

IMPLEMENTASI DAN ANALISIS KENDALI KECEPATAN MOTOR BLDC 1 kW MENGGUNAKAN ALGORITMA PID

Muchlis Dwi Ardiansyah¹, Fatkhur Rohman²

^{1,2}Teknik Otomotif-Elektronik Politeknik Negeri Malang

¹muchlisdwi56@gmail.com

²fatkhur_rohman@polinema.ac.id

Abstrak

Pemanfaatan teknologi alternatif dalam bidang otomotif maupun otomasi industri menggunakan motor *Brushless Direct Current* (BLDC) sudah banyak digunakan karena memiliki kelebihan dibanding dengan jenis mesin penggerak bertenaga elektrik lainnya. Namun motor BLDC masih memiliki beberapa kekurangan ketika menerima beban sehingga menyebabkan penurunan kecepatan putaran pada motor BLDC. Tujuan penelitian ini adalah untuk merancang dan mengaplikasikan sistem kendali kecepatan motor BLDC dengan kontrol *Proportional Integral Derivative* (PID) dan menentukan nilai parameter untuk mendapatkan persentase *error steady state* terkecil pada variasi kecepatan dan beban motor BLDC. Metode pengambilan data diambil dengan cara memasukkan nilai parameter secara *trial and error*. Sebagai simulasi beban, motor BLDC dihubungkan dengan generator yang diberi beban berupa lampu yang divariasikan. Hasil pengujian mendapatkan pemodelan blok diagram PID dengan Matlab Simulink. Hasil parameter kontrol PID diperoleh nilai $K_p = 0,8$; $K_i = 6$ dan $K_d = 0,02$. Dengan nilai parameter tersebut motor BLDC dapat mempertahankan nilai *set point* dengan kestabilan yang tinggi (*error steady state* rendah).

Kata-kata kunci: algoritma PID, *error steady state*, kendali kecepatan, motor BLDC

Abstract

The usage of alternative technology in the field of automotive and industrial automation using Brushless Direct Current (BLDC) has been widely used because it has advantages compared to other types of electric-powered drive engines. But the BLDC motor still has some disadvantages when receiving a load that causes a decrease in rotation speed on the BLDC motor. The purpose of the study is to design and apply a BLDC motor speed control system with a Proportional Integral Derivative (PID) control and determine the parameter value to obtain the smallest error steady state percentage at a speed variation and motor load the BLDC. The method of retrieving data was taken by entering parameter values by trial and error. As a load simulation, the BLDC motor was connected to a generator that was given the load in the form of a varied lamp. The test results gets the PID block diagram modeling with Matlab Simulink. The results of the PID control parameter are $K_p = 0,8$; $K_i = 6$ and $K_d = 0.02$. With these parameter values, the BLDC motor can maintain the setpoint value with high stability (low steady-state error).

Keywords: *BLDC motor, error steady state, PID algorithm, speed control.*

1. PENDAHULUAN

Teknologi berkembang sangat pesat dalam berbagai bidang baik dalam bidang pendidikan, industri, maupun transportasi yang pada dasarnya teknologi ini untuk membantu memenuhi kebutuhan manusia dalam melakukan kegiatan. Seperti dalam bidang transportasi dan otomasi industri yaitu mengembangkan motor *Brushless Direct Current* (BLDC).

Motor BLDC sudah banyak digunakan karena memiliki kelebihan motor BLDC adalah efisiensi lebih tinggi, dimensi lebih kecil. Selain itu, dengan tidak adanya sikat arang maka perawatan menjadi ringan, hampir tidak ada noise. Kelebihan lain yaitu tanggapan lebih cepat, umur pakai yang lama, dan memiliki kecepatan yang tinggi [1]. Namun motor BLDC masih memiliki beberapa kekurangan, yaitu ketika motor bekerja dan menerima beban maka akan menyebabkan penurunan kecepatan putaran pada motor BLDC.

Pengendalian kecepatan motor BLDC perlu dilakukan agar respon yang dihasilkan sesuai dengan yang diharapkan. Terdapat beberapa metode untuk mengembalikan respon salah satunya yaitu menggunakan kontrol *Proportional Integral Derivative* (PID).

Dengan adanya kontrol PID diharapkan kestabilan kecepatan putaran motor BLDC dapat mencapai *set point* meskipun motor mendapat beban.

