

Improving the Accuracy of Manual Control of an Automatic Guided Vehicle (AGV) Robot by Applying the Proportional Integral Derivative (PID) Algorithm

Satria Apsada^{1*}, Risfendra²

^{1,2} Teknik Elektro Industri, Fakultas Teknik, Universitas Negeri Padang, INDONESIA

*Corresponding Author : satriaapsada44@gmail.com

Received 2023-12-29; Revised 2024-03-25; Accepted 2024-05-17

Abstract

Robot *Automated Guided Vehicle* (AGV) pada umumnya dapat bergerak secara *autonomous* dan manual. Kendali manual robot *Automated Guided Vehicle* (AGV) dibutuhkan ketika kendali *autonomous* sedang tidak digunakan. Kebutuhan untuk kontrol manual pada AGV dapat disebabkan oleh kegagalan sistem otomatis atau untuk situasi di mana operator perlu intervensi langsung untuk keamanan atau tugas yang memerlukan keputusan manusia. Tetapi dalam pengendalian manual robot memiliki nilai deviasi terhadap gerak lurus, sehingga mempengaruhi kinerja robot dan menjadikannya semakin sulit dikendalikan. Tujuan penelitian ini untuk meningkatkan performa sistem kontrol manual robot *Automated Guided Vehicle* (AGV) dengan menggunakan algoritma PID (*Proportional Integral Derivative*) metode Ziegler Nichols I atau metode kurva reaksi yang mampu menghasilkan nilai *tuning* dengan praktis sehingga robot AGV dapat melakukan gerak lurus dengan akurasi yang lebih baik. Penelitian ini terdiri dari tahapan analisis, pengodean dan pengujian robot. Pada penelitian ini menggunakan sensor *rotary encoder* dengan motor DC PG45 dan dikendalikan mikrokontroler Arduino Atmega 2560 serta kontroler *joystick wireless*. Penggunaan kendali manual digabungkan dengan Algoritma PID metode Ziegler-Nichols I dengan nilai $K_p=0,45$; $K_i=0,3$ dan $K_d=0$ untuk Motor 1 dan Motor 2 dengan nilai $K_p=0,5$; $K_i=0,15$ dan $K_d=0$ yang dikombinasikan dengan sensor *rotary encoder*. Hasil pengujian menunjukkan peningkatan kinerja kendali robot AGV dengan nilai ketepatan jarak tempuh dari 3(tiga) meter menjadi 12(dua belas) meter gerak dapat mendekati lintasan lurus.

Keywords: *Automatic Guided Vehicle; Ziegler-Nichols-I; Kontroler PID; Joystick Wireless.*

1. Introduction

Dalam era Revolusi Industri 4.0, perkembangan teknologi mengalami kemajuan pesat, terutama dalam penerapan robotika di industri. Salah satu contohnya adalah penggunaan *Automated Guided Vehicle* (AGV) untuk meningkatkan kecepatan dan ketepatan produksi[1][2]. Penelitian ini bertujuan untuk meningkatkan kinerja pengendalian robot AGV. Robot AGV yang diteliti merupakan bagian dari penelitian sebelumnya, tentang Pemetaan 2D *Indoor* Menggunakan AGV dengan LIDAR[3]. Saat ini, robot pemindah barang sudah ada dengan sistem pengendalian *autonomous*. Tetapi ada saat dimana robot memerlukan kontrol manual. Adanya kebutuhan untuk kontrol manual pada AGV dapat disebabkan oleh beberapa faktor. Pertama, kontrol manual dapat menjadi solusi darurat atau *backup* jika terjadi kegagalan pada sistem otomatis. Kedua, dalam situasi tertentu, operator mungkin memerlukan kontrol langsung untuk memastikan keamanan atau untuk melakukan tugas-tugas yang memerlukan keputusan manusia. Kontrol manual juga dapat diperlukan selama fase perawatan, pengujian dan pengembangan robot. Mengontrol arah gerak robot AGV secara manual, dibutuhkan sebuah alat pengontrol yang nantinya akan digunakan oleh *user* sebagai alat kendali robot AGV. Sistem pengendalian

manual memiliki keterbatasan ruang gerak dikarenakan kurang sesuai bagi robot yang sering bergerak dari satu tempat ke tempat lainnya[4]. Maka, sistem pengontrolan manual ini direncanakan menggunakan *Joystick Wireless*.

