

SISTEM PENGENDALIAN MOTOR DC MENGGUNAKAN PID DENGAN METODE ZIEGLER – NICHOLS (IMPLEMENTASI PALANG PINTU PARKIR)

Rio Febrianto Nugroho 20150120133

Program Studi Teknik Elektro, Fakultas Teknik
Universitas Muhammadiyah Yogyakarta
Jalan Lingkar Selatan Taman Tirto, Bantul, Yogyakarta
E-mail : rionugroho70@gmail.com

Abstrak

Kebutuhan perpajakan menjadi kebutuhan yang mendesak dewasa ini, terutama pada daerah yang berkembang pesat. Sistem palang parkir yang digunakan terus menerus dalam jangka waktu lama mengakibatkan respon kepada motor penggerak menjadi kurang optimal. Kontrol motor diperlukan untuk mendapatkan pergerakan dari kecepatan motor atau posisi motor agar sesuai dengan nilai yang telah ditetapkan. Sistem kontrol motor yang digunakan dalam penelitian ini adalah sistem kontrol PID dengan menerapkan metode penalaan PID *Ziegler-Nichols* aturan pertama menggunakan mikrokontroler Arduino Uno. Pada metode ini berlangsung dua tahap yaitu tahap penalaan fungsi alih motor DC dan tahap penalaan dengan beberapa model penalaan. Dari beberapa model penalaan parameter PID diperoleh hasil parameter PID yang optimal yaitu $K_p = 4,41$; $K_i = 3,556$; $K_d = 1,367$. Dengan nilai K_p , K_i dan K_d tersebut motor DC dapat berjalan dengan baik, menghasilkan *settling time* cepat dan dapat mempertahankan *setpoint*.

Kata kunci : Sistem Kendali, PID, *Ziegler-Nichols*, Motor DC, Kurva S

I.PENDAHULUAN

Parkir adalah tempat pemberhentian kendaraan bermotor dalam jangka waktu yang pendek atau lama, sesuai dengan kebutuhan pengendara. Kebutuhan perpajakan menjadi kebutuhan yang mendesak dewasa ini, terutama pada daerah yang berkembang pesat. Perpajakan merupakan bidang jasa yang harus dikelola dengan baik seperti di rumah sakit, perkantoran, perbelanjaan dan tempat lainnya. Jika tidak dikelola penataan area parkir dengan benar maka dikhawatirkan kemacetan dan kedisiplinan kendaraan yang ada di area parkir tidak dapat dihindari. Sistem palang parkir yang digunakan terus menerus dalam jangka waktu lama mengakibatkan respon kepada motor penggerak menjadi kurang optimal. Hal ini dapat menyebabkan penumpukan di gerbang pintu parkir. Maka dari itu perlu adanya pengoptimalan pada kontrol motor penggerak

palang parkir untuk mendapatkan respon yang optimal.

Kontrol motor diperlukan untuk mendapatkan pergerakan dari kecepatan motor atau posisi motor agar sesuai dengan nilai yang telah ditetapkan. Kemampuan menjaga kecepatan dan posisi agar tetap sesuai dengan *set point* ketika diberi beban maupun kemampuan sistem untuk mengejar kecepatan dan posisi agar mencapai *set point* ketika motor mulai berjalan menjadi faktor yang sangat penting sebagai ukuran kinerja pengontrolan kecepatan dan posisi. Cara pengontrolan motor dengan memberikan sinyal kontrol. Pada umumnya kontrol motor di desain menggunakan kontroler PID.

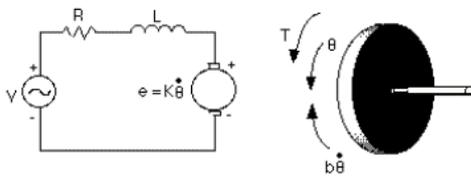
Kontroler *Proportional-Integral-Derivatif* (PID) adalah kontroler yang paling populer saat ini, karena keefektifannya, sederhana dalam implementasi dan luas penggunaannya. Konfigurasi standar kontroler PID memiliki parameter-parameter K_p , K_i dan K_d yang dapat ditentukan agar karakteristik

plant sesuai dengan kriteria desain yang diharapkan. Spesifikasi umum dalam desain adalah *rise-time*, *settling-time*, *maximum overshoot* dan *error steady state* terhadap masukan yang diberikan:

