobil anak saat dinaiki anak dan melihat respon pada sistem. Penambahan beban pada mobil anak mulai dari 8 kg hingga 15 kg. Set point 1 diberi nilai sebesar 100 RPM dan set point 2 diberi nilai sebesar 150 RPM.

Cara pengujian dan hasil pengujian

Ada beberapa cara yang dilakukan sebelum pengujian dan setelah pengujian untuk dapat melihat RPM dari mobil anak. Sebelum pengujian pastikan laptop dan arduino terhubung dengan kabel peripheral, buka code mobil anak pada software arduino, pilih tool lalu pilih serial monitor. Pada serial monitor akan terlihat gambar seperti ini

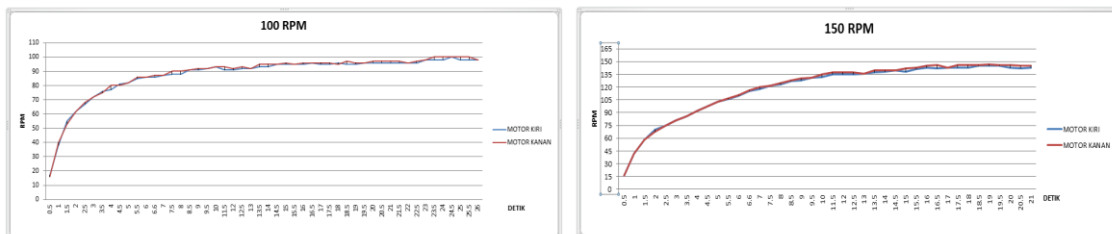


Tujuan pengujian dan analisis sistem adalah untuk memperoleh kecepatan konstan, dimana nilai RPM sesuai atau mendekati nilai *setpoint*. Pilihan Maju 1 pada android merupakan *interface* dari *setpoint* 1 dan pilihan maju 2 pada android merupakan *interface setpoint* 2. Pada pengujian kali ini, diharapkan mobil anak dapat melaju dengan kecepatan yang sama atau mendekati nilai *setpoint* yang diberikan. Adapun penambahan beban pada mobil anak saat berhenti dan kembali berjalan untuk menguji kelayakan mobil anak saat dinaiki anak dan melihat respon pada sistem. Penambahan beban pada mobil anak mulai dari 8 kg hingga 15 kg. *Setpoint* 1 diberi nilai sebesar 100 RPM dan *setpoint* 2 diberi nilai sebesar 150 RPM.

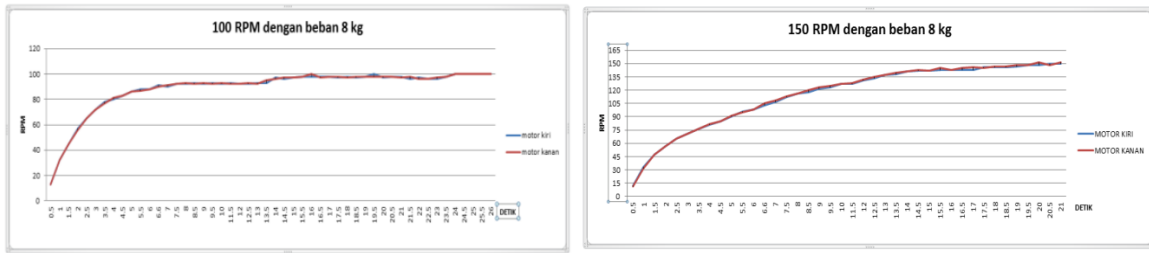
Buka microsoft word, pilih insert kemudian pilih chart. Sesudah itu secara otomatis excel akan terbuka dan terhubung dengan word. Salin semua nilai RPM pada excel sesudah itu grafik akan muncul secara otomatis dalam word seperti gambar pada hasil pengujian.