Robot dengan penggunaan kontroler *joystick wireless* dipilih karena efektivitasnya tanpa kabel penghubung yang menghambat pergerakan robot AGV[4]. Kontroler *Joystick wireless* terdiri dari dua bagian, yakni modul *transmitter*(pengirim) dan modul *receiver* (penerima). Modul *transmitter* bertugas sebagai *input* data dan mengirimkannya ke modul *receiver*. Sebaliknya, modul *receiver* berperan sebagai penerima data yang dikirim oleh modul *transmitter*[5]. Sistem *controller joystick wireless* dapat berkomunikasi dengan robot dalam jarak 0-10 meter [15]. Kecepatan motor penggerak robot diatur oleh nilai PWM pada awalnya sebagai penentu kecepatan dua roda robot untuk mencapai besar nilai kecepatan rotasi roda (RPM) yang diinginkan atau biasa disebut dengan *setpoint*. Namun, uji coba memiliki kendala, seperti orientasi yang kurang baik dan deviasi gerak lurus robot AGV. Dalam sistem pergerakan robot, terdapat berbagai faktor yang dapat menyebabkan nilai *error* pada *trajectory* menjadi signifikan, salah satunya adalah ketika roda yang digunakan mengalami selip yang dapat mengakibatkan deviasi rute[6]. Oleh karena itu, diperlukan kontrol atau algoritma untuk mengurangi kesalahan pergerakan robot AGV dalam mode manual.

Pengaturan kecepatan motor DC membutuhkan sistem kontrol yang mampu menghasilkan kecepatan yang stabil. Karena motor menunjukkan kurangnya respons yang efektif dalam menjaga ketepatan dan stabilitas sistem mengakibatkan terjadinya kesalahan yang cukup besar[7]. Penggunaan PID (*Proportional Integral Derivative*) metode Ziegler-Nichols I dalam penelitian ini dipilih karena PID merupakan suatu metode kontrol yang telah terbukti efektif[8], metode Ziegler-Nichols I juga sering digunakan pada industri[9]. PID dapat memberikan respons yang cepat dan stabil terhadap perubahan kondisi, serta dapat diatur sesuai dengan karakteristik dinamis AGV. Keunggulan PID meliputi kemampuan untuk mengatasi kesalahan sistem dan menjaga kestabilan operasional, membuatnya menjadi pilihan yang sesuai untuk mengontrol kecepatan AGV dalam mode manual[10].

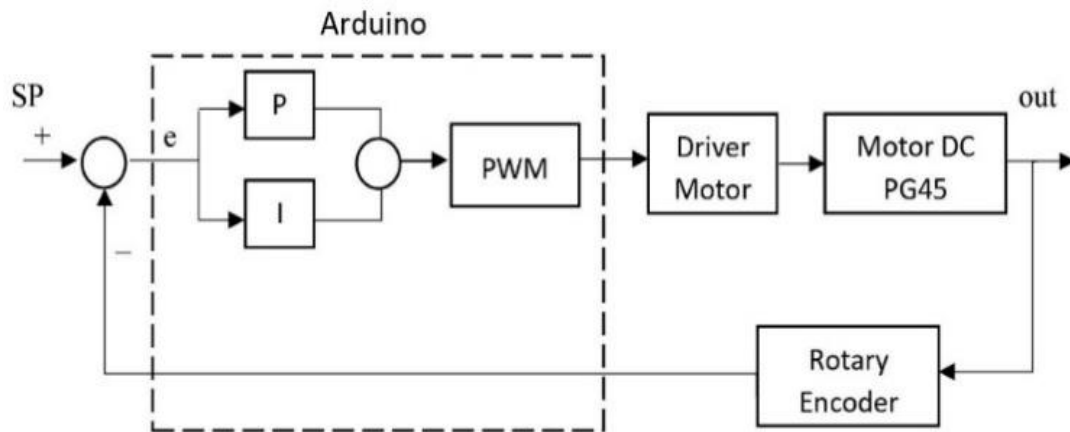
Desain robot yang sesuai menjadi penting untuk memastikan AGV dapat bergerak bebas dan dapat bergerak pada jalurnya. Sistem kendali PID diterapkan untuk mengatur kecepatan motor DC secara stabil, mengatasi kesalahan sistem, dan menjaga kestabilan operasional AGV. Perancangan AGV dengan sistem pergerakan *differential drive*, menggunakan dua roda, memberikan fleksibilitas dan manuverabilitas tinggi. Kemampuan putar dua roda independen memungkinkan AGV bergerak dengan mudah di ruang terbatas pabrik. Sistem roda *differential drive* juga dianggap sederhana dan ekonomis[11]. Dengan menerapkan kontrol PID dan kendali *wireless*, diharapkan akurasi pergerakan dan kemudahan pengontrolan AGV dapat meningkat, mendukung efektivitas produksi industri di era Revolusi Industri 4.0.

