

PENGONTROLAN KECEPATAN PADA PERMAINAN JUNGKAT-JUNGKIT AUTOMATIS DENGAN MENGGUNAKAN METODE LOGIKA FUZZY

AUTOMATIC SEESAW SPEED DESIGN CONTROLLER USING FUZZY LOGIC METHOD

Indah Dwiyana¹, Ir. Porman Pangaribuan, M.T.², Agung Surya Wibowo, ST., M.T.³

^{1,2,3}Prodi S1 Teknik Elektro, Fakultas Teknik Elektro, Universitas Telkom Jalan Telekomunikasi, Dayeuh Kolot
Bandung 40257 Indonesia

¹indahdwiyana@student.telkomuniversity.ac.id, ²porman@telkomuniversity.ac.id,

³agungsw@telkomuniversity.ac.id

Abstrak

Jungkat-jungkit adalah sebuah permainan yang terbuat dari papan lurus yang panjang dan memiliki titik tumpuan pada posisi tengah, jika salah satu sisinya naik maka sisi yang lain akan turun. Permainan ini biasanya terdapat di taman bermain. Cara bermain jungkat-jungkit adalah masing-masing anak duduk di setiap ujung, kemudian mereka bergiliran melonjakkan tubuh dari tanah. Hal ini membuat lonjakan pada kedua sisi akan berbeda. Tujuan tugas akhir ini adalah mengontrol pergerakan frekuensi ayunan jungkat-jungkit walaupun berat beban yang ada pada sisi jungkat-jungkit berbeda, frekuensi ayunan akan tetap konstan. Di sisi lain, jungkat-jungkit otomatis juga dirancang bergerak secara otomatis apabila hanya satu orang anak saja yang ingin bermain.

Pada tugas akhir ini, dibuat sebuah *prototype* jungkat-jungkit otomatis. Sistem ini menggunakan mikrokontroler, motor DC dan sensor *rotary encoder*. Memonitring frekuensi ayunan akan ditampilkan pada sebuah *Liquid Crystal Display*(LCD). Frekuensi dan sensor *rotary encoder* sebagai umpan balik sistem. Algoritma yang digunakan adalah metode logika fuzzy. Metode ini dilakukan agar mencapai hasil yang diinginkan.

Dalam tugas akhir ini, *prototype* jungkat-jungkit yang dirancang dapat mengatasi permasalahan berat beban yang berbeda pengguna dengan memiliki frekuensi ayunan yang konstan. Pada pengujian respon tanpa beban, respon sistem menunjukkan apabila Nilai input *membership function error* semakin diperbesar, maka waktu *settling time* yang diperoleh semakin cepat. *Prototype* jungkat-jungkit dapat dimainkan dengan perbedaan berat beban pada kedua sisi. Respon sistem juga tetap mengikuti *set point* walau terdapat berat beban.

Kata Kunci : *Speed control, Fuzzy Logic, rotary encoder, sistem kontrol*

Abstract

Seesaw is a game made of straight board and have a fulcrum at the middle position if the one of side up the other side will go down. This game is usually found in the playground. Playing the seesaw is the kids sitting on the top of a seesaw, and then they bump the board alternately. This makes two sides can bump alternately. The purpose of a final project is the seesaw can control the speed when the two children have a different weight and then the speed of seesaw can stable. And the other side, a design of seesaw can bump automatic if only one of the child wants to play the seesaw.

In the final project, the automatic seesaw will be made a prototype. In this system use microcontroller, DC motor, and sensor rotary encoder. DC motor used as speed drive and rotary encoder sensor as a feedback system. The algorithm using the fuzzy logic method. In this method used to archive the result.

In the final project, the prototype can stabilize the speed of seesaw and not fixated on the number of users. The prototype has a two balance that is a 50 and 40 degrees, and the speed divided into three speed. there is a slow, normal, and fast.