Hasil Pengujian dengan Menggunakan Sampling 0.5 Detik

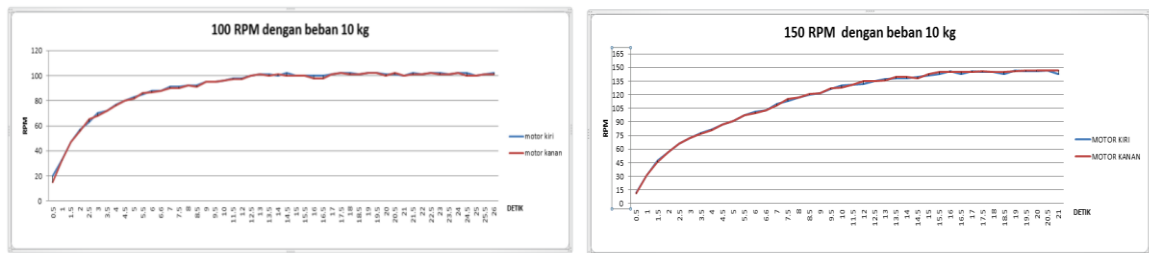
Dalam tahap pengujian ini, hasil pengujian 100 RPM dan 150 RPM didapatkan melalui nilai baca RPM pada serial monitor dalam Arduino Uno, Maksud menggunakan sampling 0.5 adalah sistem kontrol kecepatan putar pada Arduino Uno akan membaca nilai RPM dan memperbaiki kecepatan putar tiap 0.5 detik, sehingga nilai baca RPM pada Arduino Uno bisa mendekati atau sama dengan nilai *setpoint*



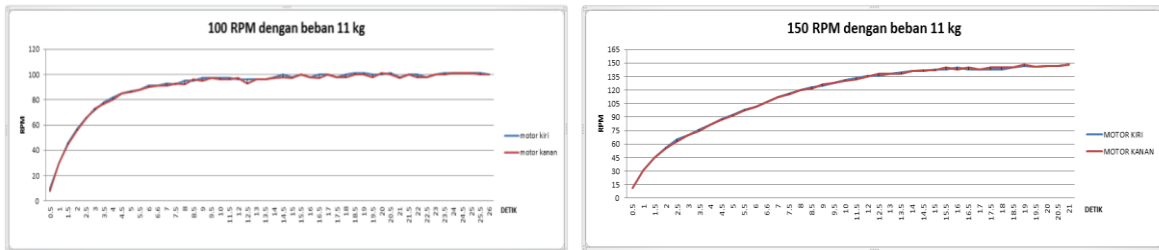
Gambar Hasil Pengujian 100 RPM dan 150 RPM Tanpa Beban Menggunakan Sampling 0.5 Detik



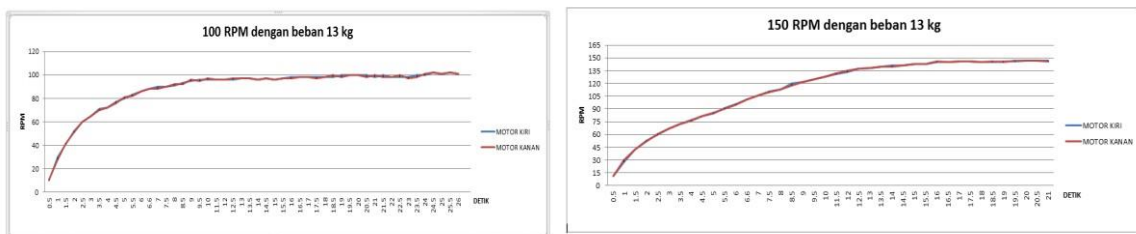
Gambar Hasil Pengujian 100 RPM dan 150 RPM dengan Beban 8 kg Menggunakan Sampling 0.5 Detik



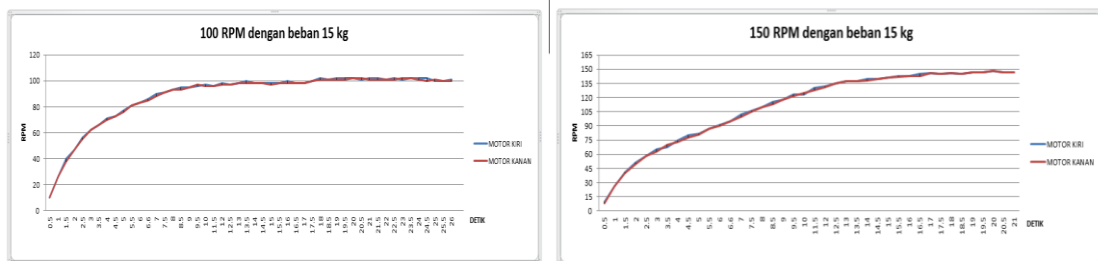
Gambar Hasil Pengujian 100 RPM dan 150 RPM dengan Beban 10 kg Menggunakan Sampling 0.5 Detik



Gambar Hasil Pengujian 100 RPM dan 150 RPM dengan Beban 11 kg Menggunakan Sampling 0.5 Detik



Gambar Hasil Pengujian 100 RPM dan 150 RPM dengan Beban 13 kg Menggunakan Sampling 0.5 Detik