2. Material and Methods

Penelitian ini dilakukan dengan metode eksperimental. menggunakan dua motor yang terletak pada sisi kiri dan kanan secara sejajar pada sumbu penggerak *Differential Drive*. Dalam penggerak tersebut digunakan motor DC PG45 dengan sensor *rotary encoder* 7ppr. Sensor *rotary encoder* akan membaca nilai putaran motor penggerak robot yang diatur oleh PWM dan dari besar nilai pembacaan sensor *rotary encoder* ini menjadi acuan apakah motor penggerak roda robot berputar sama atau seimbang.

Kendali PID dibutuhkan untuk membuat kedua roda robot dapat memiliki kecepatan yang sama atau stabil dengan cara pemrograman dan diproses oleh Arduino Mega2560 sebagai mikrokontroler robot. Untuk membuat robot dapat bergerak maju dengan baik dan menghasilkan nilai deviasi yang rendah. Pada saat bergerak lurus maju terlihat penyimpangan gerak lurus robot terhadap lintasan lurus, maka penyimpangan itulah yang menentukan besar standar deviasi robot. Pengendali gerakan diuji beberapa kali dan menganalisis hasilnya. Pengujian jarak yang ditempuh oleh robot dilakukan dengan

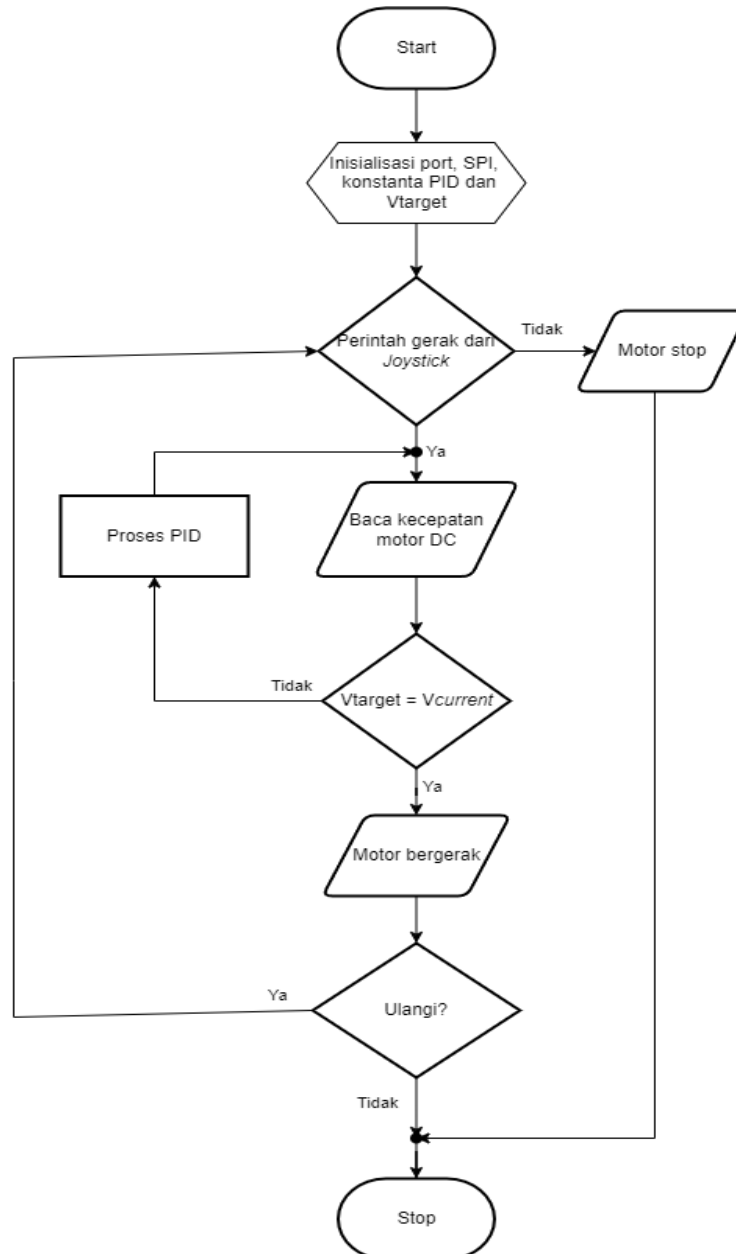
membandingkan rencana jarak tempuh dengan jarak tempuh sebenarnya. Dari data standar deviasi yang didapatkan menjadi acuan penilaian keberhasilan penelitian apabila pergerakan robot dapat bergerak dengan baik. Selain itu, evaluasi kesesuaian pergerakan robot dengan lintasan yang diinginkan juga menjadi faktor kunci dalam menentukan efektivitas dan akurasi sistem navigasi yang dikembangkan.



