

Modul Kontrol Kecepatan Empat Roda Robot dengan Kontrol PI Dilengkapi *Interface* UART

Mokh. Ivan Fa'iq Fatwa Auliah, Totok Winarno, Leonardo Kamajaya

e-mail: officialmiffa.21@gmail.com, totok.winarno@polinema.ac.id, leonardo42@polinema.ac.id

Jurusan Teknik Elektro, Politeknik Negeri Malang, Jalan Soekarno Hatta No.9 Malang, Indonesia

Informasi Artikel

Riwayat Artikel

Diterima 22 November 2022

Direvisi 20 Februari 2023

Diterbitkan 31 Juli 2023

Kata kunci:

Kontrol PI
Mobile Robot
omniwheel.
UART
Ziegler-Nichols

Keywords:

PI Control
Mobile Robot
omniwheel.
UART
Ziegler-Nichols

Penulis Korespondensi:

Mokh. Ivan Fa'iq Fatwa Auliah,
Jurusan Teknik Elektro,
Politeknik Negeri Malang,
Jl. Jalan Soekarno Hatta No. 9, Malang, Jawa Timur, Indonesia, 65142.
Email: officialmiffa.21@gmail.com
Nomor HP/WA aktif: +62 81216487927

ABSTRAK

Pada proses industri, *mobile robot* beroda difungsikan untuk meringankan tugas manusia serta menyelesaikan permasalahan industri seperti memindahkan atau menyeleksi barang. Sebagai langkah awal dalam pembuatan dan pengembangan robot, maka diperlukan sebuah media pembelajaran robotika yang dapat mencakup kontrol pada robot baik kontrol level tinggi (*high level control*) maupun kontrol level rendah (*low level control*). Salah satu kontrol level rendah (*low level control*) pada *mobile robot* adalah kontrol PI. Kontrol PI pada *mobile robot* untuk mengontrol pergerakan roda robot agar berjalan tetap stabil dan mempertahankan kecepatannya terhadap situasi yg terjadi pada robot dan lintasannya. Ada beberapa cara untuk menentukan nilai pada konstanta P dan I diantaranya menggunakan metode *Ziegler-Nichols* dan metode *Trial error*, pada modul Kontrol PI ini dapat digunakan untuk menguji kedua metode tersebut. Modul kontrol PI ini juga didesain dapat mengirimkan data dari kontrol level rendah yaitu mikrokontroler ke kontrol level tinggi yang ditangani oleh *computer* menggunakan komunikasi UART. Dari beberapa nilai variable Kp dan Ki yang diuji, didapatkan hasil bahwa nilai Kp 0,7 dan Ki 5,7 memiliki respon putaran keempat motor yang stabil serta mencapai *set point* kecepatan yaitu 150 RPM.

ABSTRACT

In industrial processes, wheeled mobile robots function to ease human tasks and solve industrial problems such as moving or selecting goods. As a first step in the manufacture and development of robots, a robotics learning media is needed that can cover the control of robots both high level control and low level control. One of the low-level controls on mobile robots is PI control. PI control on mobile robots to control the movement of the robot wheels so that it runs stably and maintains its speed against the situation that occurs on the robot and its path. There are several ways to determine the value of the P and I constants including using the Ziegler-Nichols method and the Trial error method, this PI Control module can be used to test both methods. This PI control module is also designed to transmit data from low-level controls, namely microcontrollers to high-level controls handled by computers using UART communication. From several variable values of Kp and Ki that were tested, it was found that the value of Kp 0.7 and Ki 5.7 had a stable response to the rotation of the four motors and reached the speed set point of 150 RPM.



1. PENDAHULUAN

Industri modern masa kini banyak yang mengandalkan motor DC karena keandalan dan ketahanannya, namun ketika diberi beban motor DC akan cenderung melambat kecepatannya jika tidak diberi sistem kendali [1]. Sistem kendali telah memegang peranan yang penting dalam perkembangan ilmu dan teknologi [2]. Hal lain yang sering ditemukan pada industri ialah *interface*, *interface* adalah sistem yang menghubungkan antara manusia dan mesin seperti penyetelan variable sistem kendali dan kecepatan *real time* dari motor DC [3]. Pada kegiatan praktikum mahasiswa memerlukan suatu alat peraga yang memadai dan mencakup kebutuhan praktikum sistem kendali [4]. Salah satu media pembelajaran dengan menggunakan robot roda empat yang berbasis robot beroda, jenis roda yang digunakan adalah roda omni dimana pergerakan robot ini dapat bergerak kesegala arah [5].

