

Perancangan Alat Pelatihan PID Analog Untuk Kecepatan Kecepatan Motor DC

1st Raihan Noval Lusamba

Direktorat Kampus Purwokerto

Universitas Telkom Purwokerto

Purwokerto, Indonesia

raihannovalusamba@student.telkomuniversity.ac.id

2nd Yulian Zetta Maulana, S.T., M.T

Direktorat Kampus Purwokerto

Universitas Telkom Purwokerto

Purwokerto, Indonesia

yulianm@telkomuniversity.ac.id

3rd Gunawan Wibisono, S.T., M.T

Direktorat Kampus Purwokerto

Universitas Telkom Purwokerto

Purwokerto, Indonesia

gwibisono@telkomuniversity.ac.id

Abstrak — Pengendalian kecepatan motor DC merupakan salah satu aspek penting dalam sistem pengendalian industri. Salah satu metode yang sering digunakan adalah pengontrol PID yang mampu memberikan kinerja sistem yang stabil dan presisi. Untuk meningkatkan pemahaman dan keterampilan praktis dalam penerapan PID, penelitian ini bertujuan untuk merancang alat pelatihan PID analog yang dirancang khusus untuk pengendalian kecepatan motor DC. Sistem ini mencakup perangkat keras berupa rangkaian analog PID serta perangkat lunak untuk memantau dan menganalisis kinerja sistem. Prototipe yang dikembangkan memanfaatkan potensiometer sebagai pengontrol parameter PID, motor DC sebagai aktuator, dan sensor kecepatan untuk memberikan umpan balik. Hasil pengujian menunjukkan bahwa alat ini mampu mewakili prinsip kerja PID dengan baik dan memberikan platform yang baik untuk mempelajari dan bereksperimen dengan pengendalian motor DC. Metode *Ziegler Nichols* mendapatkan hasil dari penalaan kontroler PID dengan parameter optimal $K_p = 2,475$, $K_i = 0,45$, dan $K_d = 1$. Analisis respon mendapatkan $rise\ time = 1,6584\ s$, $settling\ time = 14,6048\ s$, $overshoot = 17,6901\%$, dan $error\ steady\ state = 0,0395$. Sistem Dari hasil ini, PID mampu memberikan respon yang cepat dan stabil dengan tingkat akurasi yang baik

Kata kunci— PID analog, pengendalian kecepatan, motor DC, alat pelatihan, sistem kontrol.

I. PENDAHULUAN

Di era modern ini motor DC sangat dibutuhkan oleh kalangan industri. Motor DC juga sangat penting dalam dunia industri. Motor DC bermacam-macam karena motor DC menggunakan sumber listrik DC, oleh karena itu dalam dunia industri sering kali menggunakan penyearah untuk mengubah sumber listrik AC menjadi DC. Motor DC banyak digunakan karena kemampuannya dalam mengontrol kecepatan dan arah putaran dengan mudah [1].

Penggunaan motor DC memerlukan jenis arus DC atau arus searah. Motor DC biasanya menggunakan arus searah, namun energi mekanik berupa putaran atau gerak nantinya akan diubah. Motor listrik ada dua jenis yaitu motor AC dan DC. Perbedaan motor AC dan DC dilihat dari arus yang digunakan [2]. Motor DC sangatlah kompleks sehingga ketika motor DC bekerja pada suatu proses yang memerlukan kecepatan konstan maka sistem motor DC dapat terganggu. Penggunaan kendali dalam dunia industri saat ini sangat bermanfaat untuk diperhatikan terutama dalam mengatur

kecepatan motor DC. Memiliki respon yang cepat dan akurat serta tahan terhadap gangguan merupakan sistem pengendalian yang baik [3].

Dalam sistemnya, pengendalian kecepatan putaran motor DC juga dilakukan dengan metode *Pulse width Modulation* (PWM). PWM ini untuk memanipulasi lebar pulsa pada satu waktu. *Input* daya ke motor dapat dikontrol menggunakan variasi tegangan DC yang diterapkan ke terminal motor. Oleh karena itu diperlukannya sistem pada pengendalian yaitu dengan menggunakan PID [4].

Pelatihan alat ini bertujuan untuk mempelajari dan memahami cara pengendalian suatu sistem PID karena pada saat melakukan pelatihan terkadang alat pelatihan sangat terbatas. Dan pengontrol sistem PID juga dapat memahami pengendalian otomatis dalam pengendalian motor. parameter PID yang mudah diatur untuk setiap parameter (*Proportional, Integral, Derivative*). Pengendali PID yang digunakan pada alat pelatihan ini menggunakan analog [5].

II. KAJIAN TEORI

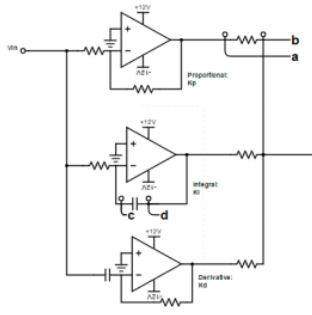
A. Motor DC

Motor DC adalah jenis motor listrik yang sering digunakan di berbagai bidang industri. Motor DC adalah perangkat elektromagnetik yang mengubah energi listrik menjadi energi mekanik, atau gerak. Motor DC menerima daya input dari sumber listrik dalam bentuk tegangan dan arus. Output motor DC adalah daya mekanik, yang dinyatakan dalam torsi dan kecepatan rotor. Penyearah memiliki karakteristik yang bervariasi, dan karakteristik ini memengaruhi tegangan input motor DC, sehingga memengaruhi bentuk gelombang arus serta torsi dan kecepatan rotor yang dihasilkan [6].