Gambar Hasil Pengujian 100 RPM dan 150 RPM dengan Beban 15 kg Menggunakan Sampling 0.5 Detik

Analisis Hasil Pengujian Menggunakan Sampling 0.5 Detik

Dari hasil pengujian 100 RPM dan 150 RPM dengan menggunakan sampling 0.5 detik didapat hasil seperti tabel dibawah ini

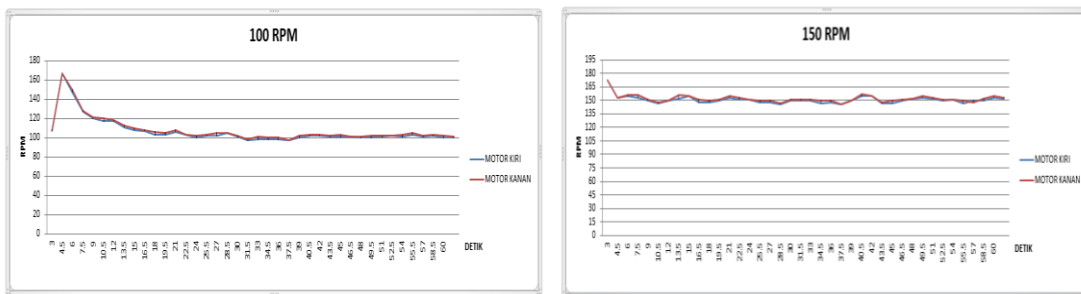
Hasil Pengujian 100 RPM	Overshoot	Settling time		Error rata-rata		Rise time	
		Motor kiri	Motor kanan	Motor kiri	Motor kanan	Motor kiri	Motor kanan
Tanpa beban	-	24	23	12.44%	11.74%	23.5 s	22 s
Beban 8 Kg	-	19	15.5	11.34%	11.3%	18.5 s	15 s
Beban 9 kg	-	21	21	10.74%	11.28%	20.5 s	20.5 s
Beban 10 Kg	-	12.5	12.5	9.16%	9.68%	12 s	12 s
Beban 11 Kg	-	14	15	9.54%	10.38%	13.5 s	14.5 s
Beban 12 Kg	-	19	19	11.24%	11.46%	18.5 s	18.5 s
Beban 13 Kg	-	18	17.5	11.52%	11.7%	17.5 s	17 s
Beban 14 Kg	-	13.5	16.5	10.66%	11.08%	13 s	16 s
Beban 15 Kg	-	14	18	11.26%	11.72%	13.5 s	17.5 s

Tabel 4.1 Hasil Analisis Pengujian 100 RPM Menggunakan Sampling 0.5 Detik

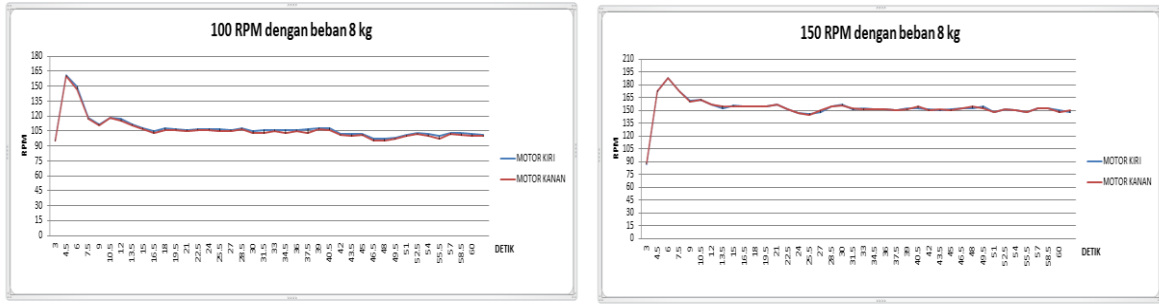
Hasil Pengujian 150 RPM	Overshoot	Settling time		Error rata-rata		Rise time	
		Motor kiri	Motor kanan	Motor kiri	Motor kanan	Motor kiri	Motor kanan
Tanpa beban	-	> 20 s	> 20 s	20.8%	19.92%	> 20 s	> 20 s
Beban 8 Kg	-	20 s	20 s	23.62%	23.18%	19.5 s	19 s
Beban 9 kg	-	> 20 s	> 20 s	25.77%	25.67%	> 20 s	> 20 s
Beban 10 Kg	-	12 s	12 s	23.43%	23.27%	11.5 s	11.5 s
Beban 11 Kg	-	> 20 s	> 20 s	22.9%	22.97%	> 20 s	> 20 s
Beban 12 Kg	-	> 20 s	> 20 s	24.78%	24.67%	> 20 s	> 20 s
Beban 13 Kg	-	> 20 s	> 20 s	24.28%	24.28%	> 20 s	> 20 s
Beban 14 Kg	-	> 20 s	> 20 s	24.35%	24.23%	> 20 s	> 20 s
Beban 15 Kg	-	> 20 s	> 20 s	25.55%	25.92%	> 20 s	> 20 s

