

Perancangan Sistem Pengendali Kecepatan Motor DC Menggunakan Kontroler *Proportional Integral Derivative* Pada Palang Pintu Parkir

Taufiqur Rohman

D4 Teknik Listrik, Fakultas Vokasi, Universitas Negeri Surabaya
taufiqur.19012@unesa.ac.id

Widi Aribowo, Ayusta Lukita Wardani, Reza Rahmadian

D4 Teknik Listrik, Fakultas Vokasi, Universitas Negeri Surabaya
widiaribowo@unesa.ac.id, ayustawadani@unesa.ac.id, rezarahmadian@unesa.ac.id

Abstrak

Penggunaan sistem palang parkir secara berkelanjutan dalam jangka waktu yang lama menyebabkan penurunan optimalitas respon pada motor penggerak. Sebagai hasilnya, terjadi penumpukan di gerbang pintu parkir. Oleh karena itu, diperlukan upaya pengoptimalan pada kontrol motor DC penggerak palang parkir agar mendapatkan respon yang lebih optimal. Tujuan dari penelitian ini adalah menciptakan sistem pengendali kecepatan motor pada palang pintu parkir menggunakan kontroler PID dengan menggunakan arduino uno r3 sebagai mikrokontroler, motor stepper sebagai objek yang dikendalikan, potensiometer sebagai pengatur kecepatan dan sensor HC-SR04 digunakan sebagai sensor untuk menyalakan motor. Apabila nilai potensiometer sudah sesuai dengan setpoint maka akan didapatkan hasil respon dari PID dan respon tersebut akan berfungsi sebagai nilai kecepatan motor. Untuk sensor HC-SR04 berfungsi sebagai sensor untuk menyalakan motor apabila objek berada pada jarak 30 – 80 cm. Dalam mencari parameter PID dilakukan dengan metode *trial error* dan metode analisis dengan membandingkan setiap parameter Kp, Ki, Kd agar mendapatkan respon yang paling optimal dari setiap pengujian. Pada pengujian disetiap parameter, didapatkan hasil optimal pada parameter PID yaitu Kp= 5, Ki=5, Kd= 1. Pada pengujian ini didapatkan nilai rise time 7.57, nilai settling time 11.48 dan overshoot (0%).

Kata Kunci: motor DC, motor stepper, PID kontroler, sensor HC-SR04.

Abstract

The prolonged use of the parking barrier system leads to a decrease in the optimal response of the driving motor. As a result, there is a buildup at the parking gate. Therefore, efforts are needed to optimize the control of the DC motor that drives the parking barrier to achieve a more optimal response. The aim of this research is to create a speed control system for the parking gate motor using a PID controller with Arduino Uno R3 as the microcontroller, a stepper motor as the controlled object, a potentiometer as the speed adjuster, and an HC-SR04 sensor to activate the motor. When the potentiometer value matches the setpoint, the PID response will determine the motor speed. The HC-SR04 sensor functions to activate the motor when an object is within a range of 30 to 80 cm. The PID parameters are determined using trial and error and analytical methods, comparing different Kp, Ki, and Kd values to obtain the most optimal response for each test. The optimal results for the PID parameters were found to be Kp=5, Ki=5, Kd=1. The testing resulted in a rise time of 7.57, settling time of 11.48, and 0% overshoot.

Keywords: DC motor, stepper motor, PID controller, HC-SR04 sensor.

PENDAHULUAN

Kebutuhan akan fasilitas parkir semakin meningkat, terutama di wilayah yang mengalami pertumbuhan pesat. Pengelolaan tempat parkir merupakan aspek layanan yang penting dan memerlukan pengelolaan yang efisien, terutama di rumah sakit, perkantoran, dan lokasi lainnya. Penggunaan terus-menerus sistem palang parkir dalam jangka waktu yang lama menyebabkan penurunan optimalitas respon pada motor penggerak. Akibatnya, dapat terjadi penumpukan di gerbang pintu parkir. Oleh karena itu, diperlukan pengoptimalan pada kontrol motor yang

menggerakkan palang parkir agar mendapatkan respon yang optimal.

Motor DC (direct current) merupakan sebuah perangkat elektromekanik dasar yang difungsikan untuk mengubah energi listrik menjadi energi (Surindra, dkk, 2020). Pengurangan kecepatan pada motor DC sering kali disebabkan oleh adanya beban, menyebabkan ketidakstabilan kecepatan, sehingga memerlukan perancangan kontroler untuk menjaga kecepatan agar tetap konstan. Dengan menerapkan kontroler PID (*Proportional, Integral, Derivative*) kecepatan motor DC dapat dicapai dan dipertahankan secara konsisten sesuai dengan nilai setpoint

yang telah ditetapkan (Muhardian & Krismadinata, 2020). PID merupakan suatu metode kontrol yang umumnya digunakan dalam rekayasa kontrol. Metode pengendali PID ini menggabungkan 3 jenis kontrol, antara lain Proportional (P), Integral (I), dan Derivative (D). Kombinasi dari ketiga jenis kontrol ini mempengaruhi kualitas respons dari sistem kontrol yang diatur (Irhas, dkk, 2020).