B. (*Proportional, Integral, Derivative*) PID

PID merupakan suatu teknik kendali yang digunakan dalam suatu kendali, terdapat tiga jenis kendali PID yang selalu digabungkan (*Proportional, Integral, dan Derivative*) parameter jenis ini dapat menentukan kualitas respon suatu kendali. Dalam menggunakan teknik kendali PID terdapat berbagai macam cara, salah satunya adalah membuat sistem kontrol kecepatan yang menghasilkan kontrol PID sesuai spesifikasi yang dibutuhkan dan tetap stabil. Apabila menggunakan kendali, PID dilakukan secara bersama-sama

atau terpisah karena masing-masing kendali mempunyai keunggulan tersendiri yaitu kendali *proportional* dapat mempercepat *rise time*, kendali integral dapat mengurangi kesalahan dan kendali *devirative* dapat mengurangi *overshoot* [7].



GAMBAR 1.
RANGKAIAN PID ANALOG PADA RANGKAIAN

Kontrol kendali PID merupakan metode yang digunakan untuk mengatur suatu proses atau sistem agar mencapai dan mempertahankan nilai target tertentu secara stabil dan akurat. Pada rangkaian analog, kontrol PID diimplementasikan menggunakan komponen elektronik seperti op-amp, resistor, dan kapasitor yang dapat diatur untuk menghasilkan sinyal kontrol yang sesuai. (P): Bagian ini menghasilkan sinyal yang berbanding lurus dengan *error* (selisih antara nilai *set point* dengan nilai proses). (I): Bagian ini bertugas menghitung akumulasi *error* dari waktu ke waktu, sehingga dapat mengoreksi *error* yang tersisa (*error steady state*). (D): Bagian ini merespon perubahan *error* yang cepat dengan menghitung turunan *error* dari waktu ke waktu dalam rangkaian analog [8].

C. Pulse Width Modulation (PWM)

Pulse Width Modulation (PWM) adalah suatu teknik dalam sistem kontrol untuk memanipulasi lebar sinyal yang dinyatakan dalam pulsa dalam satu periode. Pada sistem kendalinya dengan mengubah perbandingan lebar pulsa positif ke lebar pulsa negatif atau sebaliknya pada frekuensi sinyal tetap. PWM sering digunakan untuk mengimplementasikan modulasi data untuk telekomunikasi, pengontrol daya, regulator atau tegangan. Termasuk implementasinya untuk mengatur kecepatan motor DC pada sistem kendali otomatis [9].

D. Op-Amp

Operational Amplifier adalah IC linier digunakan untuk memperkuat sinyal listrik. Op-Amp terdiri dari beberapa Transistor, Dioda, Resistor dan Kapasitor terintegrasi yang dapat menghasilkan gain tinggi pada rentang frekuensi yang luas. Jumlah Op-Amp dalam satu paket IC dapat dibagi menjadi *Single* Op-Amp, Op-Amp Ganda dan Op-Amp *Quad*. Op-Amp mempunyai dua masukan (*input*) dan keluaran (*output*) yaitu terdapat *inverting* dan *non inverting* serta hanya mempunyai satu keluaran (*output*). Op-Amp juga mempunyai dua sambungan catu daya, yaitu catu daya positif dan catu daya negatif. Dan bentuk simbol op-amp adalah segitiga dengan jalur *input*, *output* dan catu daya [10].

E. Mikrokontroler

Arduino Uno merupakan salah satu mikrokontroler yang paling banyak digunakan dalam berbagai aplikasi elektronik dan robotika. Arduino Uno berbasis pada mikrokontroler ATmega328P yang menggunakan arsitektur RISC yang

memungkinkan proses eksekusi instruksi dalam satu siklus *clock* dan mampu mencapai kecepatan hingga 16 MHz. Arduino Uno didesain dengan 14 pin Digital I/O yang dapat digunakan sebagai input maupun *output*, termasuk 6 pin PWM (*Pulse Width Modulation*) yang sangat penting untuk mengatur kecepatan motor atau pengendalian level *output*. Selain itu, Arduino Uno dilengkapi dengan 6 pin *input* analog berukuran 10 bit yang digunakan untuk membaca sensor analog, serta *port* serial untuk komunikasi dengan komputer atau modul lainnya [11].

F. Sensor Optocoupler

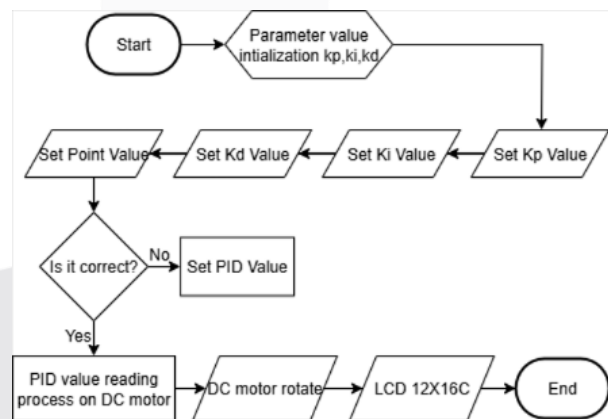
Sensor *Optocoupler* bertujuan untuk mengukur kecepatan putar motor DC non-kontak. Sensor ini bekerja dengan memanfaatkan prinsip pengukuran jarak dan pantulan sinar inframerah yang dipantulkan oleh permukaan suatu objek, dalam hal ini rotor motor yang dilengkapi dengan elemen reflektif atau fisik tertentu. Sensor *Optocoupler* mampu mendeteksi perubahan posisi atau jarak relatif terhadap objek yang dipantau dan mengubahnya menjadi sinyal elektronik [12].