Tabel 4.2 Hasil Analisis Pengujian 150 RPM Menggunakan Sampling 0.5 Detik

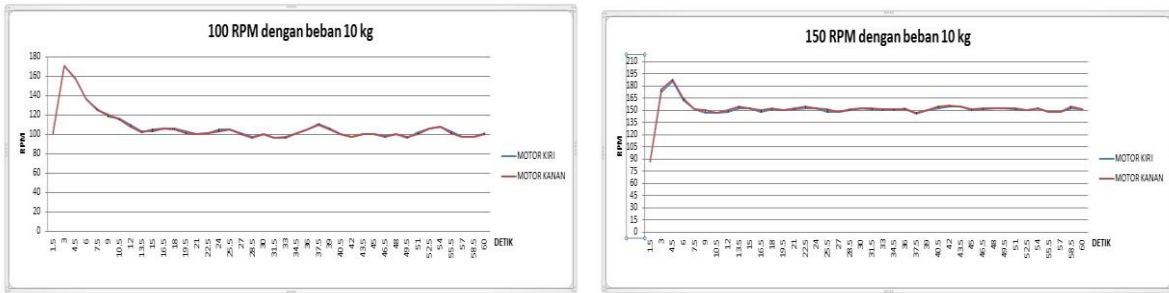
Hasil Pengujian dengan Menggunakan Sampling 1.5 Detik



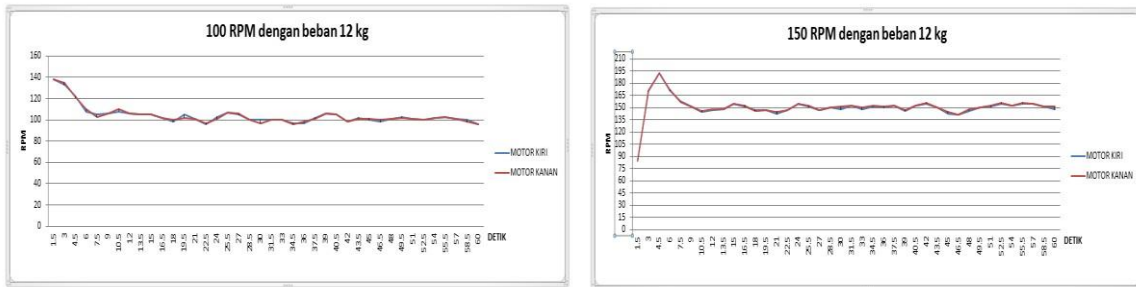
Gambar Hasil Pengujian 100 RPM dan 150 RPM Tanpa Beban Menggunakan Sampling 1.5 Detik



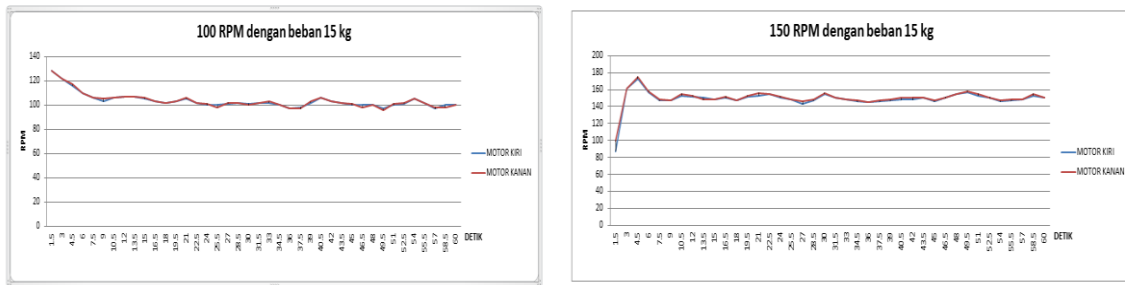
Gambar Hasil Pengujian 100 RPM dan 150 RPM dengan Beban 8 kg Menggunakan Sampling 1.5 Detik



Gambar Hasil Pengujian 100 RPM dan 150 RPM dengan Beban 8 kg Menggunakan Sampling 1.5 Detik