Keywords : *Speed control, Fuzzy Logic, rotary encoder,controller*

1. Pendahuluan

Jungkat-jungkit adalah sebuah permainan di mana papan panjang dan sempit berporos di tengah, sehingga jika salah satu ujungnya bergerak naik maka ujung yang lain bergerak turun. Jungkat-jungkit sering ditemukan di taman bermain. Pada jungkat-jungkit terdapat bagian penting yaitu tumpuan dan berat beban. Jungkat-jungkit akan seimbang jika memiliki berat beban yang sama pada kedua sisi. Kestabilan pada permainan ini sangat diperlukan mengingat berat beban pada anak berbeda-beda. Dengan perkembangan teknologi dibutuhkan permainan yang dapat menjaga kestabilan secara otomatis.

2. Dasar Teori dan Perancangan

2.1 Konsep Cara Kerja Solusi

Berdasarkan rumusan masalah, sistem dapat bekerja secara manual dan otomatis. Pengguna akan menekan tombol start dan dapat mengatur kecepatan yang diinginkan dengan menekan push button. Sensor kecepatan dan aktuator digunakan untuk mencapai kecepatan yang diinginkan dan dikendalikan pada sebuah pengendali utama. Pada tugas akhir ini, kecepatan jungkat-jungkit yang diinginkan akan tampil pada sebuah LCD(Liquid Crystal Display). Sistem jungkat-jungkit diharapkan mempunyai frekuensi ayunan yang konstan walau berat pengguna di kedua sisi jungkat-jungkit berbeda, termasuk juga pada kasus jungkat-jungkit yang dimainkan oleh satu orang anak. Berdasarkan konsep cara kerja diatas, masukan sistem adalah frekuensi ayunan. Frekuensi ayunan tersebut akan dikonversi ke dalam kecepatan putar motor DC. Nilai kecepatan putar motor DC tersebut akan dikendalikan oleh sistem kendali. Keluaran sistem berupa kecepatan awal ditambahkan dengan kecepatan yang dihasilkan oleh sistem kendali

2.2 Sistem Kontrol

Sistem kendali adalah suatu susunan yang terhubung atau terkait sedemikian rupa sehingga dapat memerintah, mengarahkan, atau mengatur diri sendiri atau sistem lain. Sistem kendali secara umum dapat diklasifikasikan menjadi sistem kontrol terbuka(*open loop*) dan sistem kontrol tertutup(*closed loop*). Perbedaan sistem kontrol terbuka dan tertutup adalah sistem *open loop* tidak memiliki blok umpan balik sedangkan pada *closed loop* memiliki umpan balik.

2.3 Kendali Logika Fuzzy

Logika fuzzy adalah cara memetakan ruang input dan ruang output. Logika fuzzy diperkenalkan oleh Prof. Lotfi A. Zadeh. Menurut Prof. Lotfi pemanfaatan logika fuzzy akan mempercepat dan mempermudah hasil dalam setiap kasus. Himpunan logika fuzzy mempunyai fungsi keanggotaan segitiga dan fungsi keanggotaan trapezium. Terdapat tiga proses dalam metode fuzzy logic ini, yaitu *fuzzification*, *fuzzy inference*, dan *defuzzification*.