III. METODE

Penelitian ini dilaksanakan dengan alur yang telah ditetapkan sebelumnya. Hal ini dilaksanakan guna mendapatkan hasil yang akurat dan tepat.

A. Flowchart Sistem

Skema sistem pada penelitian ini ditunjukkan pada gambar 2 yang menunjukkan bagaimana proses bekerja dari awal memasukkan nilai parameter PID hingga sistem mulai berjalan berdasarkan nilai PID yang diberikan.



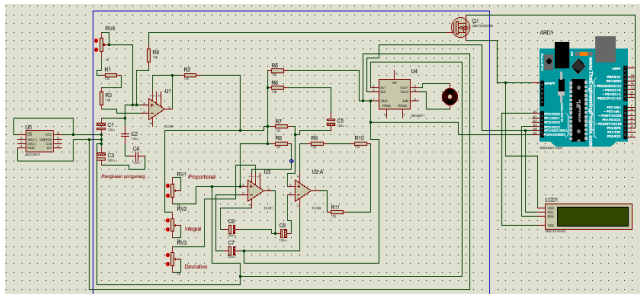
GAMBAR 2.
FLOWCHART SISTEM

Pada gambar diatas menunjukkan tentang alur sistem berjalan, proses awal mengenai inisialisasi nilai parameter PID dan sistem akan berlanjut dengan memasukkan nilai parameter pada PID (K_p , K_i , dan K_d) dan pada nilai *setpoint* sebagai suatu pergerakan pada motor DC. Sistem yang berjalan akan memastikan nilai parameter PID yang akan dimasukkan sesuai dengan nilai parameter yang akan diuji, nilai parameter yang sudah dimasukkan pada sistem maka akan diproses pada op-amp untuk memutar pada motor DC. Motor DC akan berputar berdasarkan nilai parameter PID yang diproses pada sistem. Kemudian output akan ditampilkan pada LCD dan putaran motor DC.

B. Rangkaian Skematik Sistem

Perancangan skematik sistem akan dirancang pada setiap komponen yang digunakan dalam penelitian ini. Setelah

proses pengujian setiap komponen, tentunya akan menghubungkan semua komponen yang telah diuji untuk menjadikan satu kesatuan sistem dalam penelitian ini. Rancangan skematik pada gambar 3 menjelaskan hubungan antar komponen dalam satu sistem.



GAMBAR 3.
RANGKAIAN SKEMATIK SISTEM

Gambar diatas menunjukkan rangkaian PID analog yang terdiri dari empat potensiometer: satu untuk *setpoint* dan tiga untuk pengaturan parameter PID (*proportional, integral, derivative*). Rangkaian ini menggunakan IC TL081 dan TL084, serta dilengkapi resistor, kapasitor, motor driver, dan motor DC. Sistem ini dapat beroperasi dalam kondisi *open loop* atau *close loop*. Gambar 3 menunjukkan tambahan mikrokontroler Arduino Uno, sensor *optocoupler*, dan LCD 16x2 I2C. Arduino hanya berfungsi membaca putaran motor DC melalui sensor *optocoupler* dan menampilkannya dalam bentuk RPM pada LCD. Jika sinyal keluaran sensor *optocoupler* disambungkan ke rangkaian pengurang, sistem menjadi *close loop*. Jika tidak disambungkan, sistem bekerja dalam kondisi *open loop*. Arduino tidak digunakan untuk mengontrol motor.

C. Pengujian Sistem Tanpa Pengendali

Pada pengujian kali ini dilakukan tidak ada umpan balik pada sistem, sistem pada pengujian ini hanya diberikan *input* tegangan langsung dan dipantau pada hasil respons yang didapatkan nilai *rise time* dan nilai *error steady state*.

D. Pengujian Sistem Dengan Pengendali

Pengujian pada pengendali PID untuk memastikan bahwa pengendali dapat mengatur kecepatan motor DC dengan optimal. Pengujian ini dilakukan untuk mengetahui efektifitas pada penggunaan pengendalian PID untuk mengatur kecepatan motor DC. Putaran kecepatan motor DC akan menyesuaikan dari nilai yang dimasukkan untuk mencapai nilai *setpoint* yang diatur untuk mencapai target. Pengujian ini dilakukan 2 pengujian yaitu dengan menggunakan pengendali dan tanpa pengendali. Pengujian dengan menggunakan Pengendali ini dilakukan dengan membandingkan metode *Ziegler Nichols* dan *trial error*. Pada metode *Ziegler Nichols* dengan mencari nilai parameter PID (K_p, K_i, K_d).