Gambar 1: Blok Diagram Kontroler PID

Pada tahap Material dan Metode penelitian, terdapat pemantapan mengenai blok diagram, blok diagram adalah alat visual yang dipakai untuk menggambarkan proses kerja dalam bidang ilmu rekayasa atau teknik. Dari Gambar 1, dapat disimpulkan fungsi dari masing-masing blok diagram sebagai berikut:

- Setpoint*, merupakan nilai yang ditetapkan sebagai acuan untuk kontrol variabel. Pada penelitian ini nilai *setpoint* yaitu nilai RPM yang diinginkan dan terinisialisasi pada arduino. Rentang nilai tersebut antara 0-400 RPM.
- Arduino mega2560, merupakan perangkat mikrokontroler *open-source* yang mudah digunakan[12], Arduino mega2560 berfungsi sebagai mikrokontroler untuk memproses dan menentukan *setpoint*. *Pin* yang digunakan dalam Arduino tersebut yaitu *pin* digital yang mana diantaranya difungsikan sebagai *interrupt* dan *pin* PWM. Didalam Arduino juga dilakukan proses kontrol untuk membandingkan nilai *rotary encoder* dan *setpoint* dengan metode PID dan diteruskan dengan mengatur tegangan *driver* motor oleh PWM, sehingga menghasilkan *output* kontroler.
- Driver Motor* BTS7960, merupakan sebuah modul *driver* yang memiliki fungsi krusial sebagai penerjemah data dari Arduino, memungkinkan penggunaannya untuk mengatur kecepatan motor dengan presisi dan efisiensi tinggi[13]. Tegangan dari sinyal PWM yang diatur oleh Arduino dialirkan ke modul *driver* motor BTS7960, yang berfungsi sebagai penerjemah data untuk mengatur kecepatan motor. Modul *driver* motor memungkinkan pengaturan kecepatan motor secara efisien dan memungkinkan kontrol yang presisi terhadap kinerja motor.
- Motor DC PG45, pada penelitian ini berfungsi sebagai penggerak dua roda robot AGV dengan tegangan motor maksimal 24VDC. Sensor *Rotary Encoder* juga ter-include pada motor ini. Motor PG45 merupakan objek yang dikendalikan dalam sistem kendali *close loop* penggerak dua roda *Differential Drive* pada robot AGV.
- Rotary Encoder* merupakan sensor yang difungsikan untuk mendeteksi kecepatan motor dalam satuan RPM.