Dalam pengendali PID, pada nilai parameter K_p , K_i , dan K_d harus disesuaikan dengan tepat sehingga respons sistem berjalan tepat dengan spesifikasi yang ditentukan. (Supriyanto, dkk, 2022). Pada penelitian yang dilakukan M Reza pada simulasi, kondisinya ideal sehingga kinerjanya dapat menghasilkan hasil yang berbeda jika diimplementasikan secara nyata pada perangkat keras. Hal ini terjadi karena adanya banyak faktor yang dapat mempengaruhi kinerja sistem ketika diimplementasikan dalam keadaan nyata. Sebagai contoh, penambahan pada nilai parameter K_p dapat mengurangi *steady-state error*, memperkecil waktu *rise time*, namun sekaligus dapat memperbesar nilai *overshoot*. Tetapi, jika menambahkan parameter K_i , dapat menyebabkan peningkatan *overshoot*, menurunkan waktu *rise time*, dan bahkan hampir menghilangkan nilai *steady state error*. Kemudian, dengan memasukkan parameter K_d , dapat menyebabkan penyusutan *overshoot*, mengecilkan waktu *rise time*, dan meningkatkan *undershoot* (Putera dan Hidayat, 2022).

Dalam penelitian pengendalian palang pintu parkir ini, digunakan motor DC stepper sebagai objek yang akan dikendalikan. Sehingga, berdasarkan latar belakang tersebut, penulis mencoba merancang “Perancangan sistem pengendali kecepatan motor DC menggunakan kontroler PID pada palang pintu parkir”.

Tujuan dari penelitian ini mencakup beberapa hal, seperti: (1) Untuk memahami bagaimana cara mengaplikasikan metode pengendali PID pada sistem pengendali kecepatan motor DC. (2) Untuk mengetahui hasil respon dari pengujian Sistem pengendali kecepatan Motor DC dengan menggunakan kontroler PID pada palang pintu parkir.

KAJIAN PUSTAKA

Motor DC

Motor DC merupakan tipe motor yang mengonversi energi listrik menjadi energi gerakan mekanik dengan menggunakan arus searah (DC) yang disuplai ke kumparan medan. (Arif dan Aswardi, 2020). Pada penelitian ini menggunakan motor dc stepper Nema 17 sebagai objek yang dikendalikan. Jenis utama motor stepper dibedakan berdasarkan sitem lilitan pada statornya, yaitu:

1. Stepper Unipolar

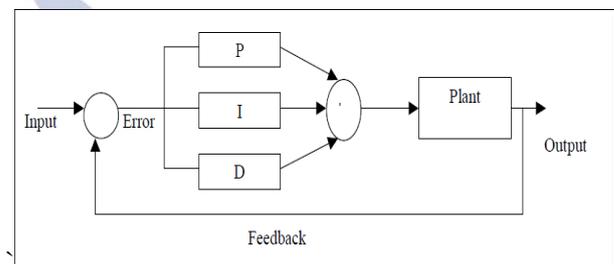
Stepper jenis ini memiliki dua lilitan yang memiliki cabang tengah atau center tap. pada datasheet motor, cabang tengah dari masing-masing lilitan dapat bersatu atau terpisah.

2. Stepper Bipolar

Stepper bipolar tidak memiliki cabang tengah (center tap). Kelebihan dari jenis ini terletak pada torsi yang lebih tinggi dibandingkan dengan unipolar yang memiliki dimensi serupa.

Kontroler PID

Sistem Kontrol PID digunakan untuk mengukur tingkat akurasi suatu sistem instrumentasi dengan memanfaatkan mekanisme umpan balik (*Feedback*) pada sistem tersebut (Hansza dan Haryudo, 2020). Kontrol PID adalah kombinasi dari pengendali *proporsional* (P), *integral* (I), dan *derivatif* (D) (Pane, dkk, 2019). Diagram kontroler PID dapat dilihat pada Gambar 1.



Gambar 1 Diagram kontroler PID
(Sumber : Talakua, 2020)

Menggabungkan kontrol proporsional, integral, dan derivatif dalam sistem kendali PID adalah untuk mencapai tujuan tertentu. Sistem kendali *proporsional* (K_p) memiliki keunggulan dalam waktu respons naik (*rise time*) yang cepat. Kendali *Integral* (K_i) memiliki kelebihan dalam meminimalisir kesalahan/*error*. Kontrol *Derivatif* (K_d) memiliki kelebihan dalam hal mengurangi nilai *overshoot* (Talakua, dkk, 2020).