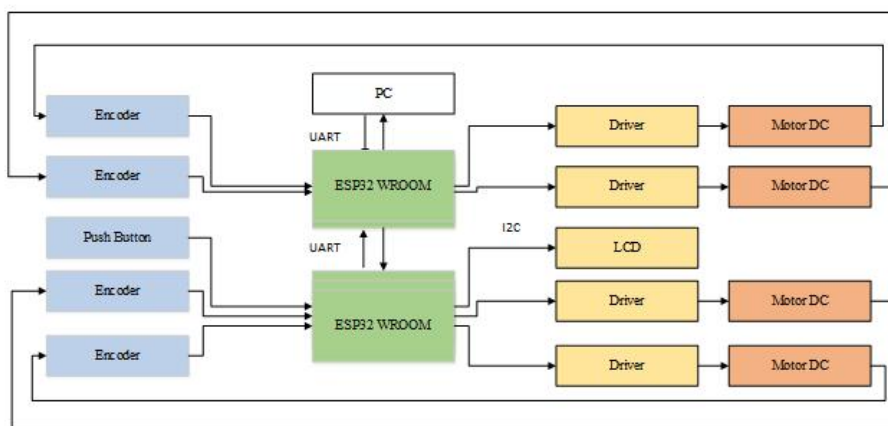
Beberapa penelitian yang dilakukan seperti Santoso S [6] Kontrol PI memiliki peran agar pergerakan robot lebih stabil dan presisi serta kecepatan motor DC yang sesuai dengan *set point* yang ditentukan. Kemudian pada penelitian Mila Fauziah [7] Kontrol PI dalam sistem kendali saling melengkapi dimana kontrol P menutupi keterlambatan Kontrol I sedangkan Kontrol I mengeliminasi *offset* yang dihasilkan kontrol P sehingga menghasilkan respon waktu tunda sebesar 0 detik, waktu naik sebesar 11,5 detik, *settling time* sebesar 19 detik, waktu puncak sebesar 2 detik, *overshoot* maksimum sebesar 46 RPM *error steady state* sebesar 0% dengan *set point* motor DC 40 RPM. Kemudian penelitian yang dilakukan oleh Ermanu [8] motor DC yang diberi sistem kontrol kecepatan saat terjadi gangguan berupa pengereman motor DC menunjukkan respon yang cepat dan stabil.

Fokus penelitian ini ialah pembuatan modul belajar yang dapat digunakan mempraktikkan sistem kendali kecepatan motor dengan basis robot beroda yang dapat bergerak kesegala arah serta dilengkapi *interface* yang dapat digunakan untuk menyetel variabel kontrol dan menampilkan kecepatan serta respon motor secara *real time*. Modul ini menggunakan sensor *rotary encoder* sebagai pembaca kecepatan motor DC, ESP32 sebagai mikrokontroler yg menangani kontrol PI dan input dari sensor, *push button*, menampilkan karakter *interface* LCD serta komunikasi dengan *computer* melalui peripheral UART.

2. METODE PENELITIAN

2.1 Diagram Blok Sistem

Pada Gambar 1 menunjukkan diagram blok dari keseluruhan modul meliputi input yaitu *push button* dan *rotary encoder* sebagai pembaca nilai kecepatan motor kemudian ESP32 sebagai mikrokontroler yang memproses semua masukan dan memberikan respon output berupa kecepatan motor pada driver BTS7960B yang akan menggerakkan kecepatan motor DC 24V dan tampilan display pada LCD 16x2. Komunikasi kedua ESP32 dan PC menggunakan peripheral UART :