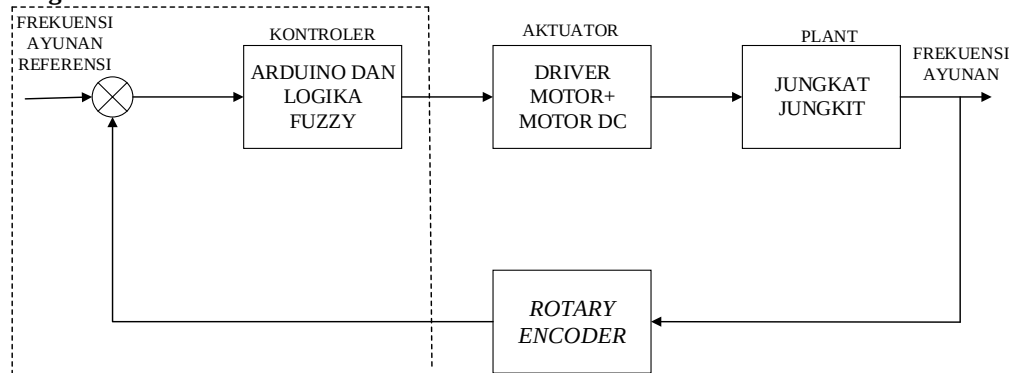
2.4. Sensor Rotary Encoder

Sensor *rotary encoder* adalah sensor yang memiliki fungsi untuk memonitor gerakan dan posisi. Sensor *rotary encoder* ini akan membentuk serial pulsa yang akan menghasilkan gerakan, posisi, dan arah. *Rotary encoder* tersusun dari suatu piringan yang tipis yang memiliki lubang-lubang pada lingkaran piringan. Poros sensor *rotary encoder* akan mengukur kecepatan rotasi per detik atau permenit dengan jumlah putaran suatu poros dalam satu menit yang disebut RPS(*Revolution Per Second*) atau RPM(*Revolution Per Minutes*).

2.5 Motor DC (*Dirrent Current*)

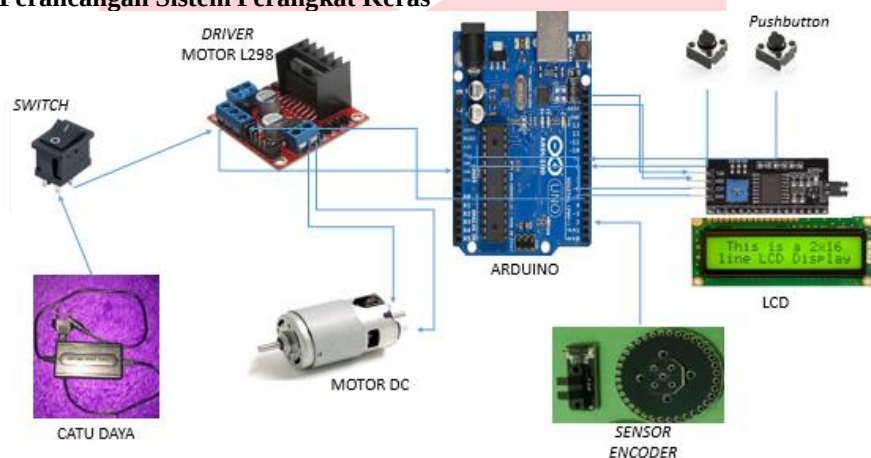
Motor DC merupakan perangkat elektromagnetik yang mengubah energi listrik menjadi energi mekanik. Motor DC membentuk suatu putaran pada motor dibutuhkan kawat yang dibentuk menjadi suatu loop yang kemudian ditempatkan pada suatu medan magnet. Sedangkan untuk laju motor dapat diatur dengan memvariasikan lilitan medan dimana terjadi proses perubahan energi listrik menjadi torsi yang dipengaruhi pula oleh besarnya beban. Untuk mengatur kecepatan motor DC dengan cara melebarkan pulsa dalam satu periode. Perbandingan pada waktu naik dan turun nya dapat menentukan kecepatan pada motor. Perputaran motor DC membutuhkan perubahan arah, pada magnet motor dalam posisi normal dapat mengubah arah motor dari perubahan polarisasi. [2] Pengendalian kecepatan motor DC berupa PWM(*Pulse Widht Modulation*). PWM berfungsi Kecepatan motor DC dikontrol oleh driver motor untuk tegangan tinggi diperlukan untuk kecepatan motor DC. Untuk menghasilkan nilai PWM tersedia sebuah mikrokontroler. Mikrokontroler dapat terhubung dengan motor DC melalui driver motor.