Arduino Uno R3

Arduino UNO R3 merupakan salah satu versi Arduino UNO yang diperkenalkan pada tahun 2011. Secara spesifik, R3 mengindikasikan bahwa itu adalah revisi ketiga dari produk tersebut. Mikrokontroler yang digunakan adalah arduino atmega328 yang diproduksi oleh Atmel. Mikrokontroler ini merupakan mikrokontroler dengan arsitektur 8 bit (Zanofa, dkk, 2020).

Driver Motor L298N

Driver motor L298N berfungsi sebagai pengendali yang memberikan umpan balik dan untuk menentukan presisi pada motor DC (Anggraini, dkk, 2020). IC L298N Merupakan IC dengan tipe H-bridge yang memiliki kemampuan untuk mengatur beban-beban induktif, seperti relay, solenoid, motor DC, dan motor stepper. Salah satu kelebihan dari modul driver motor L298N adalah kemampuannya dalam mengendalikan motor

dengan presisi sehingga memudahkan dalam pengendalian motor (Muttaqin, dkk, 2021).

Sensor Ultrasonik

Sensor ultrasonik merupakan jenis sensor yang memiliki fungsi untuk mengubah besaran fisik (suara) menjadi besaran listrik atau sebaliknya, dan kemudian mengonversinya menjadi nilai jarak. Prinsip dasar dari sensor ini adalah memanfaatkan pantulan gelombang suara ultrasonik untuk mengukur jarak suatu benda dengan menggunakan frekuensi yang ditentukan sesuai dengan sumber osilator. Sensor ultrasonik memiliki 4 pin yaitu Pin VCC, Trigger, Echo, Ground.

Potensiometer

Potensiometer merupakan sebuah komponen elektronik yang berfungsi sebagai resistor variabel yang dapat mengatur resistansi atau tahanan dalam sebuah rangkaian. Potensiometer biasanya terdiri dari tiga terminal, yaitu terminal input, terminal output, dan terminal sentuh atau penggeser. Poros pada potensiometer dapat diputar atau digeser untuk mengubah posisi terminal sentuh, yang pada gilirannya mengubah nilai resistansi yang diberikan oleh potensiometer. Potensiometer juga berfungsi untuk mengendalikan kecepatan motor DC sesuai dengan kebutuhan.

LCD 16x2 Blue Backlight with I2C Module

LCD 16x2 ini dilengkapi dengan modul I2C yang digunakan untuk menyederhanakan pengkabelan dan juga untuk meminimalisir dalam penggunaan pin pada mikrokontroler. Pada modul I2C, hanya perlu menghubungkan 4 pin pada mikrokontroler (VCC, GND, SCL, SDA) untuk mengakses LCD. Dibandingkan dengan LCD paralel biasa yang membutuhkan 8 pin mikrokontroler untuk mengakses LCD, menggunakan modul I2C hanya memerlukan 4 pin, sehingga lebih efisien dalam penggunaan pin pada mikrokontroler (Ari Ramadhan, dkk, 2020).

METODE PENELITIAN

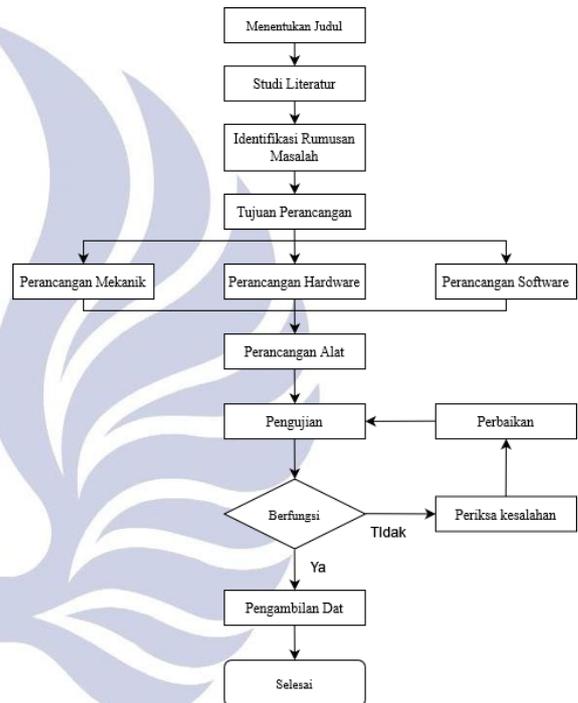
Pendekatan Penelitian

Dalam penelitian ini, digunakan pendekatan kuantitatif sebagai metode penelitian yang melibatkan analisis angka. Inputnya berupa percobaan dan penyesuaian secara real-time, dan hasilnya dicatat melalui pembacaan respons PID dari potensiometer. Pengendali PID dijalankan dengan menggunakan perangkat lunak Arduino IDE.