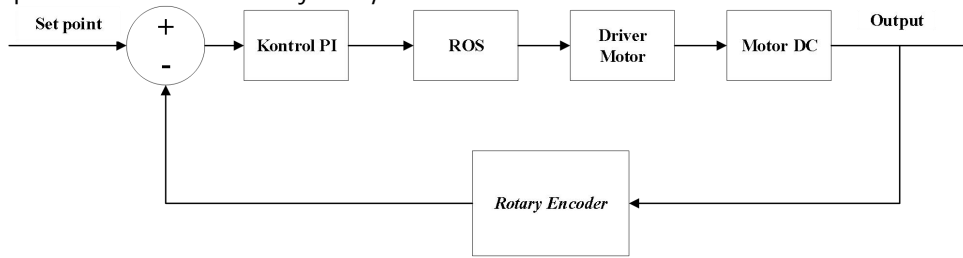
Gambar 1: Diagram Blok Sistem

2.2 Diagram Blok Kontrol

Prinsip Kerja dapat dilihat pada Gambar 2 modul sistem pembelajaran kontrol PI ini dengan memasukkan kecepatan putaran motor, kecepatan ini akan menjadi *set point* dari kontrol PI tersebut. Kemudian memasukkan variabel P dan I yang akan mengontrol kecepatan motor itu kemudian kecepatan itu akan ditampilkan pada LCD kemudian Ketika motor berputar maka *interface* yang berada pada komputer akan menampilkan grafik secara *real*



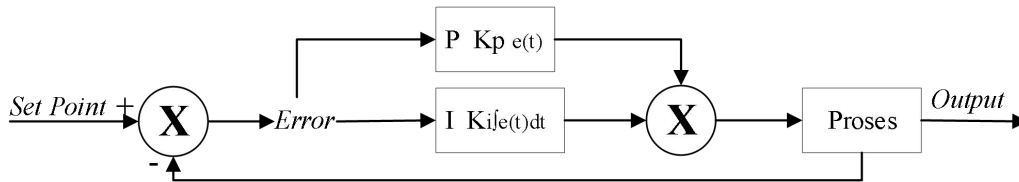
time dimana grafik itu digunakan untuk mengetahui apakah nilai K_p dan K_i yang dimasukkan beresilasi dan mengetahui respon kontrol untuk menuju *set point*.



Gambar 2: Diagram Blok Kontrol

2.3 Perancangan Kontrol PI

Pada Gambar 3 tujuan dari pemodelan kontrol PI ini adalah untuk parameter pada kontrol Propotional dan Integratif pada Modul Kontrol PI hingga modul dapat memberi kontrol pada motor agar sesuai dengan yang sesuai dengan karakteristik kontrol PI. Fungsi dari kontrol PI pada modul ini bertujuan untuk memproses sinyal kesalahan atau *error* yang biasanya didapatkan dari hambatan pada aktuator atau berkurangnya tegangan pada catu daya, kontrol PI akan mengolah nilai kesalahan yang kemudian akan dirubah menjadi sistem kendali dan diteruskan pada driver motor.



Gambar 3: Diagram Kontrol PI

Metode populer untuk mencari konstanta pengontrol P, PI, dan PID adalah metode *Ziegler-Nichols*. Metode ini dengan meniadakan keuntungan integral dan diferensial dan kemudian menaikkan *gain* proporsional sampai sistem tidak stabil. Nilai K_P pada titik ketidakstabilan disebut K_{MAX} ; frekuensi getarannya adalah f_0 . Metode ini kemudian memundurkan *gain* proporsional sejumlah yang telah ditentukan dan menetapkan *gain* integral dan diferensial sebagai fungsi dari f_0 . Konstanta P, I, dan D dapat diatur sesuai dengan Tabel 1.

Tabel 1: Metode *Ziegler – Nichols*

Jenis Kontrol	K_p	K_i	K_d
Kontrol P	0,5 K_{max}	0	0
Kontrol PI	0,45 K_{max}	1,2 f_0	0
Kontrol PID	0,6 K_{max}	2,0 f_0	0,125 / f_0

Formula Matematis dari Kontrol PI adalah:

$$u = K_p \cdot e(t) + \int_0^t e(t) dt \quad (1)$$

Kontrol PI dapat dirumuskan menjadi berbagai metode jika berdasarkan dari formula diatas, yang sering digunakan adalah metode transformasi Z, ZOH, *Bilinear transformation*, *Eulers method*. Bentuk kontroler PI diskrit adalah sebagai berikut ;

$$u(k) = K_p e_k + K_i T \sum_0^k e_k \quad (2)$$

Dalam Bahasa pemrograman, rumus diatas penulisannya dapat ditulis seperti ini ;

Mokh. Ivan Fa'iq Fatwa Auliah: Modul Kontrol Kecepatan Empat...

p-ISSN: 2356-0533; e-ISSN: 2355-9195



$$u = K_p \times error + K_i \times (error + last_error) \times TS \quad (3)$$

Langkah awal menggunakan kontrol propotional terlebih dahulu dan mengabaikan konstanta integral. Nilai konstanta ditambahkan terus hingga mendekati nilai stabil namun roda masih berisolasi. Saat menggunakan kontrol integral, nilai *time sampling* juga mempengaruhi perhitungan PI.