2.6 Perancangan Sistem



Gambar 1 Diagram Blok Sistem Kendali

2.6.1 Perancangan Sistem Perangkat Keras

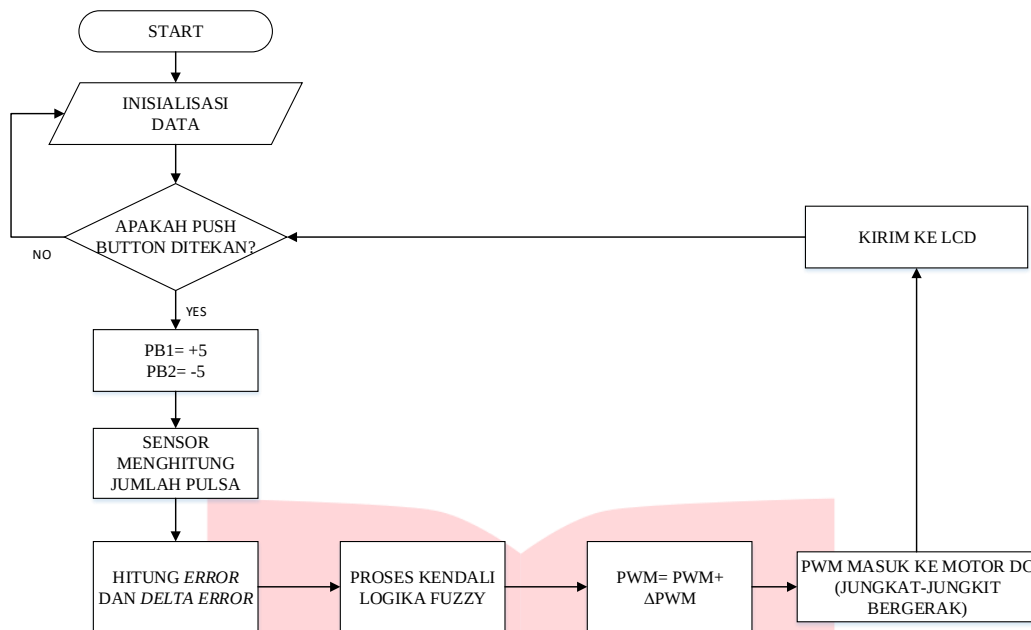


Gambar 2 Perancangan Sistem Perangkat Keras

Untuk pengimplementasian dan merealisasikan sistem, dirancanglah sebuah perangkat keras. Pada perancangan ini komponen-komponen yang digunakan adalah tombol switch sebagai tombol ON/OFF. *Pushbutton* untuk mengatur kecepatan. Arduino Uno sebagai pengendali utama. Sensor rotary encoder untuk memonitor gerakan rotasi motor DC. Driver motor DC sebagai penggerak dan pengatur kecepatan motor DC. Motor DC sebagai pengatur kecepatan pada jungkat-jungkit, *Liquid Crystal Display*(LCD) untuk menampilkan data dan Catu daya sebesar 12volt.

2.6.2 Perancangan Sistem Perangkat Lunak

Diagram alir adalah urutan proses pengurutan kerja suatu sistem. Diagram alir sistem jungkat-jungkit otomatis dapat dilihat pada gambar III.5. Setelah start, sistem akan menginisialisasi data. Data yang di inisialisai adalah *input* dan *output* . Selanjutnya adalah memasukan nilai *set point* yaitu ω referensi dengan menekan salah satu push button untuk mengatur kecepatan yang di inginkan. Nilai ω referensi masuk, sensor *rotary encoder* akan menghitung jumlah pulsa per detik berdasarkan kecepatan putar motor DC. Ketika sistem berjalan, akan terdapat nilai *error* dan *delta error*. Ketika nilai *error* dan *delta error* sudah didapat maka proses kendali logika fuzzy akan bekerja dan akan memproses keluaran sistem yang berupa nilai Δ PWM. Nilai PWM yang didapat dari proses kendali logika fuzzy akan dikirim ke motor DC. Kecepatan putar motor DC akan menyebabkan terjadinya frekuensi ayunan pada jungkat-jungkit. Jika nilai kecepatan pada jungkat-jungkit sudah memenuhi yang di inginkan (*set point*) maka sistem berjalan dengan baik. Dan akan menampilkan setpoint dan frekuensi ayunan yang diperoleh ke LCD.