Rancangan Penelitian

Pada penelitian ini dirancang dengan terstruktur dan sistematis untuk memastikan metodologi yang tepat dan dapat diandalkan. Pada Gambar 2 terdapat *flowchart*

rancangan penelitian yang dimulai dengan langkah-langkah awal seperti menentukan judul, kemudian studi literatur Berangkat dari penelitian-penelitian sebelumnya yang terdapat dalam jurnal-jurnal, lalu mengidentifikasi rumusan masalah dan membuat tujuan perancangan, selanjutnya, penelitian ini dilanjutkan dengan tahap perancangan mekanik, hardware dan software. Untuk selanjutnya yaitu perancangan alat dan langkah selanjutnya adalah pengujian alat, yang kemudian diikuti oleh proses pengumpulan data dan analisis data untuk menyusun kesimpulan dalam laporan penelitian terkait pengujian alat.

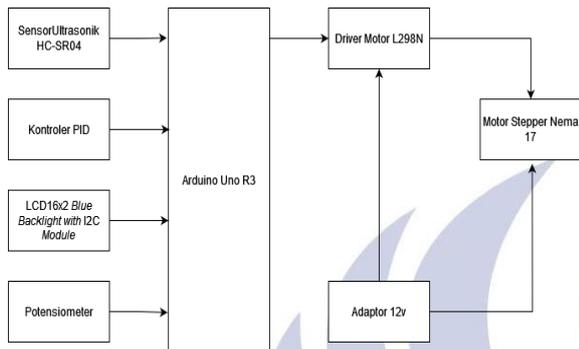


Gambar 2 *Flowchart* Perancangan Penelitian

Perancangan Alat

Proses penyusunan sistem kontrol untuk mengatur kecepatan motor DC dilakukan dengan menerapkan kontroler proporsional integral derivatif (PID). Pada palang pintu parkir bertujuan untuk mempertahankan kecepatan dan posisi motor selaras dengan setpoint yang ditentukan. Penerapan sistem kendali PID dilakukan dengan menyusun algoritma program (perangkat lunak) yang akan diinput ke dalam mikrokontroler. Sistem pengendali PID digunakan untuk mencapai akurasi tinggi pada sistem instrumentasi dengan memanfaatkan umpan balik dari potensiometer sebagai pengatur kecepatan agar sesuai dengan setpoint yang telah ditentukan pada sistem tersebut dengan menggunakan Arduino Uno R3 sebagai papan pengembangan mikrokontroler, dan *driver motor* L298N sebagai IC tipe *H-bridge*, Anda dapat mengontrol berbagai beban induktif seperti relay, solenoid, motor DC, dan motor stepper. *Driver motor* L298N memiliki

kelebihan seperti kepresisian dalam mengendalikan motor sehingga motor menjadi lebih mudah dikendalikan. *Sensor Ultrasonik HC-SR04* memiliki fungsi sebagai pengirim, penerima, dan pengendali gelombang ultrasonik yang digunakan untuk mengukur jarak dari suatu benda. *LCD 16x2 Backlight with I2C Module* digunakan untuk menampilkan hasil pengujian apakah alat sudah bekerja sesuai dengan yang diujikan. Untuk blok diagram alur rancang bangun dapat dilihat pada Gambar 3.



Gambar 3 Blok diagram alur rancang bangun

Pengujian Alat

Pada pengujian alat pada Gambar 2, untuk sensor ultrasonik apabila sensor tidak mendeteksi benda pada jarak yang sudah ditentukan maka akan dilakukan perbaikan pada program arduino dan apabila sensor dapat mendeteksi benda pada jarak yang sudah ditentukan maka sensor akan mengirim sinyal yang akan digunakan untuk menyalakan motor. Lalu pada potensio disetting agar sesuai dengan setpoint yang akan diuji dan sinyal umpan balik dari potensio akan menjadi input dan akan digunakan untuk mencari respon PID.

Pengambilan Data

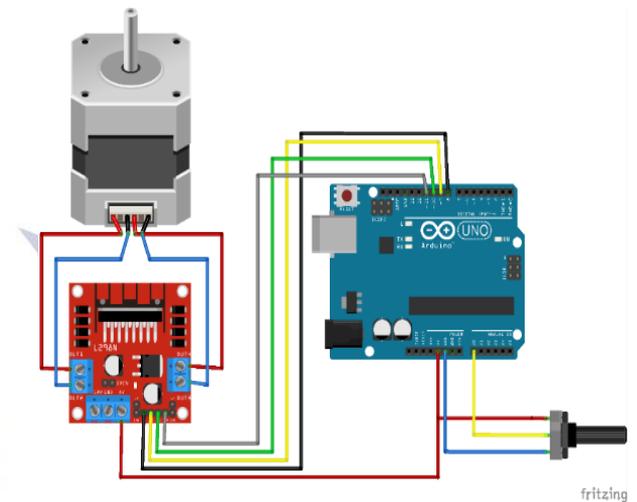
Dalam pengambilan data pada Gambar 2, data yang dianalisis berupa rise-time, settling time dan overshoot berkenaan dengan input yang diberikan. Pada pengujian kontrol PID, dalam mencari nilai parameter Kp, Ki, Kd dilakukan dengan menggunakan metode *trial and error*. Pada metode *trial and error* ini, parameter Kp, Ki, Kd ditentukan dengan cara mencari respon yang paling optimal dengan melakukan 5 kali pengujian pada setiap parameter. Setelah mendapatkan respon dari setiap parameter, maka selanjutnya menentukan parameter yang paling optimal untuk digunakan pada penelitian.