Time sampling berkaitan dengan program kontrol PI akan dieksekusi setiap 0,1/detik atau tiap *time sampling*. Nilai integral akan dikalikan dengan *time sampling*. Menentukan nilai *time sampling* 0,1detik karena dianggap dengan nilai tersebut mikrokontroler sanggup mengeksekusi program atau nilai tersebut cukup optimal.

$$TS = \frac{prescaler \times periode}{frekuensi\ timer} \quad (4)$$

Nilai *time scaler*, *period* dan frekuensi *timer* sudah ditentukan, maka dengan rumus diatas akan mencari nilai *prescaler*.

$$Prescaler = \frac{Time\ Scaler \times Frekuensi\ timer}{Priode} \quad (5)$$

$$Prescaler = \frac{0,1 \times 90Mhz}{1000}$$

$$Prescaler = 4500$$

Maka dengan nilai *prescaler* 4500, periode 1000 dan frekuensi *clock* 90 Mhz dapat membangkitkan *timer interrupt* yang akan mengeksekusi program PI setiap per 0,1 detik.

2.4 Desain Mekanik



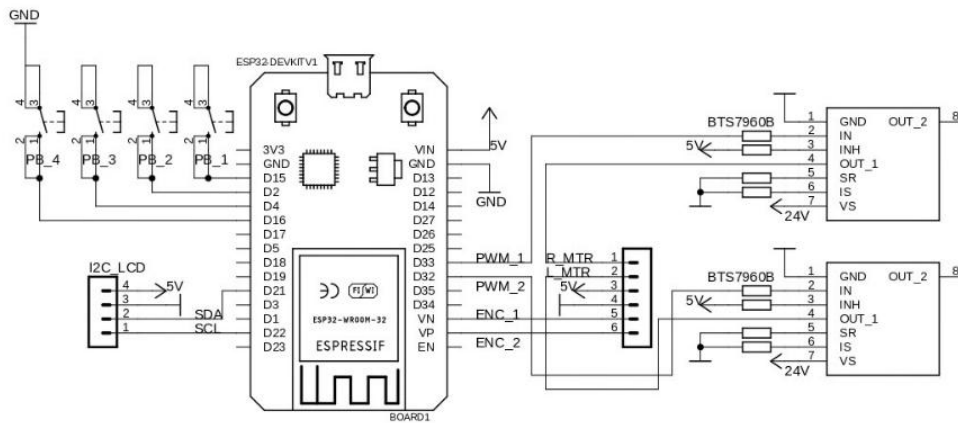
Gambar 4: Desain Mekanik

Pada gambar 4 desain mekanik, mobile robot menggunakan plat almunium 3mm sebagai *base* serta terdapat 4 roda *omniwheels* dengan dimensi 60 mm. Pada bagian atas robot terdapat 4 *push button* dan LCD 16x2 berfungsi memassukkan nilai konstanta serta *set point* dan menampilkan parameter nilai serta menu pilihan yang terdapat pada modul.

2.7 Perancangan Rangkaian Elektrikal Modul Kontrol PI

Untuk membuat modul dengan spesifikasi dapat memberi kontrol PI dan *set point* pada kontrol kecepatan motor maka diperlukan rangkaian elektrik sebagai rangkaian elektrik kontrol pada modul. Rangkaian tersebut terdiri dari ESP32, *rotary encoder*, motor DC geared, driver motor, LCD 16x2 dan *push button*.