E. Analisis Tanggapan Waktu

Pada pengujian ini dengan menganalisa tanggapan waktu ini dilakukan untuk performa sistem pengendali PID pada pengaturan kecepatan motor DC dengan menggunakan metode *Ziegler Nichols*. Nilai parameter yang digunakan pada Analisa tanggapan waktu yaitu *time rise*, *settling*, *overshoot*, dan *error steady state*. *Time rise* melakukan pengujian dengan melihat respon awal sistem saat mencapai *setpoint* awal, *time settling* melakukan pengujian mengamati waktu sistem untuk masuk dan tetap berada pada rentang toleransi 2% atau 5% dari suatu *setpoint* awal, *Overshoot* melakukan pengujian dengan mengamati respon sistem saat melebihi batas dari suatu *setpoint* dan *error steady state*

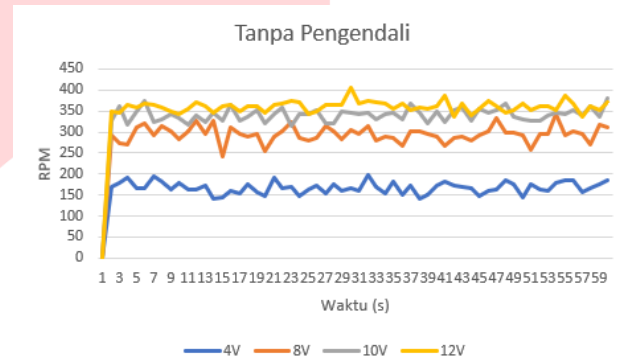
melakukan pengujian dengan mengamati selisih dari nilai *output* dengan nilai *setpoint* saat sistem yang keluar nilai sudah stabil.

IV. HASIL DAN PEMBAHASAN

Proses penelitian dilakukan dengan menguji performa dan analisa hasil yang didapatkan pada motor DC. Parameter yang digunakan pada pengujian ini adalah tanpa kontroler, menggunakan kontroler (*trial error*), dan menggunakan kontroler metode *Ziegler Nichols*. Hasil parameter yang diperoleh berupa grafik tuning dan analisa respon sistem (*time rise*, *settling time*, *overshoot*, *error steady state*) motor DC yang diperoleh dengan menggunakan Matlab.

A. Sistem Tanpa Pengendali

Pengujian pertama dilakukan dengan memberikan tegangan yang bervariasi ke motor DC dan mencatat nilai RPM yang dihasilkan.



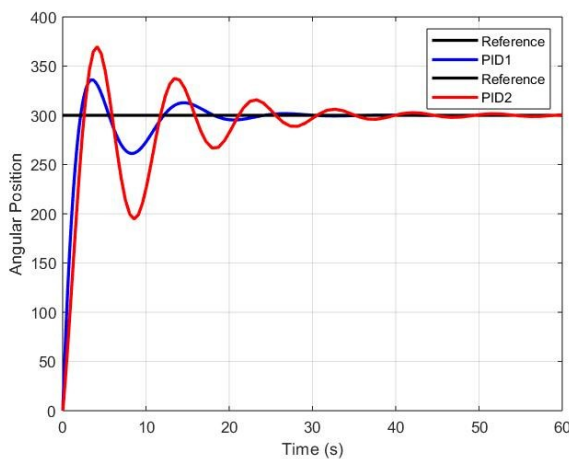
GAMBAR 4.
SISTEM TANPA PENGENDALI

Grafik tersebut menunjukkan hubungan antara waktu dan kecepatan putar motor DC tanpa sistem pengendali pada empat level tegangan: 4V, 8V, 10V, dan 12V. Sumbu waktu berada pada garis horizontal (dalam detik), sedangkan sumbu vertikal menunjukkan kecepatan motor dalam RPM. Pada tegangan 4V, kecepatan motor rendah dan cukup stabil, sekitar 150–200 RPM. Saat tegangan dinaikkan ke 8V, kecepatan motor meningkat ke kisaran 250–300 RPM, namun mulai terlihat fluktuasi. Tegangan 10V menghasilkan kecepatan 300–350 RPM dengan fluktuasi yang lebih jelas, menandakan kurangnya kestabilan. Pada tegangan 12V, kecepatan motor mendekati 400 RPM, tetapi grafik menunjukkan fluktuasi yang sangat besar. Hal ini menunjukkan bahwa semakin tinggi tegangan, semakin tinggi kecepatan motor, namun tanpa sistem kendali, motor tidak mampu menjaga kestabilan putaran, terutama pada tegangan tinggi.

B. Sistem Dengan Pengendali PID (Metode *Trial error*)

Pada pengujian dengan pengendali sistem menggunakan pengendali metode *trial error*. Percobaan metode ini dilakukan dengan memasukkan nilai parameter (K_p, K_i, K_d) dengan nilai yang bebas dan tanpa aturan dalam pemberian nilainya. Hasil yang diperoleh berupa penalaan sistem dan juga nilai dari respon sistem. Pengujian pada sistem ini dilakukan dari tiap nilai parameter (K_p, K_i, K_d) yang berbeda, pengujian ini dilakukan untuk mengetahui adanya kesalahan

nilai RPM yang didapatkan. Pada pengujian kali ini dilakukan sebanyak 2 kali..



GAMBAR 5.

SISTEM DENGAN PENGENDALI PID (METODE TRIAL ERROR)

Gambar 5 di atas merupakan grafik respon sistem terhadap input (*setpoint*) berupa kurva posisi sudut terhadap waktu selama 60 detik. Grafik ini digunakan untuk membandingkan kinerja dua buah kontroler PID yang berbeda, yaitu percobaan awal PID 1 dan percobaan kedua PID 2, dalam pengaturan sistem agar mengikuti nilai acuan sebesar 300 satuan. Kurva acuan (hitam) menunjukkan target atau nilai *setpoint* yang harus dicapai oleh sistem. Kurva biru menunjukkan respon sistem dengan kontroler PID 1, sedangkan kurva merah menunjukkan respon sistem dengan kontroler PID 2.