HASIL DAN PEMBAHASAN

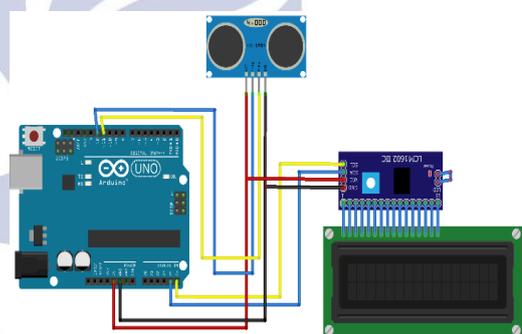
Perancangan Sistem

Pada Gambar 4 dan Gambar 5 merupakan rangkaian wiring dari motor dan wiring dari sensor yang dirancang menggunakan perangkat lunak *Fritzing*. Rangkaian ini akan berfungsi sebagai panduan dalam pembuatan rangkaian elektronika. Dapat dilihat pada Gambar 5 dan

Gambar 6 Arduino digunakan sebagai penghubung semua komponen seperti potensio yang dihubungkan pada pin A0, pin 5v, dan pin gnd. Untuk driver motor dihubungkan pada pin 8, 9, 10, 11 pada arduino. Lalu pada Gambar 6, sensor ultrasonik dihubungkan pada pin 6, 7, pin 5v dan pin gnd. Sedangkan pada LCD dihubungkan pada pin A4 dan A5, serta pin 5v dan pin gnd.



Gambar 4 Wiring motor



Gambar 5 Wiring sensor

Uji Sistem Gerakan Palang Pintu

Pada uji sistem gerakan pada palang pintu, untuk respon dari sensor diatur pada jarak antara 30cm – 80cm dari sensor. Jarak ini didapatkan dari referensi asli palang pintu parkir konvensional sehingga didapatkan data terdapat pada Tabel 1.

Tabel 1 Uji sistem gerakan palang pintu

| Jarak | Respon Sensor |
|--------|---------------|
| 25 cm | Non-Aktif |
| 50 cm | Aktif |
| 75 cm | Aktif |
| 100 cm | Non-Aktif |
| 150 cm | Non-Aktif |

Dari hasil uji coba tersebut pada jarak 25cm, 100cm, dan 150cm dari sensor, benda tidak akan terdeteksi oleh sensor. Pada jarak 50cm dan 75cm, benda akan terdeteksi oleh sensor dikarenakan dari kode arduino diatur untuk jarak pengaktifkan sensor yaitu antara 30cm – 80cm dari sensor.

Penentuan Parameter PID

Dalam pengujian respon PID, data yang dianalisis berupa rise-time, dan overshoot terhadap masukan yang diberikan. Pada pengujian kontrol PID, nilai Kp, Kd, dan Ki diperoleh melalui metode percobaan berulang (*trial and error*). Pada metode *trial and error* ini, untuk mencari nilai parameter Kp, Ki, Kd ditentukan dengan cara mencari respon yang paling optimal dengan melakukan 5 kali pengujian pada setiap parameter. Setelah mendapatkan respon dari setiap parameter, maka selanjutnya menentukan parameter yang paling optimal untuk digunakan pada penelitian. Untuk mengimplementasikan PID *controller* pada sistem perlu menentukan nilai parameter PID *controller* terlebih dahulu. Penentuan nilai parameter ini sangat berpengaruh agar didapatkan nilai parameter PID *controller* yang tepat, sehingga harus dilakukan tuning *controller*. Pada penelitian ini, metode tuning PID *controller* dilakukan secara manual.