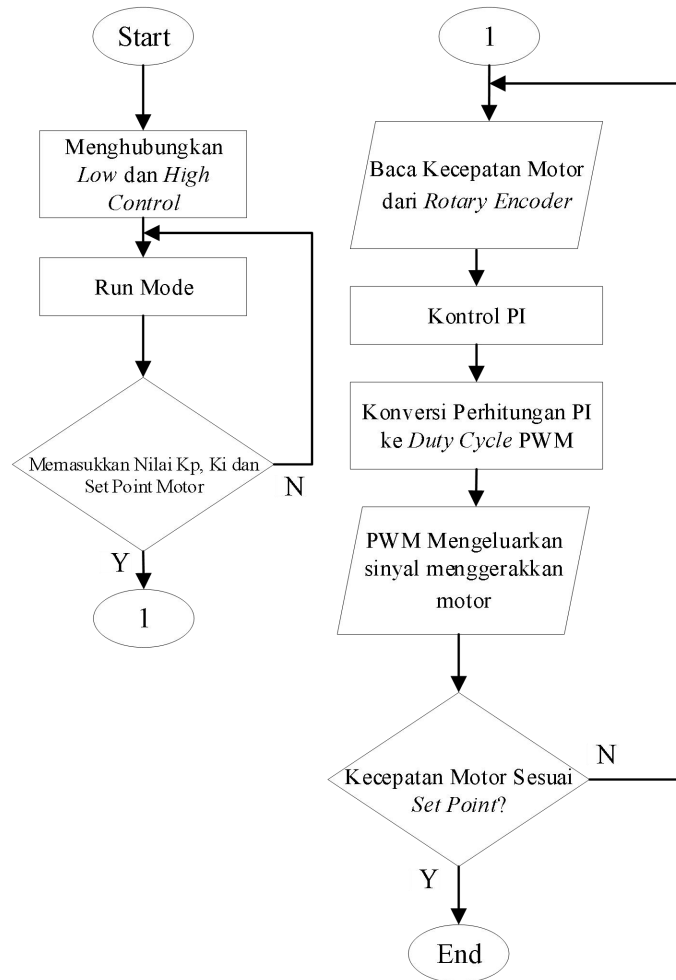
Gambar 5: Rangkaian Elektrikal

Rangkaian pada Gambar 5 digunakan sebagai Main board pada mobile robot. ESP32 berhubungan langsung dengan LCD, Tombol, dan Driver. Pada kontroler ESP32 akan ditambahkan rangkaian tambahan yaitu *shield* ESP32. Hal ini dikarenakan agar penyambungan keseluruhan elektrik tidak menggunakan kabel tipe *socket male*. Dengan menggunakan *shield* ESP32 yang didesain atau *lay out* dan dicetak menggunakan PCB maka akan lebih mempermudah untuk penyambungan rangkaian keseluruhan Input ataupun Output yang dibutuhkan.

2.9 Flowchart Software

Perancangan Software yang berada pada mikrokontroler ESP32. Pada gambar 6 dapat dicermati bahwa saat robot dinyalakan komponen seperti LCD akan melakukan *scanning* kemudian LCD akan menampilkan karakter dan push button bisa digunakan untuk memilih menu yang ada pada LCD. Pada menu disediakan menu *tuning* PI berfungsi untuk memasukkan nilai K_p dan K_i kemudian selanjutnya adalah memasukkan nilai *set point* pada menu RPM kemudian PWM akan menggerakkan motor. Apabila pada motor diberi hambatan, maka sensor *rotary encoder* akan membaca kecepatan motor aktual saat terkena hambatan selanjutnya nilai tersebut akan dibaca oleh kontrol PI sebagai nilai error dan kontrol akan mempertahankan kecepatan motor sesuai *set point* meskipun motor mengalami hambatan. Ketepatan dan kecepatan respon motor mempertahankan kecepatannya tergantung dari nilai yang diberikan pada kontrol PI.





Gambar 6 : Flowchart Sistem

3. HASIL DAN PEMBAHASAN

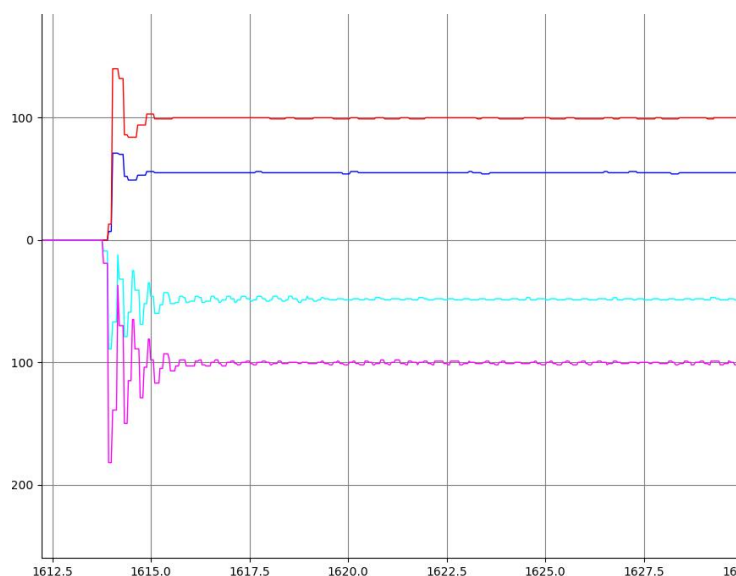
3.1 Pengujian Komunikasi peripheral UART pada *High Level Control* (PC) dan *Low Level Control* (ESP32)