2. KAJIAN PUSTAKA

2.1 PENELITIAN SEBELUMNYA

Pengontrolan motor BLDC dengan kontrol PID sudah banyak diteliti. Dengan adanya kontrol PID kecepatan motor dapat dikendalikan secara otomatis dan respon dari motor tersebut akan lebih baik.

Penelitian sebelumnya tentang sistem Kontroler PID bertujuan untuk pengembangan mobil listrik khususnya di bagian penggerak utamanya. Kontrol penggerak mobil listrik menggunakan kontrol PID yang diperintah melalui program ROS (*Robot Operating System*). [2]

Penelitian tersebut menghasilkan kesimpulan:

- Kontrol penggerak mobil listrik menggunakan kontrol PID yang diperintah melalui program ROS (*Robot Operating System*) yang saling terintegrasi dengan box kontrol atau ecu, yang mana ecu tersebut terdiri dari Mikrokontroler ATmega128, FTDI *Breakdown*, serta *Stepdown Kit* LM2596 yang mampu mengendalikan kecepatan motor BLDC 5 kW.
- Pengendalian kecepatan motor BLDC 5 kW didapatkan parameter PID hasil optimasi dengan $K_p = 11$, $K_i = 1$, dan $K_d = 10$ yang cocok dari 1000 – 4000 rpm.

Hasil penelitian lain tentang *performance* motor BLDC menyampaikan bahwa kendali kecepatan motor BLDC ditentukan oleh besarnya gangguan yang diberikan pada system. Pada penelitian ini juga menyampaikan bahwa motor yang diberi beban akan mengalami perubahan putaran sehingga perlu ditambahkan system kendali PID. [8]

2.2 BRUSHLESS DIRECT CURRENT MOTOR (BLDC)

Brushless Direct Current (BLDC) adalah motor sinkron dengan magnet permanen yang menggunakan sensor posisi dan sebuah rangkaian *inverter* untuk mengendalikan arus *armature*. Proses Bergeraknya motor BLDC tidak memerlukan *Brush* (sikat) untuk proses komutasi, tetapi digantikan dengan gulungan yang terhubung dengan kontrol elektronik. Jadi pada motor BLDC proses komutasinya menggunakan elektronik [7].



GAMBAR 1 MOTOR BLDC

TABEL 1 SPESIFIKASI MOTOR BLDC 1kW

Model	BLDC Disk Motor
<i>Input Voltage</i>	48V – 60V
Daya	800W (48V) dan 1000W (60V)
Putaran Input	3100 rpm
Putaran Output	512 rpm
<i>Gear Ratio Internal Gearbox</i>	1 : 6
Arus Tanpa Beban	2 – 5 A
Arus Dengan Beban	20 – 30 A
Torsi	15,26 Nm

2.3 SENSOR ROTARY ENCODER

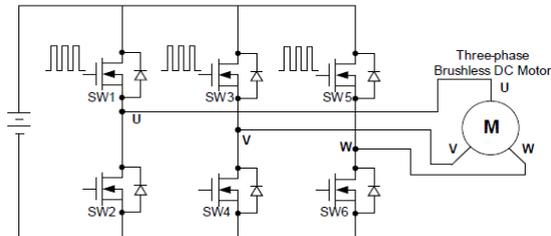
Rotary encoder adalah suatu sensor yang dapat memonitor gerakan dan posisi. *Rotary encoder* menggunakan sensor optik untuk menghasilkan pulsa yang dapat diartikan menjadi gerakan, posisi, dan arah. Sehingga posisi sudut poros dapat diolah menjadi informasi berupa kode digital oleh *rotary encoder* untuk dikirimkan ke rangkaian kendali [3].



GAMBAR 2 ROTARY ENCODER

2.4 INVERTER 3 FASA

Kontroler dan inverter berfungsi untuk merubah tegangan searah (DC) menjadi tegangan bolak – balik (AC) dan sebagai pengganti sikat dan komutator pada motor DC konvensional. Motor BLDC memerlukan inverter 3 fasa yang terdiri dari 6 komponen *switching*.