Pada penelitian pengendalian palang parkir ini menggunakan motor DC 25GA370 sebagai aktuator dan objek untuk dikendalikan. Pemilihan motor DC ini, dikarenakan mempunyai *encoder* yang terhubung dengan porosnya yang diharapkan dapat menjadi sinyal umpan balik. Pemilihan penalaan PID dengan *Ziegler - Nichols* bertujuan agar mendapatkan parameter PID yang tepat dan sesuai, karena metode *Ziegler - Nichols* adalah metode penalaan PID yang sering digunakan. Selain itu, pada penelitian ini juga dilakukan pengolahan data hasil pembacaan *error*, membandingkan nilai masukan dengan nilai umpan balik, dengan tujuan untuk mengetahui nilai *rise time*, *settling time*, *maximum overshoot* dan *steady state error*.

II.DASAR TEORI

2.1. Pemodelan Motor DC



Gambar 2. 1 Permodelan rangkaian listrik dari Motor DC

Dalam Gambar 2. 1, torsi motor dapat dinaikan secara linier dengan menaikkan arus *armature*, persamaan torsi yang dibangkitkan motor dc sebagai berikut :

$$T_m = K_a i \quad (2.13)$$

Fungsi alih dari masukan arus armatur yang menghasilkan torsi motor sebagai berikut:

$$\frac{T_m(s)}{I(s)} = K_a \quad (2.14)$$

Dimana K_a adalah konstanta armature motor yang bergantung pada jumlah lilitan, jumlah kutub medan dan luas penampang pada armature. Saat motor berputar besarnya tegangan ggl induksi adalah sebanding. Dengan konstanta motor K_b dan kecepatan sudut

putaran motor ω turunan pertama dari posisi sudut putaran motor θ .

$$e = K_b \omega \quad (2.15)$$

Menurut hukum newton, bahwa persamaan torsi dengan momen inersia dan rasio redaman dari motor sebagai berikut :

$$T_m = J \frac{d^2 \theta(t)}{dt^2} + b \frac{d\theta(t)}{dt} \quad (2.16)$$

Dari persamaan 2.13 dan 2.16 diperoleh

$$J \frac{d^2 \theta(t)}{dt^2} + b \frac{d\theta(t)}{dt} = K_a i \quad (2.17)$$

Untuk besarnya tegangan V menurut hukum kirchoff adalah

$$V = Ri + L \frac{di}{dt} + e \quad (2.18)$$

$$V = Ri + L \frac{di}{dt} + K_b \omega \quad (2.19)$$

Dengan transformasi laplace persamaan 2.17 dan 2.19 kemudian menstsubstitusikan diperoleh fungsi alih motor, dimana $K_a = K_b$

$$(Js^2 + bs)\theta(s) = K i(s) \quad (2.20)$$

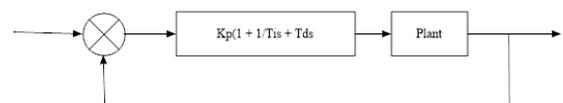
$$V = (R + Ls)i(s) + K\theta(s) \quad (2.21)$$

$$(R + Ls)i(s) = V - K\theta(s) \quad (2.22)$$

Sehingga

$$\frac{\theta}{V} = \frac{K}{s((Js+b)(Ls+R)+K^2)} \quad (2.23)$$

2.2. PID Kontroler

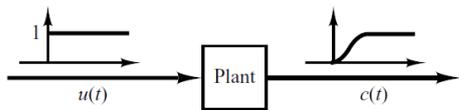


Gambar 2. 2 Blok diagram PID kontroler dari plant

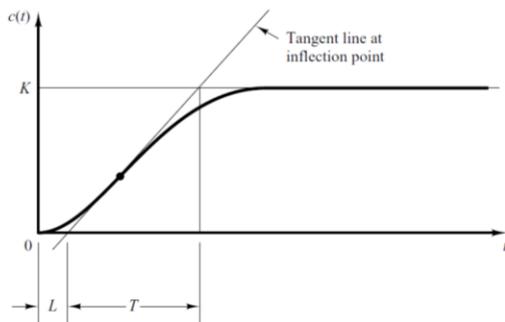
Pada Gambar 2. 2 memperlihatkan kendali PID dari sebuah *plant*. Jika diperoleh sebuah model matematika dari *plant* dapat diaplikasikan berbagai bentuk desain untuk menentukan parameter *transient* dan spesifikasi *steady state* dari sebuah sistem *loop* tertutup. Proses memilih parameter pengendalian untuk mendapatkan kinerja yang baik disebut sebagai *tuning*. Salah satu metode *tuning* yang digunakan adalah metode *Ziegler-Nichols*, dengan mengatur nilai K_p , T_i dan T_d menggunakan aturan berdasarkan eksperimen *step respons* atau berdasarkan nilai dari K_p