TABEL 1

ANALISIS RESPON SISTEM PERCOBAAN 1 DAN PERVOBAAN 2

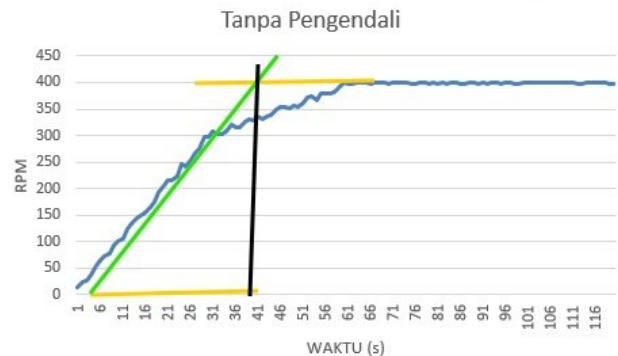
PID Controller

	Time rise (s)	Time settling (s)	Overshoot (%)	Error steady state (%)
PID 1	1,6009	16,7861	12,0015	0,00051
PID 2	1,9662	32,8449	23,2559	0,1926

Tabel 1 di atas menunjukkan perbandingan kinerja dua buah kontroler PID berdasarkan beberapa parameter kontrol sistem. Parameter yang dibandingkan meliputi *rise time*, *settling time*, *overshoot*, dan *steady-state error*. Dari tabel tersebut terlihat bahwa PID 1 memiliki *rise time* sebesar 1,6009 detik yang berarti bahwa sistem yang dikontrol oleh PID 1 dapat merespon perubahan lebih cepat dibandingkan dengan PID 2 yang memiliki *rise time* lebih lama, yaitu 1,9662 detik. Hal ini menunjukkan bahwa PID 1 lebih respon dalam mencapai nilai target awal. Untuk *settling time*, PID 1 juga menunjukkan kinerja yang lebih baik dengan waktu untuk mencapai kestabilan sebesar 16,7861 detik, sedangkan PID 2 membutuhkan waktu yang jauh lebih lama, yaitu 32,8449 detik. Hal ini menunjukkan bahwa sistem dengan PID 1 dapat mencapai keadaan stabil lebih cepat. Dari segi *overshoot* yang melampaui nilai target, PID 1 juga lebih baik dengan nilai sebesar 12,0015%, dibandingkan dengan PID 2 yang memiliki *overshoot* sebesar 23,2559%. Terakhir, pada parameter *error steady state* PID 1 kembali menunjukkan kinerja yang jauh lebih baik dengan nilai *error* yang sangat kecil yaitu 0,00051%, sedangkan PID 2 memiliki *error* sebesar 0,1926%. Artinya, PID 1 jauh lebih akurat dalam mencapai nilai akhir yang diinginkan.

C. Sistem Dengan Pengendali PID (Metode Ziegler Nichols Open loop)

Gambar 6 menunjukkan respon yang didapatkan oleh motor DC ketika diberi tegangan 12V. Hal ini bertujuan untuk mencari nilai T dan L yang dimiliki oleh motor DC ketika diberikan tegangan maksimum dalam bentuk kurva S.



GAMBAR 6.

KURVA S SISTEM

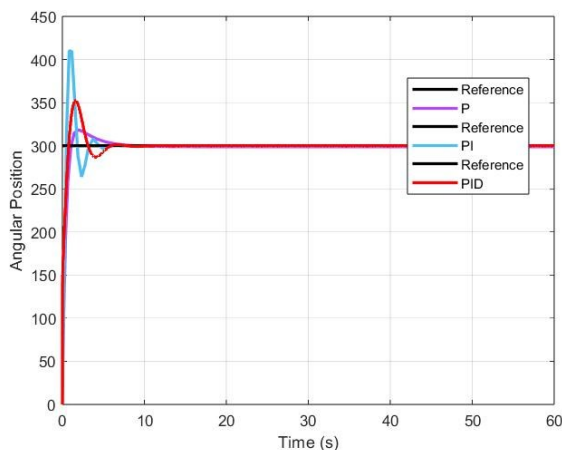
Pada gambar 6 menampilkan kurva respon sistem saat motor DC diberikan tegangan 12V, Hasil dari pembacaan motor DC dibaca oleh sensor *optocoupler* untuk mendeteksi kecepatan motor. Hasil kurva tersebut bisa mencari L dan T untuk menentukan nilai parameter PID metode Ziegler Nichols. L merupakan waktu yang dibutuhkan sistem untuk merespon Ketika diberikan tegangan sedangkan T merupakan waktu yang dibutuhkan Ketika sistem mulai mencapai kondisi stabil. L yang didapatkan sebesar 5 detik dan T yang didapatkan sebesar 37,1 detik. Nilai T dan L akan diimplementasikan kedalam nilai parameter PID *open loop*. Berikut perhitungan nilai parameter PID dengan metode Ziegler Nichols pada tabel 2.