Gambar 2. Diagram Alir Sistem

3. Pengujian dan Analisis

3.1 Nilai Output Antara MATLAB Dan Arduino

Tabel 1. Tabel Hasil Pengujian Nilai Output MATLAB dan Arduino

No	Nilai error	Nilai delta error	Hasil Output Matlab	Hasil Output Arduino
1	-62	-39.7	-35	-35
2	-26.2	-19.2	-25.5	-25.6
3	-50.1	42.1	-28.8	-28.76
4	-60	-19	-35	-35
5	0	0	0	0
6	38	41	35	35
7	55	20	35	35
8	58	10	35	35
9	45.3	-42.3	22.7	22.6
10	30	-25	9.3	9.1

Berdasarkan pengujian yang telah dilakukan, nilai *output*(PWM) yang didapat pada pengujian satu sampai sepuluh hampir memperoleh nilai output yang sama antara Arduino dan Matlab . Pada saat nilai error =(-26.2) dan delta error=(-19.2) nilai output yang diperoleh adalah 25.5 pada Arduino dan 25.6 pada Matlab. Saat diberikan nilai error = 58 dan delta error=10, nilai output yang diperoleh 22.7 pada Arduino dan 22.6 pada Matlab. Saat nilai *error* bernilai (-50.1) dan delta *error* bernilai 42.1 *Output* yang diperoleh Nilai *output* pada matlab bernilai (-28,8) dan (-28,7) pada Arduino. Nilai yang diperoleh memiliki selisih sebesar 0,1. Sedangkan saat nilai *error*= 30 dan delta *error*=(-25), nilai output yang diperoleh 9.3 pada Matlab dan 9.1 pada Arduino. Selisih yang diperoleh adalah 0,2 Karena selisih yang tidak terlalu jauh, maka di anggap mencapai target. Berdasarkan pengujian ini diketahui bahwa *fuzzy logic controller* sudah berjalan dengan baik.

3.2 Respon Sistem Tanpa Beban

3.2.1 Perubahan Nilai *Input Membership Function Error*

Dalam pengujian ini, dilakukan perubahan nilai *input membership function error* dengan 3 kondisi yang berbeda dan mendapatkan hasil sebagai berikut :

Tabel 2. Hasil Analisis Perubahan *Input Membership Function Error*

μ_F (error)	Settling Time
μ_F (error) 1 [-55 0 55]	22 detik
μ_F (error) 2 [-45 0 45]	33 detik
μ_F (error) 3 [-60 0 60]	21 detik

Berdasarkan tabel pengujian perubahan *membership function error*, diperoleh bahwa *membership delta error* 1 mempunyai settling time 22 detik, *membership function error* 2 (nilai diperkecil) memperoleh settling time 33 detik, dan *membership function* 3 (nilai diperbesar) memperoleh settling time 21 detik. Jadi, apabila nilai *membership function error* diperbesar akan diperoleh settling time yang lebih cepat.

3.2.2 Perubahan Nilai *Input Membership Function Delta Error*

Dalam pengujian ini, dilakukan perubahan nilai *input membership function delta error* dengan 3 kondisi yang berbeda dan mendapatkan hasil sebagai berikut :

Tabel 3. Hasil Analisis Perubahan *Input Membership Function Delta Error*

$\mu_{(de)}$	Settling Time
$\mu_F(de)$ 1 [-40 0 40]	18 detik
$\mu_F(de)$ 2 [-25 0 25]	32 detik
$\mu_F(de)$ [-15 0 15]	35 detik

Berdasarkan hasil pengujian perubahan *membership delta error* diperoleh , *Membership function delta error* 1 mempunyai waktu menuju ke *setpoint* yaitu 7 detik dengan settling time 81detik, *membership function delta error* 2 diperoleh waktu 12 detik untuk mencapai set point dengan settling time 33 detik dan *membership function delta error* 3 diperoleh waktu 13 detik dan settling time 31 detik.