yang menghasilkan batas kestabilan saat menggunakan pengendalian proposional saja. Aturan *Ziegler* dan *Nichols* bertujuan menentukan nilai penguatan proposional K_p , waktu integral T_i , waktu turunan T_d berdasarkan karakteristik identifikasi proses yang diberikan oleh *plant*. Seperti itu penentuan parameter kontroler PID dilakukan. Terdapat 2 metode dari aturan *Ziegler – Nichols* yaitu metode pertama dan metode kedua.

Aturan Pertama



Gambar 2. 3 Unit step response of plant (sumber : Ogata, 2002)



Gambar 2. 4 Kurva S (sumber : Ogata, 2002)

Di metode pertama menggunakan sebuah eksperimen masukan respon *unit-step* dari sebuah *plant* yang diperlihatkan Gambar 2. 3. Jika *plant* tidak melibatkan *integrator(s)* juga kutub konjugasi yang dominan, maka kurva respon *unit-step* seperti berbentuk huruf S pada Gambar 2. 4. Kurva S dapat dicirikan dengan 2 konstanta yaitu waktu tunda L dan waktu konstan T . Waktu tunda dan waktu konstan ditentukan dengan menggambar sebuah garis tangen pada sebuah titik infleksi dari kurva S dan menentukan simpangan dari garis tangen dengan sumbu waktu dan garis $c(t) = K$ yang diperlihatkan Gambar 2. 4.

Tabel 2. 1 Formula Penalaan PID pada Aturan Pertama

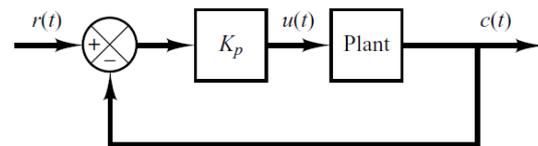
Jenis Pengendalian	K_p	T_i	T_d
P	$\frac{T}{L}$	∞	0
PI	$0,9 \frac{T}{L}$	$\frac{L}{0,3}$	0
PID	$1,2 \frac{T}{L}$	$2L$	$0,5L$

Fungsi alih $C(s)/U(s)$ dapat diperkirakan menggunakan sistem orde pertama dengan sebuah *transport lag* :

$$\frac{C(s)}{U(s)} = \frac{K e^{-Ls}}{Ts+1} \quad (2.26)$$

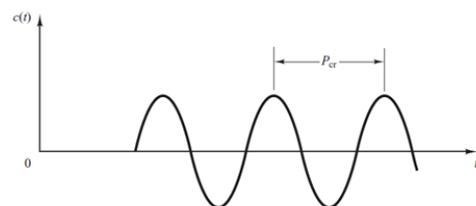
Ziegler-Nichols menyarankan untuk menetapkan nilai dan sesuai dengan rumus pada Tabel. Fungsi alih kontroler PID yang diatur menggunakan metode aturan pertama *Ziegler-nichols*, yang mana $s = -1/L$:

Aturan Kedua



Gambar 2. 5 Sistem loop tertutup dengan penguatan proposional (sumber : Ogata, 2002)

Pada metode kedua, dimana mengatur $T_i = \infty$ dan $T_d = 0$. Dengan hanya menggunakan aksi pengendali proposional saja, dengan meningkatkan nilai K_p dari 0 ke nilai kritis K_{cr} yang mana keluaran pertama menunjukkan osilasi yang berkelanjutan.