GAMBAR 3 THREE PHASE BRIDGE

2.5 SISTEM PENGENDALIAN PID

Pengendalian dengan metode PID sudah banyak digunakan di dunia rekayasa kendali. Sistem kendali PID adalah sistem pengontrolan dari kombinasi tiga macam kendali yaitu, kendali *propotional*, kendali *integral*, dan kendali *derivative* [4].

2.6 MATLAB

Matlab adalah singkatan dari Matrix Laboratory. Matlab awalnya ditulis untuk memberikan akses mudah ke perangkat lunak matriks yang dikembangkan oleh proyek LINPACK (paket sistem linier) dan EISPACK (paket sistem Eigen). Matlab adalah bahasa kinerja tinggi untuk komputasi teknis [6].

2.7 ARDUINO UNO R3

Arduino merupakan board mikrokontroler yang menggunakan IC ATmega328 yang bisa diprogram dan bersifat *open-source*.

Arduino tidak membutuhkan perangkat keras tambahan (*programmer* atau *downloader*) untuk *upload* program baru ke mikrokontroler.



GAMBAR 3 ARDUINO UNO R3

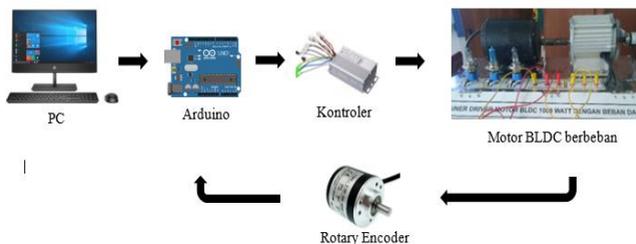
3. METODOLOGI

Jenis penelitian yang dilakukan ini adalah penelitian eksperimen yang dilakukan dengan cara mencoba secara *trial and error* terhadap variabel penelitian. Percobaan dilakukan dengan merubah nilai parameter K_p , K_i dan K_d dan diamati nilai *error steady state* yang dihasilkan. Data penelitian didapatkan dengan menghitung nilai *error steady state* yaitu menggunakan persamaan:

$$Ess(\%) = \frac{\text{nilai error steady state}}{\text{nilai set point}} \times 100\%$$

3.1 BLOK DIAGRAM PENELITIAN

Untuk tujuan pengambilan data Gambar 4 menunjukkan perencanaan blok diagram untuk penelitian ini.



GAMBAR 4 DIAGRAM BLOK SISTEM

3.2 RANCANG BANGUN SOFTWARE

3.2.1 PERANCANGAN PROGRAM PWM

Pembuatan program PWM pada kontrol PID dibuat pada *software* Matlab Simulink, komponen yang digunakan pada

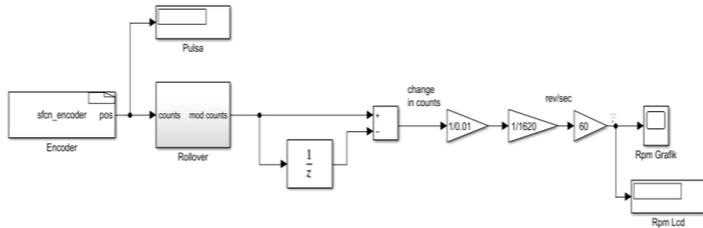
program ini terdiri dari constant, slider gain dan blok PWM. Pin PWM yang digunakan adalah pin 9 pada *board* Arduino Uno.



GAMBAR 5 BLOK DIAGRAM PWM

3.2.2 PERANCANGAN PROGRAM KECEPATAN

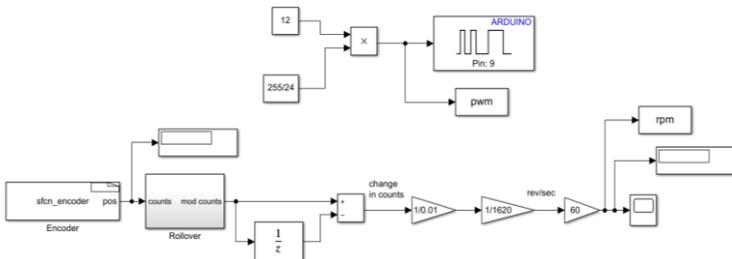
Program kecepatan menggunakan blok encoder, pembacaan kecepatan dari *rotary encoder* dihubungkan dengan sum sebagai *feedback*.