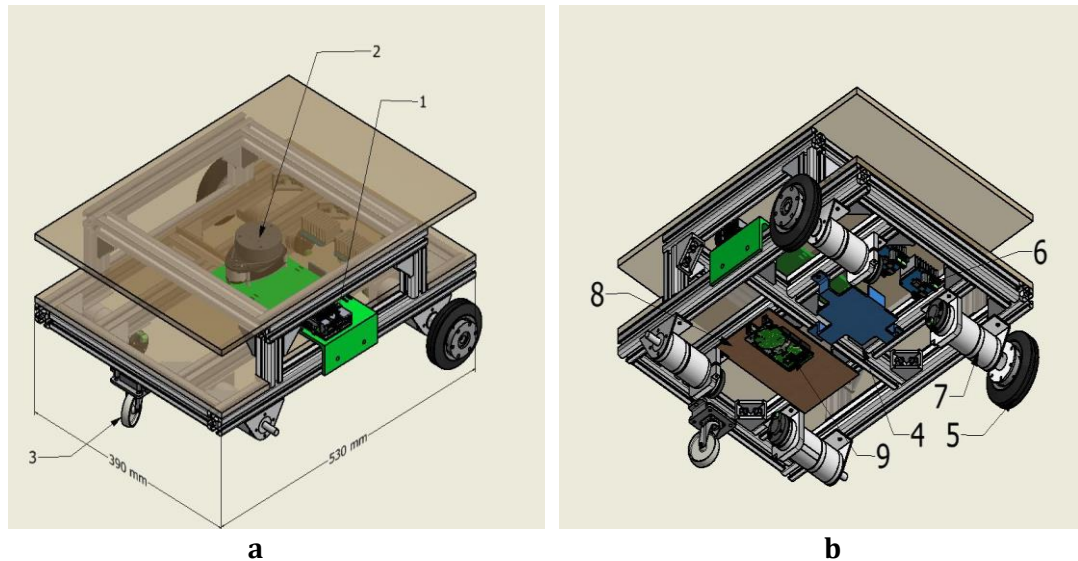


Gambar 2: Flowchart Sistem Kontrol Robot AGV

Flowchart terdiri dari simbol-simbol yang mencerminkan langkah-langkah fungsi program, dengan garis alur yang menunjukkan urutan eksekusi, simbol-simbol *flowchart* memiliki arti tersendiri[14]. Berdasarkan *flowchart* pada Gambar 2, diawali oleh inisialisasi parameter PI V_{target} dan *port*, lalu robot digerakkan melalui yang memberikan sinyal digital yang ditangkap *receiver*. Melalui *receiver* Arduino membaca data tersebut lalu dilakukan pemilahan keputusan untuk pergerakan robot maju, mundur, arah kiri dan arah kanan. Keputusan tersebut akan di proses melalui kontrol PI untuk mendapatkan kecepatan setiap motor berdasarkan arah pergerakan. Proses tersebut dihasilkan nilai dengan metode PWM untuk men-*trigger* motor dan *Encoder* memberikan *feedback* kepada kontrol PI untuk mencapai V_{target} yg telah di inisialisasi. Jika tidak ada perintah yang diberikan oleh *Joystick* maka robot akan diam.

Penelitian ini menggunakan robot AGV dari penelitian sebelumnya yang berdimensi 539 x 390 mm yang berbahan dasar aluminium profil (Gambar 4). Sebagian besar robot terbuat dari bahan aluminium mulai dari *frame*, *bracket* motor dan *hub fixed wheel*. *Frame* robot berbahan aluminium profil 30 x 30 mm dan setiap dasar tingkat robot dilapisi bahan akrilik 10 mm. Robot yang dibangun ini memiliki berat total (mekanik dan komponen

elektronik) sebesar 16.4 kg. Seluruh berat robot ditopang oleh 3 roda, satu roda *free wheel* dan dua roda tetap dengan jenis kinematika *differential drive*. Pentunjuk komponen dapat dilihat pada Tabel 1.