Gambar 2. 6 Osilasi dengan periode P_{cr} (sumber : Ogata, 2002)

Perolehan nilai kritis K_{cr} dan periode P_{cr} yang ditentukan secara eksperimen yang dapat dilihat pada Gambar 2. 6. Menetapkan nilai parameter K_p , T_i dan T_d sesuai dengan rumus yang ditunjukkan pada Tabel 2.2

Tabel 2. 2 Formula Penalaan PID pada Aturan Kedua

Jenis Pengendalian	K_p	T_i	T_d
P	$0,5K_{cr}$	∞	0
PI	$0,45K_{cr}$	$\frac{1}{1,2}P_{cr}$	0
PID	$0,6K_{cr}$	$0,5P_{cr}$	$0,125P_{cr}$

Di aturan kedua ini jika sebuah sistem tersebut diketahui model matematisnya dapat menggunakan metode *root-locus* untuk menentukan penguatan kritis K_{cr} dan frekuensi osilasi yang berkelanjutan ω_{cr} , dimana $2\pi/\omega_{cr} = P_{cr}$. Nilai – nilai tersebut dapat ditentukan dari titik persimpangan cabang *root-locus* dengan sumbu $j\omega$

2.3. Identifikasi Proses

Usaha yang dilakukan untuk mencari model matematis dan menentukan parameter proses yang optimal secara eksperimen. Kemudahan untuk menganalisa dari sistem yang cukup kompleks maka dilakukan pendekatan model matematis dengan menganggap sistem tersebut adalah sistem orde satu seperti pada persamaan berikut:

$$G(s) = \frac{K_s e^{-t_d s}}{\tau s + 1} \quad (2.36)$$

Dimana $G(s)$ merupakan fungsi alih proses orde 1 dengan waktu tunda, K_s merupakan penguatan statis dari proses, t_d merupakan waktu tunda dan τ merupakan konstanta waktu. Penguatan statis ditentukan dari.

$$K_s = \frac{\Delta PV}{\Delta MV} \quad (2.37)$$

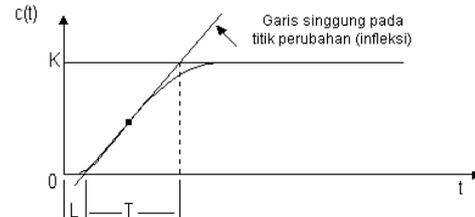
Dimana,

ΔPV adalah deviasi amplitude sinyal keluaran proses

ΔMV adalah deviasi dari amplitude sinyal kendali

Ada beberapa model yang dapat diterapkan dalam menganalisa kurva tanggapan *loop* terbuka dari sebuah proses, antara lain dengan menggunakan model tangen, model *smith* dan model dua titik.

1) Model Tangent



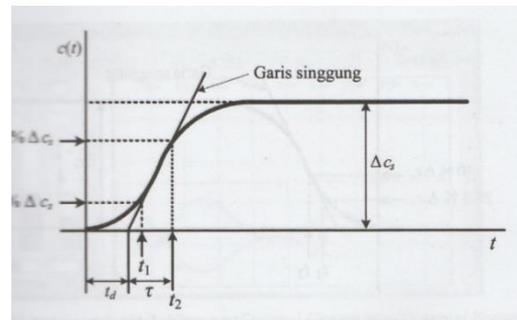
Gambar 2. 7 Identifikasi proses model tangen

Metode tangen terdiri dari menggambar garis singgung dari proses kurva reaksi pada titik belok. Penguatan dihitung dari amplitude masukan *step* dan keluaran nilai akhir *steady state*. Waktu constant dan waktu tunda dihitung dari titik persimpangan dari garis singgung dengan garis keluaran nilai akhir *steady state* dan sumbu waktu.

Dimana, $L = \text{waktu tunda}$

$T = \text{waktu konstan}$

2) Model Smith



Gambar 2. 8 Identifikasi proses model smith (sumber: Cekmas, 2017)

Metode ini didasarkan pada dua contoh waktu dari reaksi proses kurva yang sesuai dengan waktu t_1 dan t_2 dimana output proses mencapai 28,3% dan 63,2%. Waktu konstan dan waktu tunda dihitung dari t_1 dan t_2 . Dari Gambar 2. 8 waktu konstan dan waktu tunda dapat dihitung sebagai berikut :

$$t_1 = t_{(28,3\%)} \text{ dan } t_2 = t_{(63,2\%)} \quad (2.38)$$

$$t_1 = t_d + \frac{1}{3}\tau \quad (2.39)$$

$$t_2 = t_d + \tau \quad (2.40)$$

Dimana,

$$\tau_d = L = \text{waktu tunda}$$

$$\tau = T = \text{waktu konstan}$$

3) Model Dua Titik

Metode ini hampir sama dengan metode *smith* namun waktu t_1 dan t_2 dimana output proses mencapai 30% dan 60%. Waktu konstan dan waktu tunda dapat dihitung dari t_1 dan t_2 sebagai berikut :

