

Modelling and Simulation of DC Motor Speed Control Using Fuzzy-PID Algorithm

Anrico Gideon Alfano¹, Hari Maghfiroh², Irwan Iftadi³, Chico Hermanu B.A.⁴, Feri Adriyanto⁵

Department of Electrical Engineering, Faculty of Engineering
Sebelas Maret University
Surakarta, Indonesia

[1anricogideon@gmail.com](mailto:anricogideon@gmail.com), [2hari.maghfiroh@staff.uns.ac.id](mailto:hari.maghfiroh@staff.uns.ac.id), [3iftadi@gmail.com](mailto:iftadi@gmail.com), [4chicohermanu@gmail.com](mailto:chicohermanu@gmail.com),
[5feri.adriyanto@staff.uns.ac.id](mailto:feri.adriyanto@staff.uns.ac.id)

Abstract— This paper discusses the simulation of the Fuzzy-PID hybrid algorithm as a method for controlling the speed of a DC motor compared to the usual PID method. Comparisons were also made to the membership functions used in the fuzzy logic fuzzification process. Membership functions that are used are triangular, trapezoidal, and gaussian shaped function with each having 3, 5, and 7 as the number of membership to be compared. The performance of this control is compared by looking at the results of the step unit responses and the ability in tracking signals. The simulation results show that the Fuzzy-PID algorithm is superior to the ordinary PID algorithm but slower on computation time. The results of the comparison in the fuzzification process shows that the more number used in membership function, the faster the response reaches the set point, while the difference in shape has little effect on the response. The best results are achieved by fuzzy triangle shaped function with 7 membership number.

Keywords—DC Motor, PID, Fuzzy Logic, Speed Control

I. INTRODUCTION

Electric motors are by far the most ubiquitous of the actuators.[1] There are 2 kinds of electric motor: AC and DC, however, both have differences. DC motors are often more popular than AC motors because of its superiority and ease in controlling speed[2]. In addition, DC motors are also simpler than AC in conversion and in general DC motors are cheaper for the performance it offers[3], therefore this study uses DC motors as plants to be controlled.

Speed control is a very important thing to do if a DC motor is applied to a system that often has changing conditions. One of the most popular control algorithms used is the PID algorithm. However, this PID algorithm has a weakness, namely a decrease in performance when there is a changes in the system conditions [4]. This is because PID cannot tune itself when the system is running. Therefore, it is important to be able to keep the DC motor at the desired speed while the system is running [5].

To overcome this problem, scientist began to develop an intelligent control system. One of them is FLC (Fuzzy Logic Controller). This research will combine the PID algorithm with the Fuzzy Logic algorithm. The performance of this combined algorithm (Fuzzy-PID) will be compared with the performance of ordinary PID algorithms. This research will also study the effect of the various membership function used in the fuzzification process using triangular, trapezoidal, and gaussian shaped functions with a membership number of 3, 5, and 7 as its variations to be compared. The difference in performance will be seen and compared based on the simulation results

of the step unit responses and its ability in signal tracking which will be observed in MATLAB.

II. DC MOTOR MODELLING

According to Maghfiroh[6], DC motors have three methods for controlling speed that are often used: field resistance control, armature voltage control, and armature resistant control. The method used in this study is the second one, the armature voltage control as it's the most practical and the most observable one.

A. DC Motor Modelling Module

This module is used to take the data of the DC motor used in the experiment, which was its input (v) and output (rpm) with respect to time to determine the characteristics of the motor that was used as the plant. Changes in voltage input are varied with a potentiometer connected to the Arduino analog pin. Potentiometer will vary the speed by utilizing the PWM (Pulse Width Modulation) function found on the L298N H-Bridge which is connected to the PWM pin that is already available on Arduino. L298N is used as a driver to control the DC motor that will be used. Arduino UNO was used in this experiment. Wiring diagrams for the module's circuit can be seen in Figure 1.

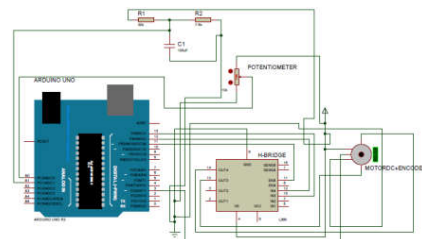


Figure 1 Wiring Diagram

Voltage reading was done using a voltage divider circuit with a ratio of 5:1 using 30k Ω and 7.5k Ω resistors which can read voltages up to 25V. Speed readings are carried out using a rotary encoder that is already connected to the DC motor. The portrait of the module used can be seen in Figure 2.