TABEL 2

PARAMETER PID OPEN LOOP

Parameter	Kp	Ki	Kd
P	7,4	∞	0
PI	6,29	17,85	0
PID	8,14	11	2,75

Pada tabel diatas didapat pengendali P (*Proportional*), gain (Kp) dihitung dengan membagi T dengan L, didapatkannya adalah sebesar 7,4. Dan nilai Ti dan Td ditetapkan menjadi nol. Untuk pengendali I (*Integral*), nilai Kp adalah 0,85 kali T/L, yaitu sekitar 6,29. Nilai Ti dihitung dengan membagi L dengan 0,28, yaitu sekitar 17,8571, sedangkan Td tetap nol karena pengendali ini tidak menggunakan *derivative*. Untuk pengendali D, nilai Kp adalah 1,1 kali T/L, yaitu 8,14. Sementara itu, Ti diambil dengan mengalikan 2,2 dengan L yaitu 11, dan Td adalah 0,55 kali L, yaitu 2,75.



GAMBAR 7.
GRAFIK SISTEM DENGAN PENGENDALIAN PID METODE
ZIEGLER NICHOLS OPEN LOOP

Pada gambar 7 menampilkan respon hasil *software* yang diperoleh sistem dengan pengendali PID dengan keadaan open loop pada metode *Ziegler Nichols*, terdapat 3 parameter PID yang dilakukan pengujian P, PI, PID. Berikut nilai parameter PID yang digunakan pada saat sistem *close loop* :

- A. Parameter P , $K_p = 4,5$, $K_i = 0$, $K_d = 0$
- B. Parameter PI, $K_p = 1,7$, $K_i = 8$, $K_d = 0$
- C. Parameter PID, $K_p = 2,475$, $K_i = 0,45$, $K_d = 1$

TABEL 3

ANALISIS RESPON SISTEM OPEN LOOP

PID Controller				
	Time rise (s)	Time settling (s)	Overshoot (%)	Error steady state (%)
P	0,7211	4,7851	6,1470	1,2162
PI	0,3828	3,9313	36,7214	0,0039
PID	0,5696	5,1030	17,425	0,0590

Pengendali P memiliki *rise time* sebesar 0,7211 detik dan *settling time* sebesar 4,7851 detik. Nilai *overshoot*nya tergolong kecil, yakni sebesar 6,1470%, tetapi masih memiliki *error steady state* sebesar 1,2162%. Pada pengendali PI, *rise time* ini lebih cepat, yakni sebesar 0,3828 detik, dan *settling time* juga lebih baik, yakni sebesar 3,9313 detik. Akan tetapi, *overshoot*nya meningkat hingga 36,7214%, yang menunjukkan adanya lonjakan yang cukup tinggi sebelum mencapai nilai yang diinginkan. Meskipun demikian, *error steady state* sangat kecil, hanya sebesar 0,0039%, yang menunjukkan bahwa sistem sangat akurat dalam mencapai nilai akhirnya. Sementara itu, pengendali PID menunjukkan adanya keseimbangan antara keduanya. *Rise time* nya sebesar 0,5696 detik dan *settling time* nya sebesar 5,1030 detik. Nilai *overshoot* yaitu sebesar 17,425%, dan *error steady state* juga cukup kecil, yakni sebesar 0,0590%. Hal ini menunjukkan bahwa PID memberikan respon yang cukup cepat, akurat, dan tidak memiliki terlalu banyak lonjakan dibandingkan dengan PI.

Sistem dengan *open loop* akan menampilkan respon terhadap motor masukan tertentu seperti tegangan dan tanpa adanya kontrol *feedback*. Respon yang didapatkan oleh *open loop* akan mendapatkan masalah seperti adanya osilasi dan tidak kestabilan pada sistem. Dengan menggunakan model matematika pada sistem ini tentunya bisa memprediksi respon pada motor yang didapatkan, sebelum menerapkan sistem *close loop* yang meminimalkan osilasi dan mencapai

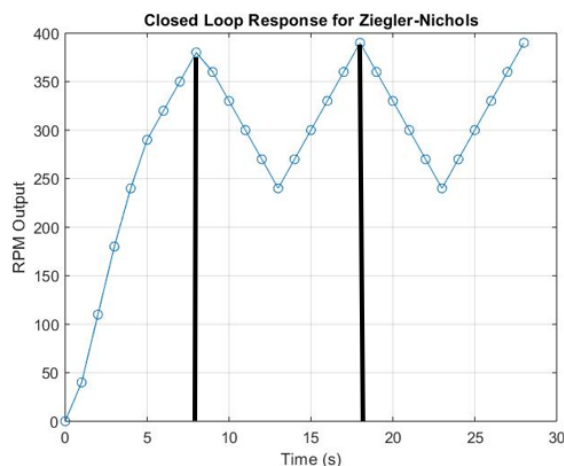
kestabilan yang cepat. Terdapat model matematika yang digunakan pada sistem *open loop*, berikut persamaan matematika yang digunakan pada persamaan 1 :

$$\frac{0.3436s + 0.1313}{s^2 + 0.1092s + 0.003955} \quad (1)$$

model matematika dapat mempresentasikan keakuratan yang cukup tinggi dalam sistem nyata. Model matematika yang didapatkan akan diimplementasikan kedalam fungsi *transfer function* dengan menggunakan Simulink di MATLAB.