$$t_1 = t_{(30\%)} \text{ dan } t_2 = t_{(60\%)} \quad (2.41)$$

$$\tau = \frac{3(t_2 - t_1)}{2} \quad (2.42)$$

$$\tau_d = t_2 - t_1 \quad (2.43)$$

Dimana,

$$\tau_d = L = \text{waktu tunda}$$

$$\tau = T = \text{waktu konstan}$$

III. PERENCANGAN DAN PEMBUATAN SISTEM

3.1. Gambaran Sistem

Sistem yang akan dirancang pada penelitian ini disajikan dalam bentuk blok diagram perancangan sistem



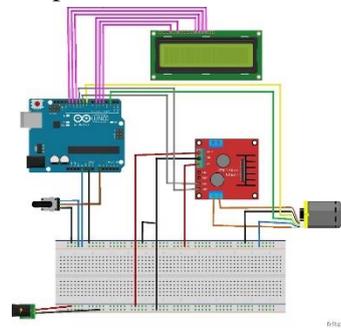
Gambar 3. 1 Blok diagram sistem keseluruhan

Pada penelitian ini dilakukan sebagai salah satu cara untuk mengoptimalkan sistem palang parkir. Dari Gambar 3.2 rancangan tersebut pada bagian awal atau masukan berupa nilai analog yang berasal dari nilai adc potensiometer. Nilai analog akan dibaca oleh mikrokontroler Arduino, di dalam mikrokontroler nilai tersebut diolah menjadi logika keluaran kemudian dikirimkan ke motor *driver* yang mana untuk menggerakkan Motor DC tersebut. Berputarnya poros motor dc membangkitkan gelombang pada *encoder* yang terhubung dengan poros motor dc. Jumlah gelombang yang dibangkitkan oleh *encoder* akan dikirimkan ke mikrokontroler kemudian dibandingkan dengan nilai adc potensiometer sehingga menghasilkan *error* yang nantinya, *error* tersebut akan di kalikan atau di kendalikan dengan PID kontroler yang terdapat

di dalam mikrokontroler Arduino. Hal tersebut akan terus berulang dilakukan hingga mencapai suatu keadaan yang mana jumlah gelombang *encoder* sama dengan nilai adc potensiometer dan nilai *error* sama dengan nol.

3.2. Perancangan Perangkat Keras

Perancangan perangkat keras merupakan skematik sistem palang parkir yang akan dibuat menggunakan komponen utama berupa motor dc dengan *encoder*, motor *driver* 1298n, potensiometer dan *board* Arduino Uno yang merupakan mikrokontroler utama sekaligus PID kontroler. Dengan bantuan *software* Fritzing skematik dapat divisualisasikan. Berikut skematik perangkat keras sistem keseluruhan pada Gambar 3. 4



Gambar 3. 2 Skematik Perangkat Keras Sistem Keseluruhan

3.3. Perancangan Perangkat Lunak

Perancangan perangkat lunak merupakan perancangan program pada *board* Arduino Uno mikrokontroler utama yang berfungsi untuk mengolah data *encoder* motor DC, mengolah nilai adc dan mengolah untuk PID kontroler. Pemrograman dilakukan menggunakan Arduino IDE untuk menulis *script* program. Program untuk sistem pengendalian motor DC menggunakan PID terdiri dari beberapa tahap dari pembacaan *encoder* motor DC sampai didapatkan nilai tanggapan dari posisi motor DC. Perancangan perangkat lunak meliputi:

- Program Pembacaan Encoder
- Program Kontroler PID
- Program Akuisisi Data

3.4. Identifikasi *Plant* Motor DC

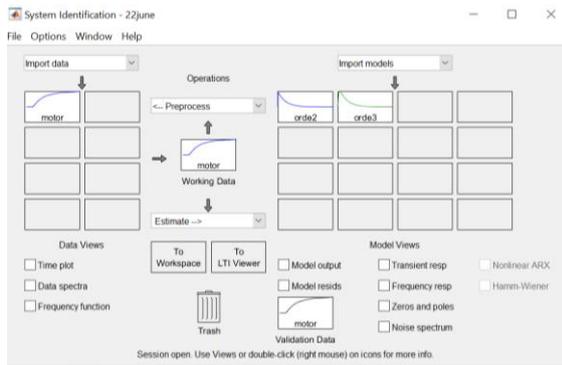
Identifikasi diperlukan untuk menentukan besaran parameter PID yang digunakan. Identifikasi parameter dapat dihitung dengan

menggunakan *System Identification Toolbox* (SIT) yaitu *tool* yang terdapat pada *software* MATLAB 2017b yang berguna untuk membuat model matematika yang didasarkan pada pengumpulan data *input/output* dari sistem. Model sistem dapat didefinisikan dengan menyesuaikan pengaturan proses model yang ingin diasumsikan sesuai dengan *input* dan *output* data dalam *System Identification Toolbox* (SIT).