GAMBAR 6 BLOK DIAGRAM SENSOR KECEPATAN

3.2.3 PERANCANGAN PROGRAM *OPEN LOOP*

Pembuatan program *open loop* bertujuan untuk mendapatkan fungsi alih (*transfer function*) motor BLDC yang akan digunakan sebagai simulasi pada *Software* Matlab Simulink.



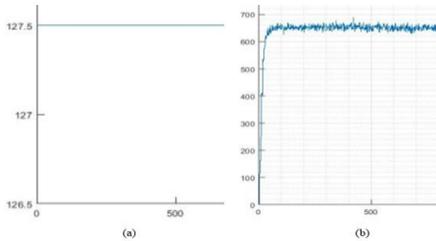
GAMBAR 7 BLOK DIAGRAM PROGRAM *OPEN LOOP*

3.2.4 PERMODELAN SISTEM

Permodelan sistem bertujuan untuk mendapatkan nilai *transfer function* dari motor BLDC.

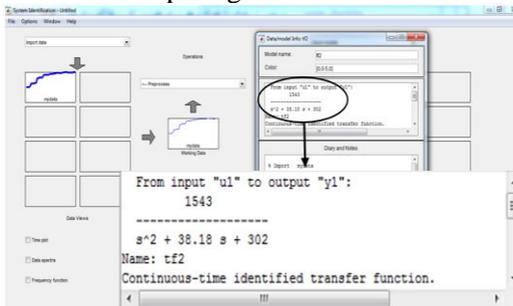
a) IDENTIFIKASI SISTEM

Proses identifikasi sistem akan mendapatkan model pengganti dari motor BLDC menjadi blok *transfer function*. Untuk mendapatkan nilai *transfer function* perlu memasukkan grafik input dan output dari sistem pada *System Identification Toolbox* pada Matlab Simulink.



GAMBAR 8 PENGUJIAN KARAKTERISTIK (A) INPUT PWM (B) OUPUT KECEPATAN

Grafik input dan output diolah dimasukkan kedalam *System Identification Toolbox* seperti gambar dibawah ini:



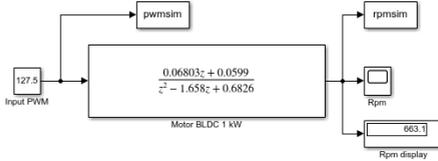
GAMBAR 9 SYSTEM IDENTIFICATION TOOLBOX MATLAB

Nilai *transfer function* tersebut berbentuk tranformasi laplace, sehingga perlu diubah ke transformasi z, dan mendapatkan nilai sebagai berikut:

$$tf = \frac{0,06803z + 0,0599}{z^2 - 1,658z + 0.6826}$$

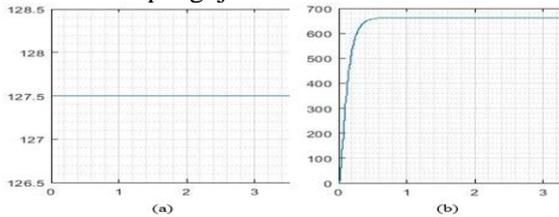
b) VALIDASI TRANSFER FUNCTION

Validasi ini digunakan untuk memastikan bahwa nilai *transfer function* akurat.



GAMBAR 10 BLOK DIAGRAM TRANSFER FUNCTION

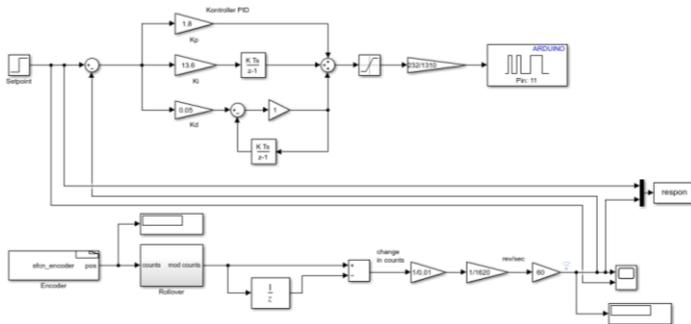
Grafik hasil simulasi pengujian:



GAMBAR 11 SIMULASI KARAKTERISTIK (A) INPUT (PWM) DAN (B) OUTPUT (KECEPATAN)

Pembacaan kecepatan pada simulasi dan percobaan langsung menunjukkan hasil yang relatif sama.