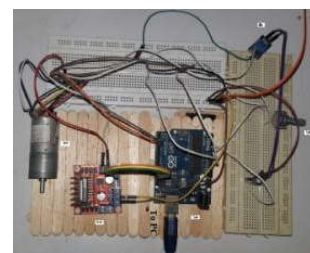


Figure 2 Portrait of the Module

B. System Identification

After the data is retrieved using the module, the next step is to identify the system to find its mathematical model. This model will determine the behavior of the motor which was based on the data taken. The system identification process is done using the MATLAB System Identification feature. The mathematical model of the motor obtained by System Identification is shown in equation (1).

$$\frac{138.5}{s^2 + 1.475s + 12.81} \quad (1)$$

III. CONTROLLER DESIGN

A. PID Configuration

The design of the PID's block diagram was made using the Simulink feature contained in MATLAB. The block diagram used for the PID in this study is shown in Figure 3.

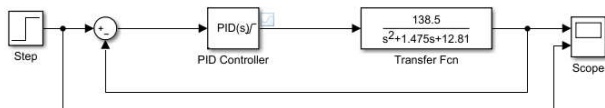


Figure 3 PID Block Diagram

The PID configuration used in this study is the ideal configuration as shown in equation (2). The K_p , K_i , and K_d parameters are determined through self-tuned.

$$y(t) = K_p[e(t) + K_i * \int e(t)dt + (K_d * \frac{de(t)}{dt})] \quad (2)$$

B. Fuzzy Logic Controller

Fuzzy Logic Controller (FLC) is a fuzzy logic based control that provides a way to change linguistic control strategies based on expert knowledge into (generally) an automatic control strategies[7][8]. According to Lee[9], the use of FLC can be seen as a step to model human decision making in the framework of the concept of fuzzy logic with the approximate approach.

The FLC method used in this study is the Mamdani method. There are 2 inputs used, namely, error (e) and voltage (v). The output of this FLC is the K_p parameter of PID. Membership functions that are compared are triangular, trapezoidal, and gaussian shaped functions with a membership number of 3, 5 and 7 as shown in figures 4, 5, and 6. These figures show fuzzy membership output

3 membership has N, ZE, and P as its member. 5 membership has NB, NS, ZE, PS, and PB, while 7 membership has NB, NM, NS, ZE, PS, PM, and PB. The rule base that was design for each membership is shown on table 1, 2, and 3.

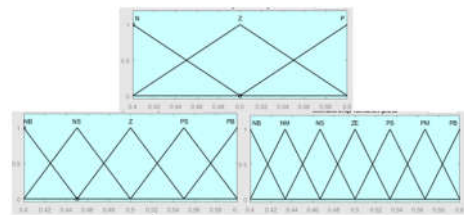


Figure 4 Triangular Function

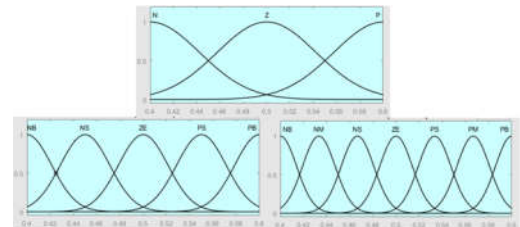


Figure 5 Gaussian Function

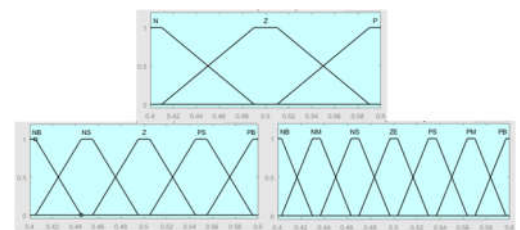


Figure 6 Trapezoid Function

Table 1 Rule Matrix for 3 Membership

v	^E	N	ZE	P
S		N	ZE	P
M		N	ZE	P
L		N	ZE	N

Table 2 Rule Matrix for 5 Membership

v	^E	NB	NS	ZE	PS	PB
VS		NB	NB	ZE	PS	PB
S		NS	NS	ZE	PS	PS
M		NB	ZE	ZE	ZE	PS
L		NS	NS	ZE	NS	NS
VL		NB	NS	ZE	NS	NB

Table 3 Rule Matrix for 7 Membership

v	^E	NB	NM	NS	ZE	PS	PM	PB
VVS		NB	NM	NM	ZE	PM	PM	PB
VS		NB	NM	NM	ZE	PS	PM	PB
S		NM	NS	NS	ZE	PS	PS	PM
M		NB	NM	ZE	ZE	ZE	PS	PS
L		NS	NS	ZE	ZE	ZE	NS	NS
VL		NB	NM	NS	ZE	NS	NM	NM
VVL		NB	NB	NM	ZE	NS	NM	NB