Input dan *output* data untuk motor dc 25GA370 telah diakuisisi dengan menggunakan *serial monitor* Arduino IDE, dimana data keluaran (*output*) diamati sebagai kecepatan sudut (rad/s) dan data masukan (*input*) diamati sebagai tegangan pada motor. Untuk memasukan data pada SIT dengan menggunakan perintah berikut :

$$\text{motor} = \text{iddata}(\text{output}, \text{input}, 1) \quad (3.4)$$

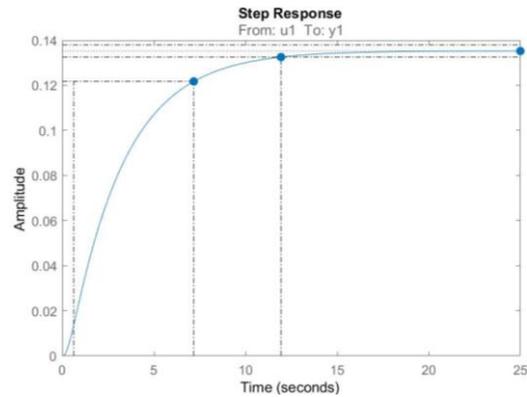
Dengan menggunakan SIT penalaan *transfer function* lebih cepat dan lebih sederhana daripada menggunakan metode konvensional. Berikut hasil dari identifikasi sistem yang telah dilakukan.



Gambar 3. 3 Hasil identifikasi plant

Dari Gambar 3. 10 menggunakan persamaan 3. 4. dari gambar tersebut menghasilkan fungsi alih *plant*. Dari data tersebut dapat mengetahui fungsi alih *plant* dan grafik. Bentuk fungsi alih yang dihasilkan dari Gambar 3. 10 dapat dilihat pada persamaan 3.5 dibawah ini

$$\frac{3.019}{s^3 + 23 s^2 + 73.75 s + 22.32} \quad (3.5)$$



Gambar 3. 4 Grafik step respon sistem fungsi alih plant

3.5. Penalaan Parameter PID

Tujuan dari penalaan ini yaitu agar mendapatkan parameter yang tepat dengan karakteristik *plant* sehingga respon keluaran dengan respon masukan yang diberikan sesuai dan *responsive*. Penalaan ini menggunakan metode kurva *S Ziegler-Nichols* dan penalaan terbagi atas 3 model identifikasi proses penalaan parameter PID yaitu model tangen, model smith dan model dua titik.

Tabel 3.1 Parameter PID

Tipe Model	Kp	Ti	Td
Model Tangent	4,41	1,24	0,31
Model Smith	1,2474	0,7	0,175
Model Dua Titik	4,96	3,32	0,83

IV.PENGUJIAN DAN ANALISA

Dalam pengujian ini dilakukan 2 macam pengujian yaitu pengujian dengan setpoint 90 derajat dan pengujian dengan setpoint 180 derajat. Setiap pengujian dilakukan 4 percobaan yang mana percobaan tersebut yaitu pengujian sebelum pengendalian PID, pengujian dengan penalaan model tangen, pengujian dengan penalaan model Smith dan pengujian dengan penalaan model dua titik. Hasil dari pengujian berupa grafik berdasarkan data yang tercatat dan juga analisa tanggapan sistem berupa *rise time*, *settling time*, *overshoot* dan *error steady*

state (ess) terhadap *setpoint*. Dalam hal ini waktu *rise time* menggunakan konstanta 0 – 90% dari waktu yang diperlukan oleh tanggapan untuk naik dan *settling time* menggunakan persentase perubahan sebesar 2%.