Pengujian ini bertujuan untuk mengetahui protokol serial komunikasi mikrokontroler dengan PC dimana dari ESP32 dan PC yang saling terhubung secara peripheral UART akan saling mengirimkan data yang diperlukan dalam hal ini adalah respon kecepatan motor. Ada beberapa cara untuk menerima dan mengirim data yang akan digunakan namun diharuskan dengan node yang sama, sistem data ini menggunakan *subscriber* dan *publisher* data antar dua *hardware*.





Gambar 7: Kecepatan motor pada LCD 16x2



Gambar 8: Tampilan grafik respon kecepatan motor

Pada Gambar 8 menunjukkan adanya komunikasi antara mikrokontroler dengan PC (RQT) melalui protokol ROS dimana yang terkirim dari mikrokontroler adalah kecepatan motor yang terkontrol dengan kontrol PI kemudian RQT menampilkan grafik respon perilaku motor dan pada gambar 7 mikrokontroler juga menampilkan pada LCD yaitu kecepatan motor dimana nilai putaran motor juga tertampil dengan nilai yang sama pada PC (RQT). Pada RQT yang tertampil pada PC dan Nilai yang terbaca pada Mikrokontroler (ESP32) yang tertampil pada LCD 16x2 memiliki kesamaan yaitu keduanya membaca kecepatan motor 1 dan 3 adalah 50 RPM sedangkan motor 3 dan 4 adalah 100 RPM.

3.2 Pengujian dan Pengamatan Respon Motor Terhadap Nilai Kp dan Ki

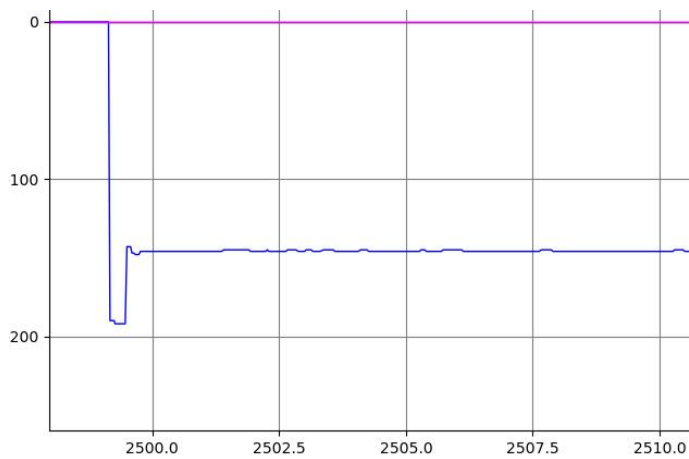
Pengujian ini motor tanpa diberi beban namun diamati respon motor mengalami osilasi atau tidak. Pengujian ini menggunakan motor DC geared 24V nilai Kp dan Ki yang digunakan berasal dari perhitungan *Ziegler Nichols* dan *trial error*. Perhitungan dari *Ziegler Nichols* diperoleh nilai Kp 0,7 dan Ki 5,7.