C. Fuzzy-PID Configuration

This concept applies the Double Tuning PID concept where the K_i and K_d parameters are determined using offline tuning while the K_p value is set using Fuzzy Logic as a system control and can change as the system goes (online). The block diagram for the Fuzzy-PID configuration is shown in Figure 7.

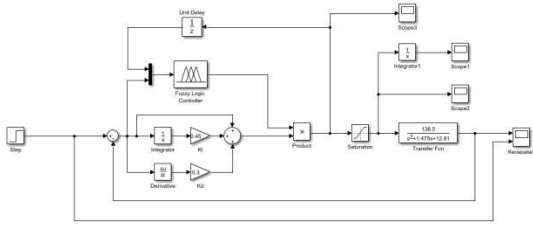


Figure 7 Fuzzy-PID Block Diagram

IV. SIMULATION RESULT

A. PID Performance

The self-tuned PID results in the unit response as shown in Figure 8. The PID is able to reach the setpoint at 8.5 seconds with a little overshoot and stabilizes at 9.4 seconds. The results of the PID responses are shown in table 4. The quantitative results shown are determined based on MATLAB calculations that calculate settling time with a tolerance of 2% from setpoint.

Table 4 PID Response Result

Settling Time(s)	Overshoot	Undershoot	Peak	Peak Time(s)
2.2655	0.0017	-	100.0017	4.8

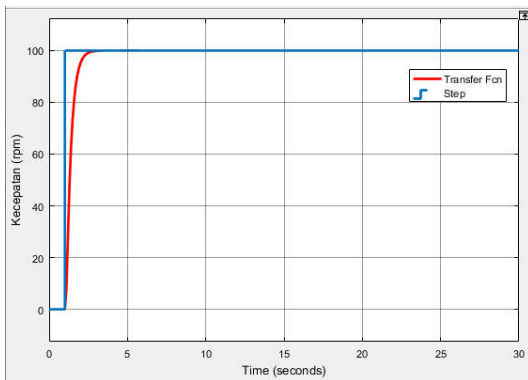


Figure 8 PID Response Result

B. Fuzzy-PID Performance (3 Membership Comparison)

Figure 9 shows the comparison results of 3 membership number to the 3 different shapes. At first glance, looks like there is no difference at all, but Figure 10, which is a zoomed in figure 9 will show the differences.

The simulation is run with the following parameters: $t = 10$, step time = 0.1 and final value = 100 rpm. Until $t = 10$, the response has not yet reached the final value of 100 rpm. The closest is 99.9 rpm. Therefore, the response time is extended to 30 seconds. After the time extension, the responses appear to be able to reach a final value of 100 rpm with different times for each membership function. The test results can be seen in table 5. The parameter terms that will be used in all the tables starting from this point have the following meanings:

- F = Membership Function (shape)
- TL = Triangular
- TP = Trapezoid
- GS = Gaussian
- P = Peak
- PT = Peak Time
- SPT = Time needed to reach setpoint 100
- O = Overshoot
- U = Undershoot
- ST = Settling Time (2% tolerance)