3.2.5 PERANCANGAN PROGRAM KONTROL PID



GAMBAR 12 BLOK DIAGRAM KONTROL PID

4. HASIL DAN PEMBAHASAN

4.1 PENGUJIAN *OPEN LOOP*

Pengujian sistem *open loop* dengan beban sebesar 86,5 Watt dapat dilihat bahwa kecepatan motor BLDC mengalami penurunan.



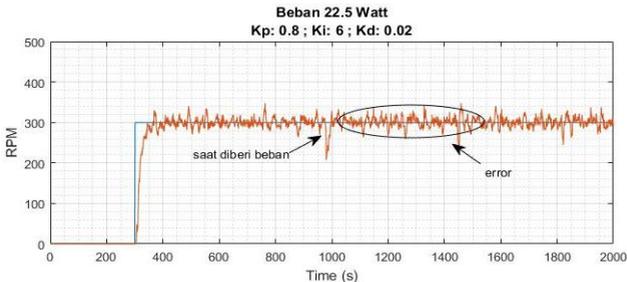
GAMBAR 13 PERUBAHAN KECEPATAN AKIBAT PEMBEBANAN 86,5WATT

4.2 PENGUJIAN *CLOSE LOOP*

Langkah-langkah pengujian *close loop* dengan beban lampu 22,5 Watt, 65 Watt, 86,5 Watt. Motor BLDC dioperasikan kemudian diberikan beban tersebut. Melihat respon kendali, dengan mengamati pembacaan nilai besaran kecepatan pada tampilan scope yang terdapat pada *software* Matlab Simulink kemudian ditampilkan pada monitor PC.

4.2.1 Pengujian *Close Loop* dengan Beban Lampu 22,5 Watt

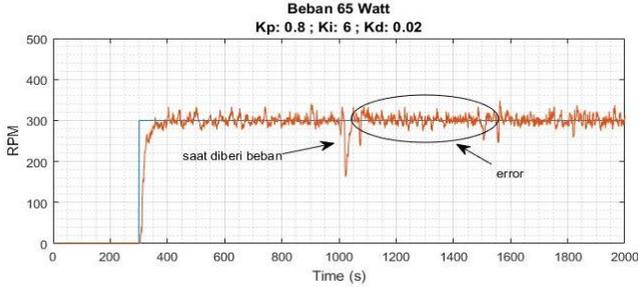
Hasil pengujian dengan beban lampu 22,5 Watt dengan sistem kendali PID.



GAMBAR 14 *ERROR STEADY STATE* BEBAN 22,5WATT DAN KECEPATAN 300RPM

4.2.2 Pengujian *Close Loop* dengan Beban Lampu 65 Watt

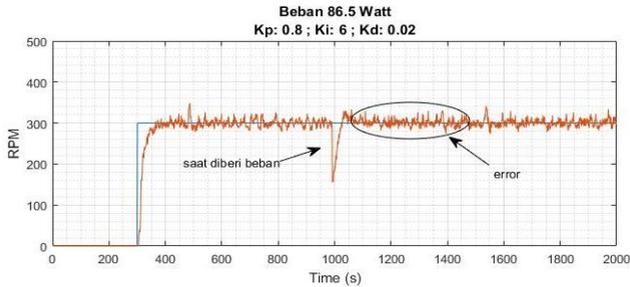
Hasil pengujian dengan beban lampu 65 Watt dengan sistem kendali PID.



GAMBAR 15 *ERROR STEADY STATE* BEBAN 65 WATT DAN KECEPATAN 300RPM

4.2.3 Pengujian *Close Loop* dengan Beban Lampu 86,5 Watt

Hasil pengujian dengan beban lampu 86,5 Watt dengan sistem kendali PID.