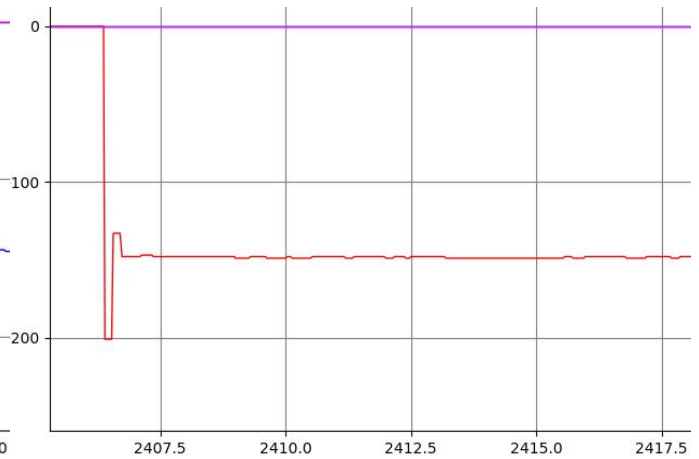
Tabel 2 : Respon Motor

Kp	Ki	Set Point	Motor 1	Motor 2	Motor 3	Motor 4	Keterangan
0,1	0,7	150 RPM	10 RPM	0 RPM	0 RPM	0 RPM	Tidak Mencapai <i>set point</i>
0,3	2,7	150 RPM	60 RPM	50 RPM	70 RPM	60 RPM	Tidak Mencapai <i>set point</i>
0,7	5,7	150 RPM	150 RPM	150 RPM	150 RPM	150 RPM	Sesuai <i>set point</i>
1	8,7	150 RPM	230 RPM	230 RPM	230 RPM	230 RPM	<i>overshoot</i>
1,4	12,7	150 RPM	230 RPM	230 RPM	240 RPM	230 RPM	<i>overshoot</i>

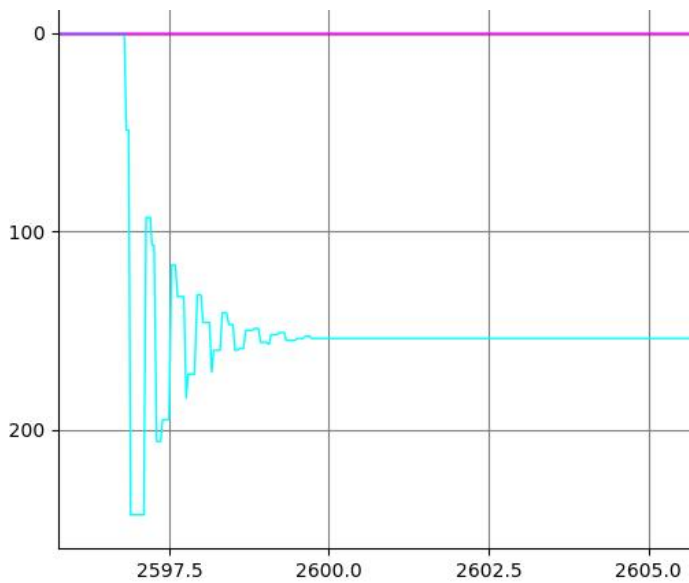
Pada Tabel 2 menunjukkan bahwa nilai konstanta yang sama jika diterapkan pada motor yang berbeda dengan spesifikasi motor yang mirip akan memiliki respon yang berbeda baik secara kecepatan maupun kestabilan pada motor, Pada percobaan yang telah dilakukan Nilai Kp 0,7 dan Ki 5,7 menunjukkan respon yang baik dimana motor mencapai nilai *set point* yaitu 150 RPM dan pada Gambar 9, 10, 11 dan 12 grafik respon motor stabil dan tidak mengalami osilasi secara terus menerus.