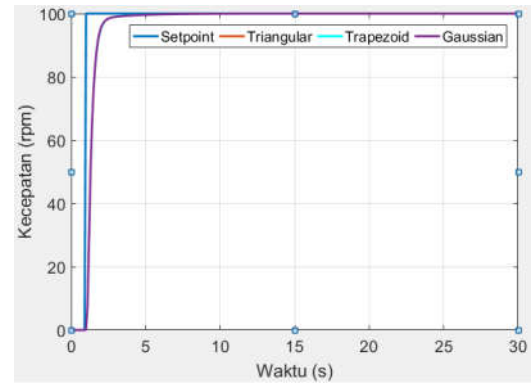


Figure 9 Step Response Results on 3 Membership Number

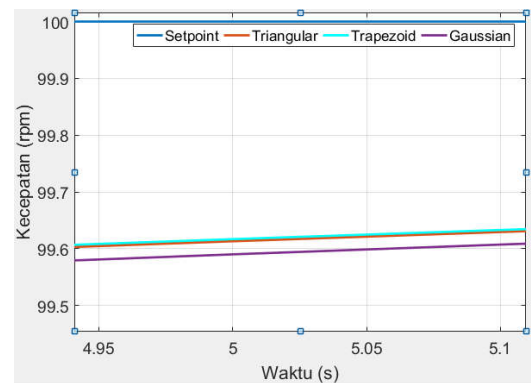


Figure 10 Zoomed in Result of Figure 9

Table 5 Quantitative Result of 3 Membership

F	P (rpm)	PT (s)	O	U	ST (s)
TL	100	25.9	-	-	2.3549
TP	100	25.9	-	-	2.3625
GS	100	25.8	-	-	2.3568

C. Fuzzy-PID Performance (5 Membership Comparison)

Test on 5 membership number uses the same parameter as 3 membership but this time, the time was directly treated as 30 ($t=30$). The results of the responses based on table 6 shows that the time it took for the responses to reach the setpoint is faster compared to the 3 membership. The same applies for settling time. Figure 11 shows the graphical results comparison of 5 memberships to the 3 different shapes. Because the difference is not very visible, the zoomed in figure of the result will also be shown in Figure 12.

Table 6 Quantitative Result of 5 Membership

F	P (rpm)	PT (s)	O	U	ST (s)
TL	100	25.1	-	-	2.3101
TP	100	25.5	-	-	2.3475
GS	100	25.5	-	-	2.3375

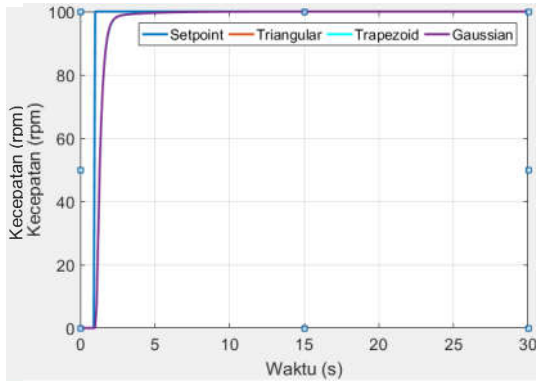


Figure 11 Step Response Results on 5 Membership Number

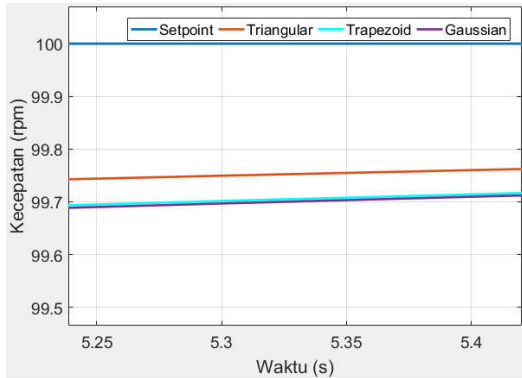


Figure 12 Zoomed in Result of Figure 11

D. Fuzzy-PID Performance (7 Membership Comparison)

Data from the 7 membership test results presented in table 7 shows that the triangle function reaches setpoint the fastest with a time of 23.7 seconds, followed by gaussian function at 23.9 seconds, and finally by the trapezoid function at 24.3 seconds. The fastest settling time is also achieved by the triangle function with a time of 2.3069 seconds which only 0.02 seconds away compared to the other functions.

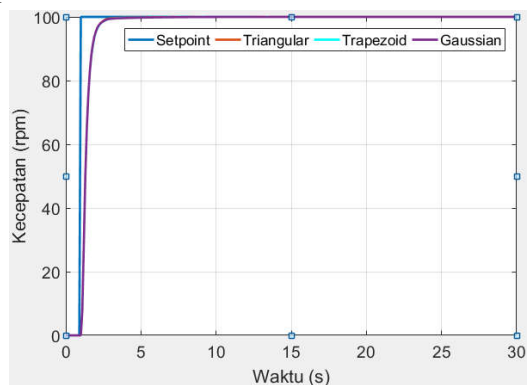


Figure 13 Step Response Results on 7 Membership Number

These results indicate an increase from the previous two test results both in terms of time to reach the setpoint and settling time. Graphical results of the comparison of functions in 7 memberships are shown in Figure 13, while the zoomed in results are shown in Figure 14.