D. Sistem Dengan Pengendali PID (Metode *Ziegler Nichols Close loop*)

Dalam penerapan metode *Ziegler-Nichols* pada sistem *close loop*, dibutuhkan dua parameter utama yaitu K_u (nilai penguatan maksimum) dan T_u (periode osilasi). K_u diperoleh dengan menaikkan nilai K_p secara bertahap, sementara parameter K_i dan K_d disetel ke nol. Tujuannya adalah untuk mendapatkan kondisi di mana sistem menunjukkan osilasi yang stabil dan konstan. Ketika sistem (misalnya motor DC) menunjukkan pola osilasi yang tidak meredam ataupun membesar, nilai K_p saat itu dianggap sebagai K_u . Sedangkan nilai T_u diperoleh dari pengukuran waktu antara dua puncak gelombang osilasi yang terjadi. Nilai K_u dan T_u inilah yang kemudian digunakan untuk menghitung konstanta PID agar sistem memiliki respon yang lebih cepat dan stabil. Pada penelitian ini, nilai K_u dan T_u diperoleh melalui gambar 8 yang menunjukkan bahwa sistem berosilasi stabil.



GAMBAR 8
SISTEM OSILASI STABIL

Pada gambar 8 menunjukkan bahwa sistem mampu melakukan respon yang berosilasi secara stabil. Respon ini mendapatkan nilai $K_p = 4,5$ sehingga didapatkan nilai K_u pada sistem sebesar 4,5 dan waktu titik temu sebesar 10 detik, sehingga nilai pada $T_u = 10$. Nilai K_u dan T_u ini yang akan diimplementasikan untuk mencari nilai parameter PID metode *Ziegler Nichols* dengan sistem *close loop*. Terdapat nilai parameter yang didapatkan pada tabel 4.

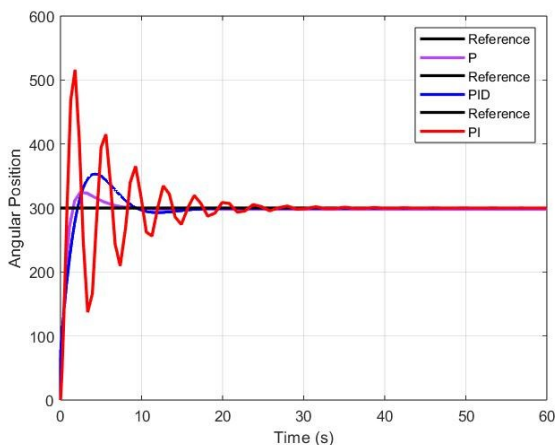
TABEL 4

PARAMETER PID CLOSE LOOP

Parameter	K_p	K_i	K_d
P	4,5	∞	0
PI	1,7	8	0
PID	2,475	0,45	1

Pada pengendali P, nilai K_p dihitung dari $0,4 \times K_u$. Jika nilai K_u adalah 11,25 maka K_p menjadi 4,5. Parameter T_i

bernilai tak hingga (∞), artinya tidak digunakan, dan Td bernilai 0, yang menunjukkan tidak ada aksi *integral*. Untuk PI, nilai Kp dihitung $0,38 \times Ku$, hasilnya adalah 1,7. Nilai Ti diambil dari $0,8 \times Pu$, yaitu sebesar 8, dan Td tetap 0 karena pengontrol ini tidak menggunakan *derivative*. Dan pada pengontrol PID, Kp diperoleh dari $0,55 \times Ku$, yaitu 2,475. Nilai Ti adalah $0,45 \times Pu$ (hasilnya 0,45) dan Td adalah $0,1 \times Pu$, yaitu 1. Pengendalian ini menggunakan aksi ketiga pengendalian secara bersamaan *proportional*, *integral*, dan *devirative* sehingga memberikan pengendalian yang lebih lengkap dan seimbang terhadap sistem. Hasil pengujian yang diperoleh dari ketiga parameter PID metode *close loop* Ziegler Nichols terdapat pada gambar 9.



GAMBAR 9.
GRAFIK SISTEM DENGAN PENGENDALIAN PID METODE
ZIEGLER NICHOLS CLOSE LOOP

Gambar 9 menunjukkan hasil grafis yang diperoleh ketika nilai parameter P, parameter PI, dan parameter PID dijalankan. Kurva parameter P direpresentasikan dengan warna biru, kurva parameter PI direpresentasikan dengan warna hijau, dan kurva parameter PID direpresentasikan dengan warna hitam. Hasil grafik yang diperoleh menunjukkan bahwa parameter P tidak menunjukkan osilasi pada sistem, sedangkan parameter PI dan PID menunjukkan osilasi pada saat motor DC pertama kali dijalankan. Hasil analisis respon sistem yang diperoleh dari hasil grafik terdapat pada tabel 5.

TABEL 5.
RESPON ANALISIS SISTEM CLOSE LOOP

PID Controller	Time rise (s)	Time settling (s)	Overshoot (%)	Error steady state (%)
P	1,0415	6,1646	8,0410	1,9948
PI	0,6062	22,0710	71,9772	0,0065
PID	1,6584	14,6048	17,6901	0,0395

Pada tabel diatas menunjukan pengendali P memiliki *time rise* 1,0415 detik, dengan *settling time* 6,1646 detik. Nilai *overshoot*-nya adalah 8,0410%, dan *error steady state* tercatat sebesar 1,9948%. Artinya, *overshoot*-nya kecil, *error steady state* masih cukup tinggi. Pengendali PI menunjukkan *rise time* yang lebih cepat, yaitu 0,6062 detik. Namun, waktu untuk mencapai kestabilan sangat lama, yaitu *settling time* nya 22,0710 detik, dan *overshoot*-nya sangat tinggi, yaitu 71,9772%. Namun, *error steady-state*-nya sangat kecil, hanya 0,0065%, yang menunjukkan bahwa

sistem dengan PI mampu mencapai nilai akhir yang sangat dekat dengan *setpoint*. Sementara itu, pengendali PID menghasilkan *rise time* paling lambat, yaitu 1,6584 detik, dan *settling time*-nya 14,6048 detik. *Overshoot*-nya 17,6901%, dan *error steady state* cukup rendah, yaitu 0,0395%. Hal ini menunjukkan bahwa PID memberikan keseimbangan antara akurasi akhir dan respons sistem, meskipun waktu responsnya tidak secepat PI.