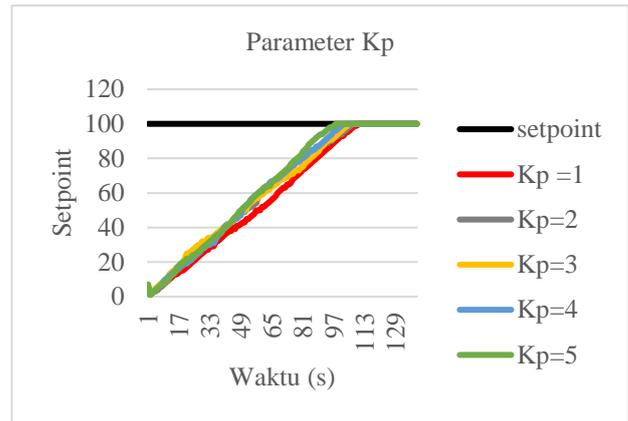
Pengujian Parameter Kp

Dari hasil pengujian yang telah dilakukan, nilai terbaik diperoleh pada Kp = 5, Ki= 1, Kd= 1. Pada nilai tersebut didapatkan *rise time* sebesar 77.20, *settling time* 100.02 sehingga disimpulkan semakin besar nilai Kp maka nilai dari rise time settling time yang dihasilkan akan lebih cepat, sedangkan pada parameter Kp= 1 Ki= 1 Kd= 1 dengan nilai dari *rise time*= 83.60 dan untuk nilai *settling time*= 107.02 yang lebih lambat dari parameter lainnya. Pada Pada Tabel 2 terdapat hasil dari pengujian parameter Kp.

Tabel 2 Parameter Kp

| Parameter Pid | | | Rise Time (s) | Settling Time (s) | Overshoot (%) |
|---------------|----|----|---------------|-------------------|---------------|
| Kp | Ki | Kd | (s) | (s) | (%) |
| 1 | 1 | 1 | 83.60 | 107.02 | 0 |
| 2 | 1 | 1 | 80.40 | 105.02 | 0 |
| 3 | 1 | 1 | 81 | 103.02 | 0 |
| 4 | 1 | 1 | 77.20 | 100.02 | 0 |
| 5 | 1 | 1 | 70.93 | 96.02 | 0 |

Pada Gambar 6 dapat diketahui untuk parameter Kp=5 memiliki respon lebih cepat untuk mencapai setpoint daripada parameter lainnya sehingga dapat disimpulkan, semakin besar Kp maka respon akan semakin cepat.



Gambar 6 Grafik parameter Kp

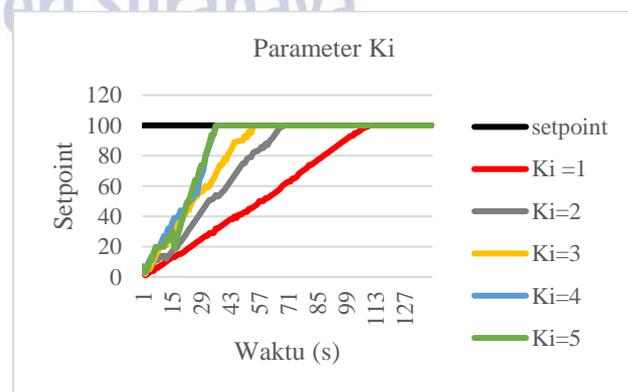
Pengujian Parameter Ki

Pada Tabel 3 untuk nilai terbaik yaitu pada Kp= 1 Ki= 5 Kd= 1. Pada nilai tersebut didapatkan nilai rise time sebesar 26.95, settling time 35.68 sehingga disimpulkan bahwa semakin besar nilai Ki maka nilai dari *rise time* dan *settling time* yang dihasilkan akan lebih cepat, sedangkan pada Kp= 1 Ki= 1 Kd= 1 memiliki nilai rise time= 133 dan settling time= 147.80 yang lebih lambat dari parameter lainnya.

Tabel 3 Parameter Ki

| Parameter Pid | | | Rise Time (s) | Settling Time (s) | Overshoot (%) |
|---------------|----|----|---------------|-------------------|---------------|
| Kp | Ki | Kd | (s) | (s) | (%) |
| 1 | 1 | 1 | 83.60 | 107.02 | 0 |
| 1 | 2 | 1 | 53.53 | 66.53 | 0 |
| 1 | 3 | 1 | 42 | 53.76 | 0 |
| 1 | 4 | 1 | 28.02 | 35.76 | 0 |
| 1 | 5 | 1 | 26.95 | 35.68 | 0 |

Pada Gambar 7 dapat diketahui untuk parameter Ki=5 memiliki respon lebih cepat untuk mencapai setpoint sedangkan pada Ki=1 memiliki respon yang lambat daripada parameter lainnya sehingga semakin besar Ki maka respon akan semakin cepat dalam mencapai setpoint.