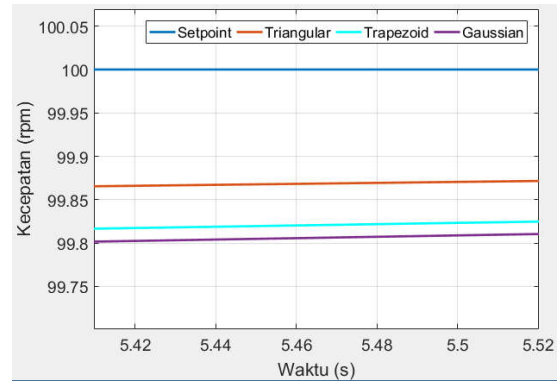


Figure 14 Zoomed in Result of Figure 13

Table 7 Quantitative Result of 7 Membership

F	P (rpm)	PT (s)	O	U	ST (s)
TL	100	23.7	-	-	2.3069
TP	100	24.3	-	-	2.3295
GS	100	23.9	-	-	2.32

The best response results from each membership number are taken and compared in table 8. It can be seen that the best function is achieved by the triangle function which always has the fastest settling time in the three test results. In terms of time to reach the setpoint, the triangle function is only lost to the gaussian function by a difference of 0.1 seconds in 3 memberships. For the number of memberships, the best result is achieved by the highest number of members, 7.

Table 8 Results Comparison of the Best Responses

Membership Number	Best Function	SPT (s)	O	U	ST (s)
3	Triangular	25.9	-	-	2.3549
5	Triangular	25.1	-	-	2.3101
7	Triangular	23.7	-	-	2.3069

E. PID and Fuzzy-PID Response Comparison

Table 9 shows that PID is able to reach a setpoint of exactly 100 rpm with a faster time compared to Fuzzy-PID despite experiencing overshoot at the start. The Fuzzy PID response used in this comparison is the best response, which is the triangle function with a membership number of 7.

Table 9 PID & Fuzzy-PID Step Response Comparison

Algorithm	SPT (s)	O	U	ST (s)	Computation Time (s)
PID	9.4	0.0017	-	2.2655	12.1
Fuzzy-PID	23.7	-	-	2.3069	0.54

F. Signal Tracking Performance

The last experiment is tracking. The Fuzzy-PID control used for this test is the control with the best results from previous experiment, which is fuzzy with a triangle function and membership number of 7. The highest setpoint to be achieved in this first tracking experiment is 1 rpm which is quite far from the setpoint tested before (100 rpm). Figure 15 shows the response of the two controllers to the signal shape change over time.

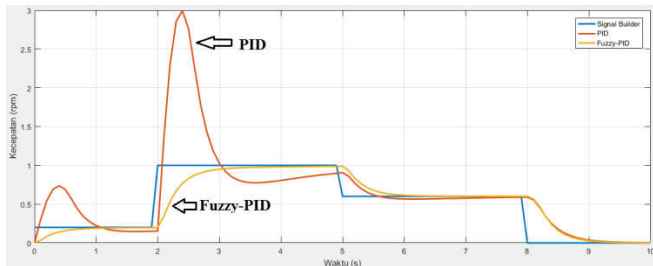


Figure 15 PID & Fuzzy-PID Response Comparison with maximum setpoint of 1 rpm

Based on Figure 15 it can be seen that both PID (orange line) and Fuzzy-PID (yellow line) are able to follow the changes in signal. However, PID often experiences overshooting, whereas Fuzzy-PID always manages to follow signals without overshoot and results in a finer response than PID. This is because Fuzzy-PID was set to always adjust the K_p gain parameter in accordance with the change in setpoint so that if there is a change in the setpoint either up or down, drastically or normally, the K_p gain value can be adjusted to increase or decrease.

This is also because the predetermined gain parameter is set for the 100 rpm setpoint both for PID and Fuzzy-PID. But Fuzzy-PID wasn't affected by the changes because it can adjust itself. If the tracking reaches setpoint 100 or at least not far from it, the results of the two are not that much different which can be seen in Figure 16.