Dari pengujian – pengujian yang telah dilakukan dengan berbagai macam model dan *setpoint* yang diberikan disusun hasil dalam bentuk tabel berupa gambar sebagai berikut :

Tabel 4. 1 Hasil pengujian dengan *setpoint* 90 derajat

Pengujian Pada Setpoint 90 derajat									
Nama	Penalaan Parameter PID					Tr	Ts	Mo	ess
Model	Kp	Ti	Td	Ki	Kd	(s)	(s)	(%)	(%)
Tanpa Pengendalian PID	1	0	0	∞	0	0,27	0,58	31,7	25,8
Tangent	4,41	1,24	0,31	3,556	1,367	0,62	4	18,9	0
Smith	1,2474	0,7	0,175	1,782	0,218	0,65	2,78	18,9	0,92
Dua titik	4,96	3,32	0,83	1,494	4,117	0,64	6,3	7	0,46

Tabel 4. 2 Hasil pengujian dengan *setpoint* 180 derajat

Pengujian Pada Setpoint 180 derajat									
Nama	Penalaan Parameter PID					Tr	Ts	Mo	ess
Model	Kp	Ti	Td	Ki	Kd	(s)	(s)	(%)	(%)
Tanpa Pengendalian PID	1	0	0	∞	0	0,37	0,5	0	18,4
Tangent	4,41	1,24	0,31	3,556	1,367	0,7	3,5	11	0
Smith	1,2474	0,7	0,175	1,782	0,218	0,43	2,7	21,6	0,23
Dua titik	4,96	3,32	0,83	1,494	4,117	1,08	7,9	10,8	0,92

Berdasarkan pada Tabel 4.2 dan Tabel 4.3 hasil pengujian pengendalian PID terhadap sudut, didapat sebuah nilai karakteristik respon waktu yang akan menjadi acuan untuk menentukan model dan parameter yang tepat. Sehingga system pengendalian dapat sesuai dengan *setpoint* yang diberikan. Dari hasil pengujian system sebelum mengalami pengendalian memiliki *error steady state* bisa dikatakan bahwa system tersebut tidak optimal.

Hasil pengujian yang mendapatkan nilai *overshoot* lebih dari 20 %, *rise time* yang lama, waktu *settling time* yang lama dan memiliki *error steady state* lebih dari 0,5% bahwa parameter PID tersebut kurang optimal untuk mengendalikan posisi motor DC pada rentang sudut tertentu.

Berdasarkan Tabel 4.2 dan Tabel 4.3 hasil pengujian pengendalian PID motor DC didapatkan acuan karakteristik pengendalian untuk menentukan waktu *settling time* dan *error steady state* yaitu : $K_i > K_d$ untuk waktu *settling time* cepat dan *error steady state* kecil sedangkan $K_i < K_d$ untuk waktu *settling time* lambat dan *error steady state* yang besar. Selain itu juga didapatkan acuan karakteristik

pengendalian untuk menentukan besar osilasi (*overshoot*) yaitu : $K_i > K_d$ untuk menghasilkan osilasi besar dan $K_i < K_d$ untuk menghasilkan osilasi kecil. Berdasarkan pengujian yang dilakukan maka dapat disimpulkan bahwa system kendali PID dapat bekerja dengan baik sesuai harapan

V.PENUTUP

5.1. Kesimpulan

Setelah melakukan tahap perencanaan dan pengujian sistem, maka dapat diambil kesimpulan sebagai berikut:

1. Keluaran tanggapan dari sensor *encoder* motor dipengaruhi oleh parameter PID yang diberikan. Nilai keluaran pembacaan sensor sudah dalam satuan (angka) dengan menggunakan program untuk memperoleh data berasal dari *library 'PID_v1.h'*.
2. Penalaan parameter PID dengan menggunakan metode *Ziegler – Nichols* aturan pertama, menghasilkan 3 model penalaan parameter PID. Masing – masing model menghasilkan parameter

yang berbeda – beda sesuai formula masing – masing model, maka menghasilkan nilai tanggapan keluaran yang berbeda – beda.