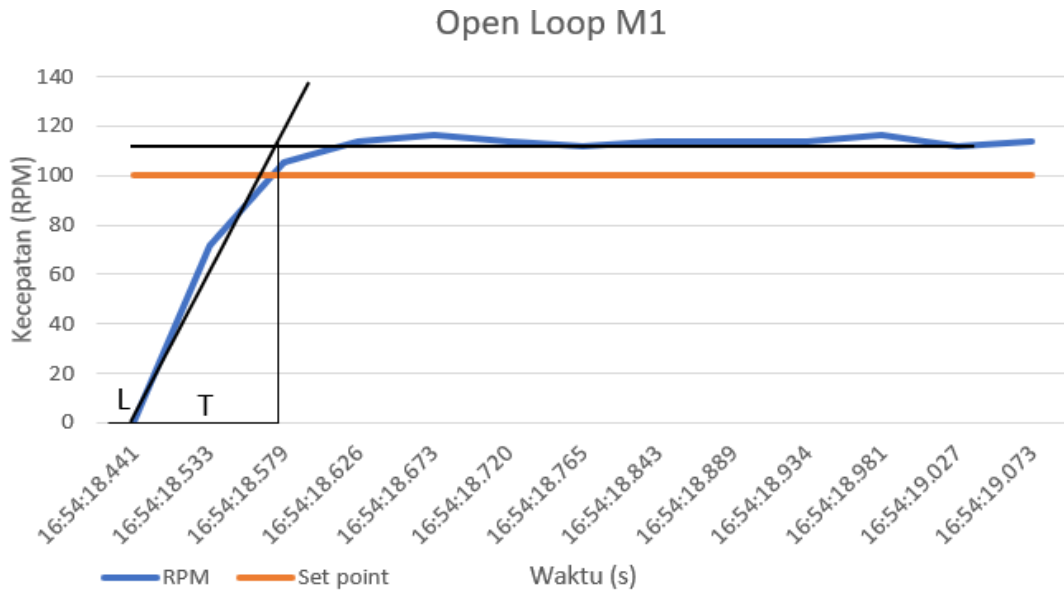
Gambar 3: (a) Tampak Atas, (b) Tampak Bawah Robot AGV
Tabel 1: Komponen Robot

Kode	Nama	Penjelasan
1	Mini Pc RasperyPi	Mikrokontroler untuk Lidar
2	Lidar	Sensor pendeteksi jarak
3	Roda <i>Free Wheel</i>	Roda depan yang bergerak bebas
4	Penyimpanan Baterai	Penyimpanan baterai pada robot
5	Roda <i>Fixed Stering Wheel</i>	Roda penggerak utama robot
6	Motor Driver	Pengendali kecepatan motor dc
7	Motor DC PG45	Penggerak utama robot
8	Papan Rangkaian	Jalur penghubung terminal komponen
9	Arduino mega 2560	Mikrokontroler pengendali utama robot