3.2.3 Perubahan Nilai *Output*

Dalam pengujian ini, dilakukan perubahan nilai *output* dengan 3 kondisi yang berbeda dan mendapatkan hasil sebagai berikut :

Tabel 4. Hasil Analisis Perubahan *Output*

Nilai <i>Output</i>	<i>Settling time</i>
Output 1 [-15 0 15]	3 detik
Output 2 [-20 0 20]	20 detik
Output 3 [-30 0 30]	12 detik

Dari hasil pengujian diatas, ketika diberikan nilai *output* 1 *settling time* yang diperoleh adalah 3. Saat diberikan nilai *output* 2, *settling time* yang diperoleh adalah 20 detik. Sedangkan pada nilai *output* 3 *settling time* yang diperoleh adalah 12 detik. Berdasarkan hasil yang diperoleh, nilai *output* 1 memperoleh *settling time* yang lebih cepat dibandingkan dengan *output* 2 dan *output* 3 yaitu 3 detik.

3.2.4 Perubahan Nilai *Set Point*

Dalam pengujian ini, dilakukan perubahan nilai *set point* dengan 3 kondisi yang berbeda dan mendapatkan hasil sebagai berikut :

Tabel 5. Hasil Analisis Perubahan *Set Point*

Nilai <i>Set Point</i>	<i>Settling time</i>
<i>Set Point</i> 1 [40]	5 detik
<i>Set Point</i> 2 [70]	4 detik

Berdasarkan hasil pengujian perubahan *set point* seperti pada Gambar IV.4 dan IV.5 , ketika nilai *setpoint* 50 *settling time* yang diperoleh adalah 5 detik dan ketika *setpoint* bernilai 70 *settling time* yang diperoleh adalah 10 detik. Dari hasil yang diperoleh, dapat disimpulkan bahwa sistem sudah mengikuti *set point* yang di berikan. Jadi, sistem yang digunakan sudah baik.

3.3 Respon Sistem Terhadap Saat ada Berat Beban

Pengujian ini bertujuan untuk mengetahui respon sistem saat *Setpoint* yang diberikan adalah frekuensi terendah ayunan yaitu 10 dengan *sampling time* 300ms(*Millisecond*) ketika terdapat berat beban di kedua sisi jungkat-jungkit. Pada pengujian ini menggunakan *prototype* jungkat-jungkit yang akan diberikan berat beban yang berbeda pada kedua sisi. Berat beban yang digunakan sebesar 100gram, 200gram, dan 300gram. Untuk pembacaan respon sistem pada pengujian ini digunakan PLX-DAQ. Tabel pengujian ini dapat dilihat pada tabel di bawah ini :

Tabel 5. Hasil Respon Sistem Saat Ada Berat Beban

Pengujian ke-	Berat beban	<i>Settling Time</i>
1	100gr vs 0gr	37 detik
2	100gr vs 200gr	62 detik
3	100gr vs 300gr	84 detik
4	200gr vs 100gr	62 detik
5	200gr vs 300gr	63 detik

6	300gr vs 100gr	83 detik
7	300gr vs 200gr	66 detik
8	0gr vs 200gr	79 detik

Berdasarkan hasil pengujian yang dilakukan, pengujian 1(100gram-0gram) unggul dengan *settling time* selama 37 detik. Saat pengujian 2, berat beban disatu sisi diperbesar menjadi 200gram dan memperoleh *settling time* selama 62 detik. Pengujian 3, berat disatu sisi diperbesar lagi menjadi 300gram dengan *settling time* selama 84 detik. Pada pengujian 4, berat beban di sisi satu sebesar 200gram dan disisi kedua 100 gram memperoleh *settling time* selama 62 detik. Pada pengujian 5, disalah satu sisi diperbesar menjadi 300 gram dan memperoleh *settling time* selama 63 detik.

Pada pengujian 6, berat beban di sisi 1 jungkat-jungkit sebesar 300gram dan di sisi kedua sebesar 100gram memperoleh *settling time* selama 83 detik. Pada pengujian 7, di sisi kedua jungkat-jungkit diperbesar berat bebannya menjadi 200 gram dan memperoleh *settling time* selama 66 detik. Pada pengujian 8, disisi satu jungkat-jungkit memiliki berat 0gram dan di sisi kedua 200gram memperoleh *settling time* selama 79 detik. Dari hasil percobaan diatas, dapat diketahui bahwa walaupun berat beban pada kedua sisi berbeda, sistem mengikuti *setpoint* yang diinginkan.