Gambar Hasil Pengujian 100 RPM dan 150 RPM dengan Beban 12 kg Menggunakan Sampling 1.5 Detik



Gambar Hasil Pengujian 100 RPM dan 150 RPM dengan Beban 12 kg Menggunakan Sampling 1.5 Detik

4.5 Analisis Hasil Pengujian dengan Menggunakan Sampling 1.5 Detik

Dari hasil pengujian 100 RPM dan 150 RPM dengan menggunakan sampling 1.5 detik didapat hasil seperti tabel 4.3 dan tabel 4.4 dibawah ini

Hasil Pengujian 100 RPM	Overshoot		Settling time		Error rata-rata		Rise time	
			Motor kiri	Motor kanan	Motor kiri	Motor kanan	Motor kiri	Motor kanan
Tanpa beban	67	67	21 s	21 s	6.45%	7.88%	1.5 s	1.5 s
Beban 8 Kg	61	60	40.5	40.5	7.95%	6.04%	1.5 s	1.5 s
Beban 9 kg	62	62	22.5	22.5	6.57%	6.07%	1.625 s	1.625 s
Beban 10 Kg	71	71	40.5	40.5	7.15%	6.78%	1.5 s	1.5 s
Beban 11 Kg	47	47	55.5	55.5	4.65%	4.78%	2 s	2 s
Beban 12 Kg	38	38	40.5	40.5	4.125%	4.05%	1.5 s	1.5 s
Beban 13 Kg	37	37	46.5	46.5	3.18%	3.2%	1.5 s	1.5 s
Beban 14 Kg	45	46	51	51	4%	4.15%	2 s	2 s
Beban 15 Kg	28	28	15	22.5	11.26%	11.72%	1.5 s	1.5 s

Tabel 4.3 Hasil Analisis Pengujian 150 RPM Menggunakan Sampling 1.5 Detik

Hasil Pengujian 100 RPM	Overshoot		Settling time		Error rata-rata		Rise time	
			Motor kiri	Motor kanan	Motor kiri	Motor kanan	Motor kiri	Motor kanan
Tanpa beban	23	23	7.5 s	7.5 s	0.6%	1.43%	1.5 s	1.5 s
Beban 8 Kg	38	38	30	30	1.78%	1.85%	1.5 s	1.5 s
Beban 9 kg	27	28	51	51	1.22%	0.84%	1.625 s	1.625 s
Beban 10 Kg	36	38	7.5	7.5	0.52%	1.13%	1.5 s	1.5 s
Beban 11 Kg	38	40	7.5	7.5	0.58%	1%	2 s	2 s
Beban 12 Kg	43	43	48	48	0.2%	0.65%	1.5 s	1.5 s
Beban 13 Kg	37	38	45	45	0.7%	0.37%	1.5 s	1.5 s
Beban 14 Kg	23	25	33	33	0.1%	0.3%	2 s	2 s
Beban 15 Kg	23	25	10.5	10.5	0.65%	0.13%	1.5 s	1.5 s

Tabel 4.4 Hasil Analisis Pengujian 150 RPM Menggunakan Sampling 1.5 Detik

KESIMPULAN

Dalam penelitian yang diambil beberapa tahap, diambil kesimpulan sebagai berikut:

1. Metode *fuzzy logic control* yang diimplementasikan pada sistem dapat bekerja dengan baik walaupun diberi gangguan yang berbeda-beda.
2. Nilai RPM kanan dan RPM kiri akan selalu berdekatan / sama saat berjalan lurus.
3. Sistem yang menggunakan sampling 1.5 detik lebih cepat untuk mencapai/ mendekati nilai *setpoint* di banding sistem yang menggunakan sampling 0.5.

DAFTAR PUSTAKA

- [1] Ikhsan Ismail. 2015. *Implementasi Logika Fuzzy Dan Kalman Filter untuk Kendali Lengan Robot Menggunakan Gestur Tangan Manusia*. Bandung . Tugas Akhir : Jurusan Teknik Elektro Universitas Telkom.
- [2] Depoinovasi. (2015). Motor DC Geared Kursi Roda Elektrik [online]. Available : <http://www.depoinovasi.com/produk-728-motor-dc-geared-kursi-roda-elektrik.html>. Diakses tanggal : 28 Februari 2016.
- [3] Wolber, David, et al., *App Inventor : Create Your Own Android Apps*, 1st ed. Canada : O'Reilly, 2011.
- [4] DR. Eng. Agus Naba. 2009. *Tutorial Cepat & Mudah Fuzzy Logic dengan MATLAB*. Malang : Agus Naba.