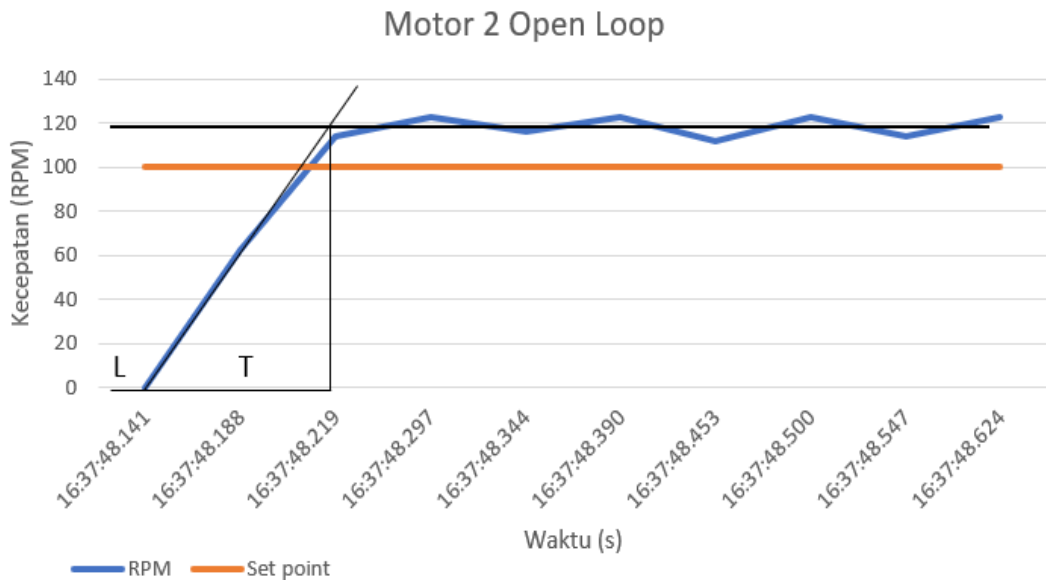
3. Results and Discussion

Setelah menyelesaikan perancangan alat, langkah berikutnya adalah melakukan pengujian. Dalam tahap ini, alat akan diuji untuk mengevaluasi sejauh mana kinerja sistem dapat berjalan dengan efektif. Pengujian dimulai dari, *tuning* PID, pengujian nilai PWM hingga pengujian gerak robot.

Tuning PID adalah proses penalaan untuk menentukan konstanta PID (KP, KI, dan KD) dengan metode *Ziegler-Nichols*. Proses *tuning* dilakukan dengan menerapkan posisi garis singgung terhadap kurva tanggapan *plant* untuk memperoleh nilai konstanta waktu PID yaitu waktu tunda (L) dan waktu konstan (T). Proses *tuning* dilakukan dengan cara membuat sistem *open loop* terlebih dahulu. Kemudian, diberikan *step* respons untuk melihat respon dari Motor 1 dan Motor 2 robot AGV.



Gambar 4: Respon Sistem *Open Loop* Motor 1



Gambar 5: Respon Sistem *Open Loop* Motor 2

Gambar 4 dan Gambar 5 menunjukkan sistem dalam keadaan *open loop* pada motor 1 dan motor 2, sehingga respon langkah yang diberikan pada titik *setpoint* 100 RPM menghasilkan nilai waktu tunda konstan (*Lag Time* - L) dan waktu konstan (*Time Constant* - T) seperti yang tercatat dalam Tabel 2.

Tabel 2: Hasil Penelaan Berdasarkan Metode Ziegler-Nichols

Motor	<i>Lag Time</i> (L)	<i>Time Constant</i> (T)
Motor 1	0.092	0.046
Motor 2	0.047	0.031

Setelah mendapatkan nilai konstanta L dan T, langkah berikutnya adalah memasukkan nilai-nilai tersebut ke dalam tabel penalaan metode *Ziegler-Nichols* I, sebagaimana terlihat dalam Tabel 3 untuk motor 1 dan Tabel 4 untuk motor 2.

© The Author(s)
Published by Universitas Negeri Padang
This is an open-access article under the: <https://creativecommons.org/licenses/by/4.0/>

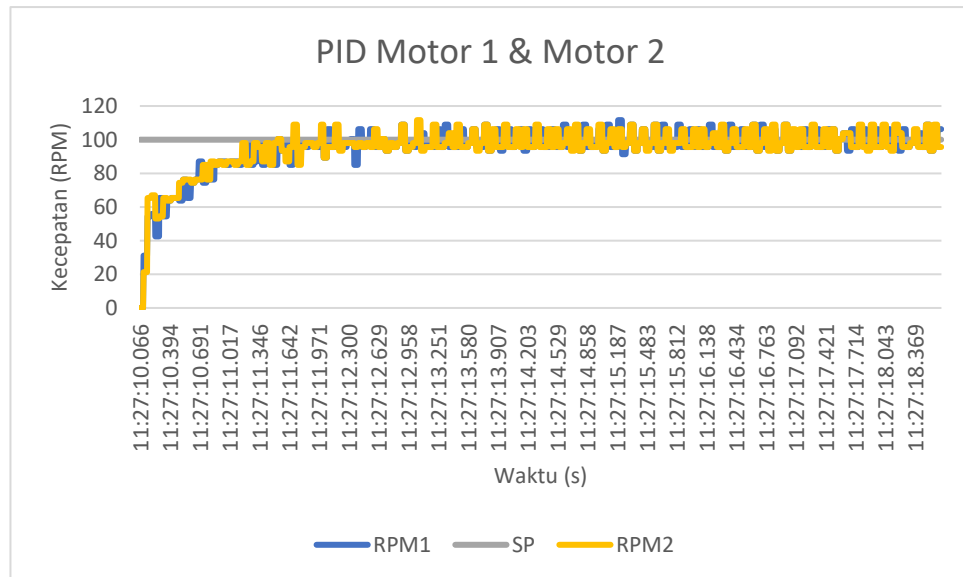
Tabel 3: Akumulasi Parameter PID Motor 1