3. Berdasarkan pengujian sistem sebelum kendali, nilai tanggapan yang dihasilkan tidak sesuai dengan *setpoint* yang diberikan, sedangkan sistem setelah kendali, nilai tanggapan yang dihasilkan sesuai dengan *setpoint* yang diberikan, maka sistem kendali bekerja sesuai yang diharapkan
4. Model penalaan parameter PID yang berbeda menghasilkan karakteristik yang berbeda beda, model tangent menghasilkan 0% *error steady state*, model smith menghasilkan waktu *settling time* yang cepat dan model dua titik menghasilkan nilai osilasi (*overshoot*) yang kecil.
5. Berdasarkan pengujian yang dilakukan didapatkan parameter yang sesuai kriteria yaitu model tangent dengan $K_p = 4,41$; $K_i = 3,556$; $K_d = 1,367$ yang menghasilkan waktu *rise time* 0,7 s, waktu *settling time* 3,5 s, *overshoot* 11 s dan *error steady state* sebesar 0% pada pengujian *setpoint* 180 derajat.

5.2. Saran

1. Penelitian selanjutnya diharapkan mampu mengembangkan aplikasi PID ini lebih baik
2. Diharapkan dapat mengimplementasikan PID ke hal yang lain

DAFTAR PUSTAKA

- Bolton, William. 2015. *Instrumentation and Control System 2nd edition*. MA: Elseiver
- Cekdin, Cekmas. 2017. *Sistem Teknik Kendali*. Yogyakarta: ANDI OFFSET
- D.J.P. Darat. Keputusan Direktur Jenderal Perhubungan Darat Nomor: 272/HK.105/DRJD/96 tentang Pedoman Teknis Penyelenggaraan Fasilitas Parkir. 1996.
- Dorf and Bishop. 2001. *Modern Control Systems 9th edition*. New Jersey: Pretince Hall
- Firmansyah, Ahmad. 2007. *Dasar – dasar Pemrograman MATLAB*. Di <https://ilmukomputer.org/wp-content/uploads/2007/08/firman-dasarmatlab.pdf> (di akses pada 30 juli 2019).
- Getting Started with MATLAB*. The MathWorks inc. 2007
- Hidayat, Pribadhi. 2013. *Sistem Pengaturan Kecepatan Putaran Motor Pada Mesin Pemutar Gerabah Menggunakan Kontroler Proporsional Integral Deferensial (PID) Berbasis Mikrokontroler*. Skripsi. Universitas Brawijaya
- Jogiyanto. 1990. *Analisis dan Disain Sistem Informasi*. Yogyakarta: ANDI OFFSET
- Lailis Syafaah, Diding Suhardi, Ilham Pakaya. 2016. *Kontrol Propotional-Integral (PI) Optimal Pada Motor Servo DC Menggunakan Algoritma Particle Swarm Optimization (PSO)*. Senaspro UMM:613-618.
- Nanang, dkk. 2009. *Pengaturan Posisi Motor Servo DC dengan Metode P, PI dan PID*.
- Ogata, Katsuhiko. 2002. *Modern Control Engineering 4th edition*. New Jersey : Pretince Hall
- Putra,Adityan Ilmawan. 2013. *Sistem Pengaturan Posisi Sudut Putar Motor DC Pada Model Rotary Parking Menggunakan Kontroler PID Berbasis Arduino Mega 2560*. Skripsi. Univeristas Brawijaya.
- Raharja,Sakti. 2016. *Rancang Bangun Prototype Palang Parkir Menggunakan Mikrokontroller ATMEGA 8535*. Jurnal. Universitas Pasir Pengaraian.

- Rahmat, Wiyono. 2018. *Pengendali Motor Servo Posisi dengan Kendali PID Berbasis Mikrokontroler Atmega*. The^{7th} URECOL Stikes Muhammadiyah Surakarta.
- Ramdani, Evan. 2015. "Parameter Identifikasi Transfer Fungsi Menggunakan MATLAB". *SETRUM – Vol 4 no. 1 (hlm 30 – 36)*. Cilegon: Universitas Sultan Ageng Tirtayasa Cilegon.
- Sianipar, Rismon Hasiholan. 2018. *Dasar Sistem Kontrol dengan MATLAB*. Yogyakarta: ANDI OFFSET
- Siswanto, Bambang. 2010. *Pengaturan Posisi Motor DC Menggunakan Kontroller PID Berbasis Genetik Algoritma (GA)*. Tesis. Institut Teknologi Bandung.
- Sumiati,Ruzita. 2009. *Kendali Motor DC Menggunakan Mikrokontroller ATMEGA 8535 Dengan Kontrol PID Metode Ziegler-Nichols*. Tesis. Institut Teknologi Bandung
- Triwiyatno, Aris. 2015. *BUKU AJAR SISTEM KONTROL ANALOG*. Universitas Indonesia