Gambar 7 Grafik parameter Ki

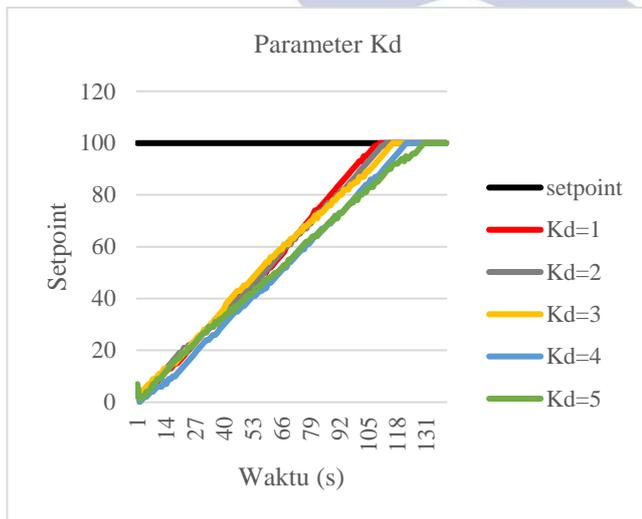
Pengujian Parameter Kd

Pada Tabel 4 didapatkan respon terbaik yaitu pada $K_p = 1$ $K_i = 1$ $K_d = 1$. Pada nilai tersebut didapatkan rise time sebesar 83.60, settling 107.02 sehingga disimpulkan bahwa semakin kecil nilai K_d maka *rise time settling time* yang dihasilkan akan lebih cepat, sedangkan pada $K_p = 1$ $K_i = 1$ $K_d = 5$ memiliki nilai rise time= 97.05 dan settling time= 128.02 yang lebih lambat dari parameter lainnya.

Tabel 4 Parameter Kd

| Parameter Pid | | | Rise Time | Settling Time | Overshoot |
|---------------|-------|-------|-----------|---------------|-----------|
| K_p | K_i | K_d | (s) | (s) | (%) |
| 1 | 1 | 1 | 83.60 | 107.02 | 0 |
| 1 | 1 | 2 | 87.80 | 110.02 | 0 |
| 1 | 1 | 3 | 90 | 114.02 | 0 |
| 1 | 1 | 4 | 87.20 | 120 | 0 |
| 1 | 1 | 5 | 97.05 | 128 | 0 |

Pada Gambar 8 dapat dianalisis jika pada $K_d = 5$ menghasilkan respon yang lebih lambat dalam mencapai setpoint sehingga dapat diambil kesimpulan jika semakin besar nilai K_d maka akan mempengaruhi nilai dari rise time dan settling time. Sedangkan pada $K_p = 1$ memiliki respon yang lebih cepat dalam mencapai setpoint dari pada parameter lainnya sehingga dapat disimpulkan semakin kecil nilai K_d maka respon akan semakin cepat mencapai setpoint.



Gambar 8 Grafik parameter Kd

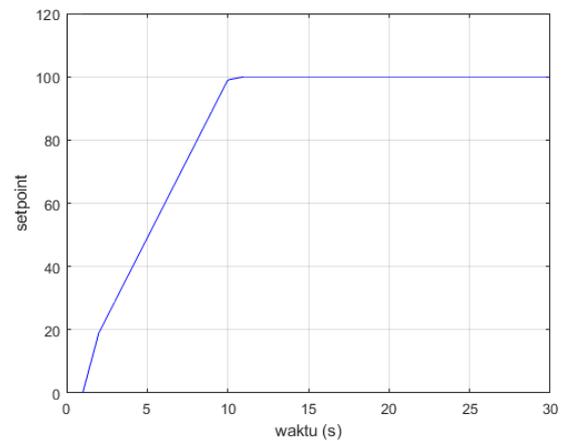
Hasil pengujian respon optimal

Dari analisis pada pengujian di setiap parameter, dapat diambil data paling optimal yaitu $K_p = 5$, $K_i = 5$, $K_d = 1$. Dan untuk hasil pengujian parameter tersebut dalam mencari respon PID dapat dilihat pada Tabel 5.

Tabel 5 Hasil pengujian respon optimal

| Parameter Pid | | | Rise Time | Settling Time | Overshoot |
|---------------|-------|-------|-----------|---------------|-----------|
| K_p | K_i | K_d | (s) | (s) | (%) |
| 5 | 5 | 1 | 7.57 | 11.48 | 0 |

Pada tabel 5 didapatkan hasil pengujian respon paling optimal dari hasil pengujian disetiap parameter. Pada parameter PID $K_p = 5$ $K_i = 5$ $K_d = 1$ didapatkan nilai rise time 7.57s dan settling time 11.48s. Nilai ini menunjukkan bahwa respon paling optimal didapatkan pada pengujian tersebut. Pada Gambar 9 terdapat grafik dari hasil respon optimal.



Gambar 9 Grafik hasil pengujian respon optimal

PENUTUP

Kesimpulan

Dengan berdasarkan pengumpulan dan hasil analisis data yang telah dilakukan, dapat disimpulkan bahwa *prototype* palang pintu parkir dengan pengendali PID ini berhasil dibuat dengan menggunakan metode *trial and error* dan metode analisis. Cara memperolehnya yaitu dengan menguji setiap parameter dengan melakukan 5 kali pengujian pada setiap parameter K_p , K_d , K_i untuk mendapatkan respon paling optimal. Hasil pengujian menunjukkan bahwa sistem mampu mencapai kecepatan sesuai dengan setpoint yang ditentukan pada penelitian ini menggunakan setpoint 100 dengan mengimplementasikan kontroler PID. Dari hasil analisis telah dilakukan, diperoleh nilai *rise time* sebesar 7.57, nilai *settling time* sebesar 11.48 dan *overshoot* 0% pada nilai parameter $K_p = 5$, $K_i = 5$, dan $K_d = 1$.