V. KESIMPULAN

. Pengujian dengan metode penalaan menunjukan bahwa metode Ziegler Nichols lebih baik atau unggul dibandingkan dengan metode *trial error* dan sistem dengan tanpa pengendali. Parameter yang didapat Kp = 2,475, Ki = 0,45, dan Kd = 1. Analisis respon mendapatkan *rise time* = 1,6584 s, *settling time* = 14,6048 s, *overshoot* = 17,6901%, dan *error steady state* = 0,0395. Sistem Dari hasil ini, PID mampu memberikan respon yang cepat dan stabil dengan tingkat akurasi yang baik. Alat pelatihan PID analog yang dirancang berhasil mengendalikan kecepatan motor DC secara stabil dengan respon yang cukup cepat (*rise time* : *trial error* 1,6 detik, Ziegler Nichols open loop 0,39 detik dan Ziegler Nichols close loop 1,6 detik) serta mencapai kestabilan sesuai yang diinginkan (*settling time* : *trial error* 16,8 detik, Ziegler Nichols open loop 3,9 detik dan Ziegler Nichols close loop 14,6 detik).2

REFERENSI

- [1] R. A. B. S. Putra, A. R. Al Tahtawi, and K. Wijayanto, "Pengendalian Kecepatan Motor DC Menggunakan Metode Fuzzy Integral Controller," Pros. Ind. Res. Work. Natl. Semin., vol. 12, pp. 52–56, 2021, doi: 10.35313/irwns.v12i0.2657.
- [2] N. E. Anwar and H. Ferdilla, "Pengaturan Kecepatan Dan Pengendalian Motor DC 5 V-110 V Menggunakan IC Tipe NE 555," J. Teknol. Ris. Terap., vol. 1, no. 2, pp. 113–123, 2023, [Online]. Available: <https://doi.org/10.35912/jatra.v1i2.31572657.6>
- [3] N. Soedjarwanto, "Pengendalian kecepatan motor DC menggunakan buck-boost converter berbasis IoT," J. Inform. dan Tek. Elektro Terap., vol. 11, no. 3s1, pp. 943–950, 2023, doi: 10.23960/jitet.v11i3s1.3399.
- [4] S. Pokhrel, "Pengendalian Kecepatan Motor Dc Menggunakan PWM (Pulse Width Modulation)," Αγαη, vol. 15, no. 1, pp. 37–48, 2024.
- [5] N. Alamsyah, R. Muhyaddin, and A. Darmawansyah, "Media Pembelajaran Pada Mata Kuliah Robotika," JTEK J. Teknol. Komput., vol. 2, no. 2, pp. 190–195, 2022.
- [6] I. Hudati, A. P. Aji, and S. Nurrahma, "Kendali Posisi Motor DC dengan Menggunakan Kendali PID," J. List. Instrumentasi dan Elektron. Terap., vol. 2, no. 2, pp. 1–6, 2021, doi: 10.22146/juliet.v2i2.71148.
- [7] M. Irhas, I. Iftitah, and S. A. Azizah Ilham, "Penggunaan Kontrol Pid Dengan Berbagai Metode Untuk Analisis Pengaturan Kecepatan Motor Dc," JFT J. Fis. dan Ter., vol. 7, no. 1, p. 78, 2020, doi: 10.24252/jft.v7i1.13846.
- [8] A. S. Bhandari, A. Chaudhuri, S. Roy, S. Negi, and M. Sharad, "Single chip self-tunable N-input N-output PID

control system with integrated analog front-end for miniature robotics,” Proc. 2017 IEEE 14th Int. Conf. Networking, Sens. Control. ICNSC 2017, pp. 109–114, 2017, doi: 10.1109/ICNSC.2017.8000076.

[9] A. Pranata, “Automatic Scroll Saw System Dengan Teknik Kendali Kecepatan Pulse Width Modulation (PWM) Berbasis Arduino UNO,” J-SISKO TECH (Jurnal Teknol. Sist. Inf. dan Sist. Komput. TGD), vol. 4, no. 1, p. 69, 2021, doi: 10.53513/jsk.v4i1.2602.

[10] A. Ridhoi, K. Setyadjit, and B. Hariadi, “Pengaturan Lampu Penerangan Menggunakan,” J. Tek. Ind., vol. 24, no. 1, pp. 45–57, 2021.

[11] D. Setiawan, A. F. Boy, A. Hafidz, and I. Ishak, “Implementasi teknik pwm pada rancang bangun alat deteksi kecepatan kendaraan berdasarkan perputaran roda berbasis mikrokontroler,” J. SAINTIKOM (Jurnal Sains Manaj. Inform. dan Komputer), vol. 19, no. 1, p. 40, 2020, doi: 10.53513/jis.v19i1.224.

[12] M. A. Ulum and S. I. Haryudo, “Perancangan Sistem Monitoring Kecepatan Putar Motor DC Berbasis Internet Of Things Menggunakan Aplikasi BLYNK,” J. Tek. Elektro, vol. 9, no. 1, pp. 855–862, 2020.