4. Kesimpulan

Berdasarkan hasil pengujian yang telah dilakukan , maka dapat disimpulkan sebagai berikut:

1. Kecepatan motor dan frekuensi ayunan berbanding lurus dengan nilai PWM yang diberikan. Ayunan terjadi ketika PWM bernilai lebih dari 60.
2. Perbandingan nilai *output* antara Arduino dan simulasi pada MATLAB hampir memiliki nilai yang sama, terdapat perbedaan selisih sebesar 0,1 dan 0,2.
3. Pada pengujian respon tanpa beban, respon sistem menunjukkan apabila Nilai input *membership function error* semakin diperbesar, maka waktu *settling time* yang diperoleh semakin cepat. Jika nilai *membership function delta error* mendekati nilai *membership function error settling time* yang diperoleh akan semakin cepat.
4. *Prototype* jungkat-jungkit dapat dimainkan dengan perbedaan berat beban pada kedua sisi. Respon sistem juga tetap mengikuti *set point* walau terdapat berat beban.
5. Sistem kendali logika fuzzy sudah berjalan dengan baik pada *prototype* jungkat-jungkit otomatis.

Daftar Pustaka :

- [1] Dionisius, Ramaditya Putra Fatruan. "Rotary Encoder," . Available : <https://www.scribd.com/doc/95731167/Rotary-Encoder>. [Diakses 04 November 2016, 12:26 WIB]
- [2] Huang, Han-Way, *PIC Microcontroller: An Introduction to Software and Hardware Interfacing*. United States of America. ISBN 1-4018-3967-3
- [3] Aryandhi, Yudha Dwi, dan Talakua Mozart Wilson. "PENERAPAN INFERENSI FUZZY UNTUK PENGENDALI SUHU RUANGAN SECARA OTOMATIS PADA AIR CONDITIONER (AC)", Prosiding FMIPA Universitas Pattimura 2013, ISBN 978-602-97522-0-5
- [4] Persada, Wira. "Aturan Fuzzy,". Available: <http://arihputra.blogspot.co.id /2010/11/aturan-fuzzy.html> . [Diakses 16 Oktober 2016, 20:12 WIB]
- [5] Smith, Jack R, *Programming The PIC Microcontroller with MBasic*. United States of America. ISBN 0-7506-7946-8
- [6] Barret, Steven F, *Arduino Microcontroller Processing for Everyone!. Second Edition*. Southbourn Melbourne University. ISBN 9781608458608
- [7] *Penyusunan Formulir dan Flowchart*, Edisi Perama. Yogyakarta: Andi OFFSET ; Semarang: Lembaga Pendidikan Komputer Wahana, 1995. ISBN 979-555-295-3
- [8] Gani, Fadlan Nuran. "Pulse Widht Modulation(PWM),' . Available: <http://robotic-electric.blogspot.co.id/2012/11/pulse-width-modulation-pwm.html> . [Diakses 30 November 2016, 15:11 WIB]

- [9] Salamena, Vicky. "SIMULASI KARAKTERISTIK ARUS DAN KECEPATAN MOTOR DC TERHADAP MASUKAN PENYEARAH GELOMBANG PENUH DI SIMULINK-MATLAB, '. Available: [file:///C:/Users/USER/Downloads/teknologi 2012 9 1 8 salamena%20\(1\).pd](file:///C:/Users/USER/Downloads/teknologi%202012%209%201%208%20salamena%20(1).pd) . [Diakses 30 November 17:05 WIB]
- [10] Rismawan, Agung. 2015. *Konsep Sistem Kendali, Sistem Kendali Terbuka&Tertutup Dan Contoh Aplikasinya*. Diakses September 20, 2017. <https://serbatelekomunikasi.wordpress.com/2015/02/12/8/>
- [11] Sugiono, Djoko. 07 February 2015. Model Matematika Motor DC . Diakses Oktober 01 2017. <http://www.vedcmalang.com/ppptkboemlg/index.php/menuutama/listrik-electro/1322-motor-dc>