Saran

Ada beberapa rekomendasi yang dapat diimplementasikan untuk meningkatkan kinerja sistem agar mencapai hasil yang lebih optimal, yaitu untuk penelitian selanjutnya diharapkan mampu mengembangkan penggunaan metode PID dengan menggunakan metode yang lain dan untuk menggunakan metode PID diharapkan dapat aplikasikan

pada alat lain, lalu untuk mencari respon PID dapat dikembangkan lagi dalam mencari parameter Kp, Ki, Kd.

DAFTAR PUSTAKA

- Anggraini. Siti Fatimah, Ma'arif. Alfian dan Puriyanto. Dwi Ricky. 2020. *Pengendali PID pada Motor DC dan Tuning Menggunakan Metode Differential Evolution (DE)*. TELKA - Telekomunikasi Elektronika Komputasi Dan Kontrol, 6(2), 147–159.
- Arif. Dio Taufiq dan Aswardi. 2020. *Kendali Kecepatan Motor DC Penguat Terpisah Berbeban Berbasis Arduino*. 06(02), 33–43.
- Hansza. Rizky dan Haryudo. Subuh Isnur. 2020. *Rancang Bangun Kontrol Motor DC dengan PID Menggunakan Perintah Suara dan Monitoring Berbasis Internet of Things (IOT)*. Jurnal Teknik Elektro, 9(2), 477–485.
- Irhas. Muhammad, Iftitah dan Ilham Siti Asyiqah Azizah. 2020. *Penggunaan Kontrol Pid Dengan Berbagai Metode Untuk Analisis Pengaturan Kecepatan Motor DC*. JFT : Jurnal Fisika Dan Terapannya, 7(1), 78.
- Muhardian. Reza dan Krismandinata. 2020. *Kendali Kecepatan Motor DC Dengan Kontroller PID dan Antarmuka Visual Basic*. JTEV (Jurnal Teknik Elektro Dan Vokasional), 6(1), 328.
- Muttaqin. Imam Rama dan Santoso. Dian Budi. 2021. *Prototype Pagar Otomatis Berbasis Arduino Uno Dengan Sensor Ultrasonic Hc-SR04*. JE-Unisla, 6(2), 41.
- Pane. Jago, Surya. Ade P, Novita. Siskka, Mazmur, Romi Mazmur P, Aryza Selly, Hamdani, Rizky dan Ahmad. 2019. *Implementasi PID Dalam Mengendalikan Motor Menggunakan Metode PID dan Mikrokontroler Atmega*. Sainteks 2019, Mv, 196–201.
- Putera. Muhammad Reza Aditya Nurkholis dan Hidayat. Rahmad. (2022). *Kendali Kecepatan Motor Dc Menggunakan Pengendali Pid Dengan Encoder Sebagai Feedback*. Studi, P., Elektro, T., Karawang, U. S., & Uno, A. 7(1), 50–56.
- Ramadhan. Muhammad Ari., Noertjahjono. Sidik dan Wahyuni. Febriana Santi. 2020. *Rancang Bangun Akses Kunci Pintu Gerbang Indekos Menggunakan E-Ktp (Elektronik Kartu Tanda Penduduk) Berbasis Mikrokontroller*. JATI (Jurnal Mahasiswa Teknik Informatika), 4(2), 239–246.
- Supriyanto. Hadi, Suryatini. Fitria, Martawireja. Abdur Rohman Harits dan Rudiansyah. Hendy. 2022. *Implementasi Kontroler PID Dengan Metode Tuning Ziegler-Nichols Dan Cohen-Coon Pada Sistem SCADA Kendali Level Air*. 8 (September), 149–157.
- Surindra. Mochamad Denny, Widyaningsih. Wiwik Purwanti, Margana, Supriyo dan Mulud. Teguh Harjono. 2020. *Sistem Kontrol Proportional Integral Derivative (Pid) Untuk Mengatur Kecepatan Motor Dc Menggunakan Mikrokontroller*. Prosiding Seminar Nasional NCIET, 1(1), 528–534.
- Talaku. Eddy Lybrech, Utama. Yoga Alif Kurniadan Andriyanto, dan Indra. 2020. *Optimasi Kontrol Pid Untuk Kendali Kecepatan Motor DC*. 1–8.
- Zanofa. Arif Pratama, Arrahman. Ristiandika, Bakri. Muhammad dan Budiman. Arief . 2020. *Pintu Gerbang Otomatis Berbasis Mikrokontroler Arduino Uno R3*. 1(1), 22–27.