GAMBAR 16 *ERROR STEADY STATE* BEBAN 86,5 WATT DAN KECEPATAN 300RPM

TABEL 2 *ERROR STEADY STATE (%)* PADA 300 RPM

Parameter	Beban (Watt)			Rata-rata
	22.5	65	86.5	
Kp = 0,8 Ki = 6 Kd = 0,02	10,22	11,22	8,00	9,81
Kp = 1,6 Ki = 17,2 Kd = 0,04	11,00	13,33	10,44	11,59

$K_p = 0,5$ $K_i = 7,2$ $K_d = 0,01$	10,33	11,78	9,78	10,63
$K_p = 1,5$ $K_i = 10,5$ $K_d = 0,04$	10,33	11,33	9,67	10,44
$K_p = 1,8$ $K_i = 13,6$ $K_d = 0,05$	9,00	11,33	10,22	10,18

Pada kecepatan 300 rpm mendapatkan nilai *error steady state* terkecil dengan parameter $K_p = 1,8$; $K_i = 6$ dan $K_d = 0,02$ dengan rata – rata *error* sebesar 9,81%.

TABEL 3 OVERSHOOT (%) PADA 300 RPM

Parameter	Beban (Watt)			Rata-rata
	22.5	65	86.5	
$K_p = 0,8$ $K_i = 6$ $K_d = 0,02$	25,22	38,64	28,12	30,66
$K_p = 1,6$ $K_i = 17,2$ $K_d = 0,04$	38,02	81,05	50,06	56,38
$K_p = 0,5$ $K_i = 7,2$ $K_d = 0,01$	37,73	58,72	67,26	54,69
$K_p = 1,5$ $K_i = 10,5$ $K_d = 0,04$	83,54	40,00	38,65	54,06
$K_p = 1,8$ $K_i = 13,6$ $K_d = 0,05$	39,87	52,23	38,77	43,62

Pada table pembacaan diatas didapatkan bahwa dengan kecepatan yang sama dan beban yang bervariasi nilai *overshoot* terkecil didapatkan dari parameter PID $K_p = 0,8$; $K_i = 6$ dan $K_d = 0,02$.

5. KESIMPULAN

Dari hasil pengujian yang telah dilakukan, ada beberapa kesimpulan yang dapat diambil yang diantaranya adalah sebagai berikut:

1. Perancangan dan pengimplementasian kontrol PID pada motor BLDC menggunakan mikrokontroler dengan menambahkan sensor *rotary encoder*. Pemodelan blok diagram sistem PID menggunakan Matlab Simulink.
2. Ketika motor mendapat beban, nilai parameter K_p , K_i , K_d berpengaruh terhadap kestabilan kecepatan putar (rpm) motor. Sehingga dengan penambahan nilai parameter, motor BLDC dapat mempertahankan nilai *set point* yang diharapkan dengan kestabilan yang tinggi (*error steady state*)

rendah). Nilai parameter yang didapatkan $K_p = 0,8$ $K_i = 6$ dan $K_d = 0,02$.

6. DAFTAR PUSTAKA

- [1] Hidayat. (2014) Pengembangan Hybrid PID-ANFIS (Proportional Integral Derivative-Adaptive Neuro Fuzzy Inference Systems) sebagai Pengendali Kecepatan Mesin Arus Searah Tanpa Sikat (MASTS). Desertasi Universitas Gadjah Mada. Yogyakarta.
- [2] Hardianto, A. R. (2017). *Pembuatan dan Analisis Kontroler PID Untuk Kecepatan Motor BLDC*. Malang.
- [3] Risqiawan, A. (2009). *Sekilas Rotary Encoder*. Penelitian Energi Listrik ITB.
- [4] Rosalina., Qosim, I., Mujirudin, M. 2017. Analisis Pengaturan Kecepatan Motor DC Menggunakan Kontrol PID (*Proportional Integral Derivative*). *Seminar Nasional TEKNOKA*, 2, 89-94.
- [5] Gustomo, B. (2015). *Arduino Uno*. Dipetik Januari 9, 2019, dari https://www.academia.edu/14708355/Arduino_Uno
- [6] Natick, M. (2018). *Matlab Programming Fundamentals*. The MathWorks Inc
- [7] Yadamale, P. (2003). *Brushless DC (BLDC) Motor Fundamentals*. U.S.A: Microchip Technology Incorporated.
- [8] Zoni, M. (2017). Perbaikan Performance Kendali Kecepatan Motor Brushless DC (BLDC) dengan Pengendali Logika Fuzzy. *SENASPRO*, 299.
- [9] Zhao, J. (2011). *Brushless DC Fundamentals*. MPS Technology.