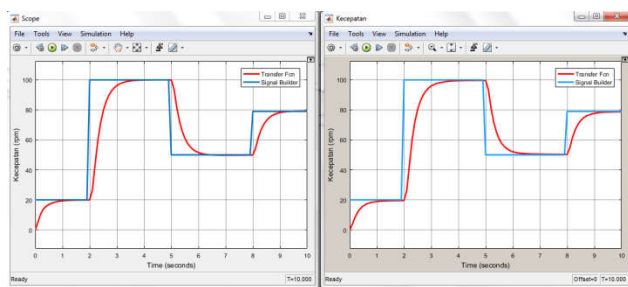


Figure 16 PID (left) and Fuzzy-PID (right) Response Comparison with maximum setpoint of 100 rpm

As explained before, the PID response is limited to a predetermined gain, so a significant change from 100 rpm to 1 rpm makes the PID set to 100 rpm unable to respond properly. Unlike the case with Fuzzy-PID which can adjust its K_p gain following the drastic changes of setpoint.

Another experiment was carried out on tracking with a signal around 0-20 (with maximum being 20 rpm) which

can be seen in Figure 17. The results showed that the PID experienced overshoot when trying to reach 10 rpm, but did not experience it when trying to reach 20 rpm, while Fuzzy-PID manage to maintain its stable response as always. This shows that the initial setting of the PID gain parameter is only able to tolerate a little changes (in this case as far as 20 rpm from the initial 100 rpm), whereas Fuzzy-PID can tolerate much greater changes than that.

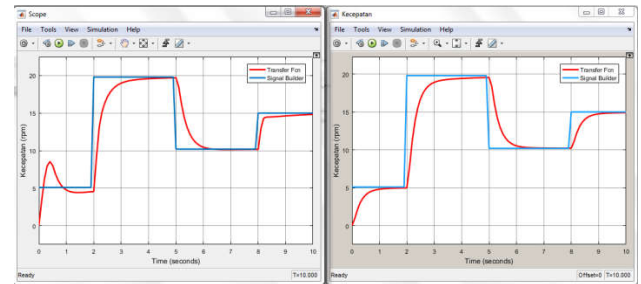


Figure 17 PID (left) and Fuzzy-PID (right) Response Comparison with maximum setpoint of 20 rpm

V. CONCLUSION

DC motor speed control with the Fuzzy-PID algorithm has been simulated and compared with the PID algorithm. The simulation results shows that the Fuzzy-PID algorithm has better performance than the PID algorithm in following setpoint changes that are quite far from the initial setting, but PID is superior in step unit response performance in terms of reaching setpoint much faster with fast computing time as well compared to Fuzzy-PID.

Comparisons have also been done to the effect of the membership function (shape) chosen and the number of memberships used in the fuzzy logic fuzzification process. The simulation result shows that the more number used as the membership, the faster the response time to reach the setpoint. Changes in membership functions (shape) only have little effect. In this study, the triangle function shows results with the best time compared to the trapezoid function and the gaussian function.

REFERENCES

- [1] D.G. Alciatore and M.B. Histand. "Actuators" in *Introduction to Mechatronic and Measurement Systems*, 4th ed. New York: McGraw-Hill, 2012.
- [2] Hassan, A.A., Al-Shamaa, N.K., & Abdalla, K.K. "Comparative Study for DC Motor Speed Control Using PID Controller," *International Journal of Engineering and Technology (IJET)*, vol.9, pp. 4181-4192, Desember 2017.
- [3] Rai, J.N., Singhal M., & Nandwani, M., "Speed Control of DC Motor Using Fuzzy Logic Technique" *IOSR Journal of Electrical and Electronics Engineering*, vol.3, pp. 41-48, Januari 2012
- [4] H. Maghfiroh, "Optimasi Sistem Kendali PID dengan *Double Tuning* dalam Implementasi Pengendalian Kecepatan Motor DC berbasis PLC," *Skipripsi*. p.39, 2913
- [5] Tripathi, N., Singh, R., & yadav R., "Analysis of Speed Control of DC Motor – A review study" *International Research Journal of Engineering and Technology (IRJET)*, vol.2, pp. 1616-1621, November 2015
- [6] Maghfiroh, H., Ataka, A., Wahyunggoro, O., & Cahyadi, A.I., "Optimal Energy Control of DC Motor Speed Control: Comparative

Study” 2013 International Conference on Computer, Control, Informatics and Its Applications, vol.13, pp. 89-93, November 2013

- [7] M. Mukaidono, *Fuzzy Logic for Beginners*, World Scientific, 2011.
- [8] Setiadji. *Himpunan & Logika Samar beserta Aplikasinya*. Graha Ilmu, 2009.
- [9] Lee, C.C., “Fuzzy Logic in control systems: fuzzy logic controller II,” in *IEEE Transactions on Systems, Man, and Cybernetics*, vol.20, no.2, pp. 419-435, Maret-April 1990.