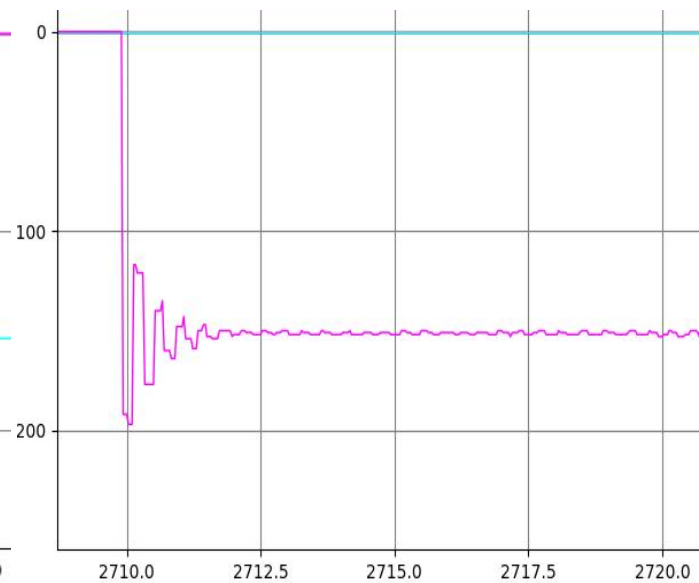
Gambar 9: Respon Motor 1



Gambar 10: Respon Motor 2



Gambar 11: Respon Motor 3



Gambar 12: Respon Motor 4



4. KESIMPULAN

Pengaturan Kecepatan motor DC dapat dilakukan dengan metode kontrol PI dengan syarat untuk koreksi nilai *error* pada motor tersebut terdapat sensor *rotary encoder*. Kontrol PI akan memberi kontrol agar putaran motor stabil pada kecepatan yang ditentukan dan apabila diberi gangguan respon motor akan cepat menyesuaikan untuk Kembali pada titik kestabilan *set point*. Namun perlu diketahui dengan nilai konstanta yang sama belum tentu nilai tersebut akan memberi respon yang sama meskipun motor memiliki spesifikasi yang sama selain itu dengan perhitungan *Ziegler Nichols* dalam perumusan nilai K_p dan K_i tidak selamanya akan memberi respon yang sama pada motor maka diperlukan sebuah percobaan yang konstan untuk menentukan nilai K_p dan K_i yang sesuai dengan respon motor. Pada percobaan digunakan 4 variabel K_p dan K_i yang sama pada keempat motor dan pada nilai K_p 0,7 dan K_i 5,7 kecepatan motor mencapai *set point* 150 RPM dan respon motor stabil tidak berkurang maupun lebih saat berputar.

DAFTAR PUSTAKA

- [1] Murtono, A., Kamajaya, L., & Shulton, M. (2021). Implementasi Kontrol PID untuk Analisis Pengaturan Kecepatan Motor DC Menggunakan STM32. September, 310–314.
- [2] Saka, P., Asa, G., & Priyambodo, S. (2016). SISTEM PEMBELAJARAN KONTROL PID (PROPORSIONAL INTEGRAL DERIVATIF) PADA PENGATUR KECEPATAN MOTOR DC PID (PROPORTIONAL INTEGRAL DERIVATIVE) CONTROL LEARNING SYSTEM ON DC MOTOR SPEED CONTROLLER S1 Teknik Elektro IST Akprind Yogyakarta Intisari Keywords : Jurnal Elektrikal, 3, 72–77.
- [3] Haryanto, H., & Hidayat, S. (2016). Perancangan HMI (Human Machine Interface) Untuk Pengendalian Kecepatan Motor DC. *Setrum : Sistem Kendali-Tenaga-Elektronika-Telekomunikasi-Komputer*, 1(2), 58. <https://doi.org/10.36055/setrum.v1i2.476>.
- [4] Syukriah, P., Romdlony, M. Z., & Wibowo, A. S. (2019). Perancangan Alat Peraga Kendali Pid Analog Pada Sistem Kecepatan Putar Motor Dc. *EProceedings of Engineering*, 6(2), 2810–2817.
- [5] Priyambudi, A., Firman, B., & Kristiyana, S. (2018). Kendali Kecepatan Motor Arus. *Jurnal Teknologi Technoscientia*, 10(2), 209–217.
- [6] Santoso, S., & Mursyid, S. (2015). Kontrol Proportional Integral (Pi) Pada Robot Line Follower. *Jurnal Sains Dan Informatika*, 1. <http://jsi.politala.ac.id/index.php/JSI/article/view/73%0Ahttps://jsi.politala.ac.id/index.php/JSI/article/download/73/40>.
- [7] Fauziyah, M., Dewatama, D., & Atisobhita, M. (2017). Implementasi Kontrol PI Pada Pengaturan Kecepatan Motor DC. *Prosiding Serminal Nasional Teknologi Elektro Terapan*, 01(01), 217–222.
- [8] Hakim, E. A., Suhardi, D., Diah, C., & Khair, I. (2020). Modul Sistem Kontrol Kecepatan Motor Dc Berbasis Komputasi Digital. 71–79.
- [9] Ellis, G. (2012). *Control System Design Guide: Using Your Computer to Understand and Diagnose Feedback Controllers*. <https://books.google.com/books?id=hTXsXTt5kewC&pgis=1>.
- [10] Fadhila, I., Siradjuddin, I., & Putri, R. I. (2021). Trajectory Tracking Robot Omnidirectional 4 Roda Dengan Visualisasi Rviz. 8.