Kontrol	Kp	Ti	Td
P	0,5	-	0
PI	0,45	0,3	0
PID	0,6	0,3	0,184

Tabel 4: Akumulasi Parameter PID Motor 2

Kontrol	Kp	Ti	Td
P	0,6	-	0
PI	0,5	0,15	0
PID	0,7	0,15	0,09

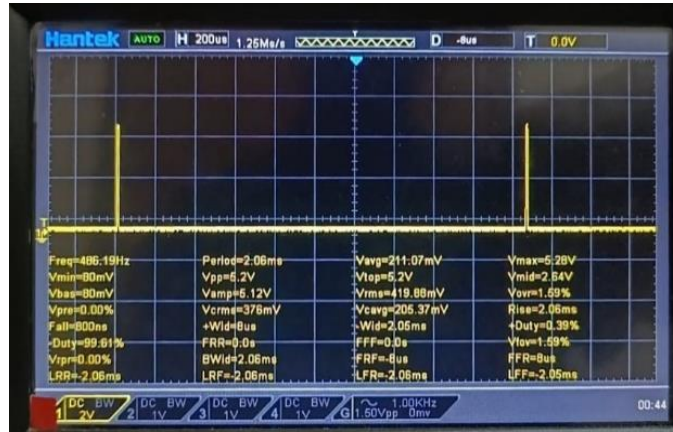
Pengujian sistem ini dilakukan dengan cara memberikan nilai *setpoint* 100 RPM, hal ini dilakukan dengan tujuan untuk melihat nilai *error steady state* pada saat *real time*, dengan kondisi nilai Kp dan Ki nya sudah dimasukkan pada sistem. Pengujian dilakukan dengan cara meng-*input* hasil perhitungan PI pada tabel Ziegler-Nichols metode I kedalam program Arduino IDE untuk diproses oleh *mikrokontroler* Arduino Mega2560. Setelah program di *upload* selanjutnya dilakukan ujicoba motor dengan sensor *Encoder*. Bentuk grafik respon sistem dapat dilihat pada Gambar 6.



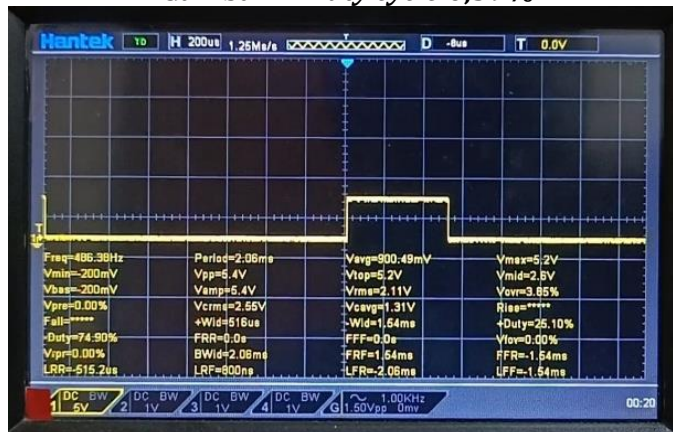
Gambar 6: Respon Sistem PID Pada Motor 1 dan Motor 2

Pulse Width Modulation (PWM) adalah teknik untuk mengatur lebar pulsa dalam satu periode sinyal pulsa, menghasilkan tingkat tegangan rata-rata yang dapat diubah-ubah. Ini memungkinkan kontrol daya yang efektif dengan memvariasikan siklus tugas antara 0% dan 100%, sementara frekuensi gelombang tetap konstan[16]. Pengujian nilai PWM dilakukan untuk melihat bentuk *duty cycle* dilakukan guna mengetahui besaran arus keluaran pada *driver* motor sebagai sumber daya gerak motor DC. Hasil pengujian besaran nilai *duty cycle* dapat dilihat pada Gambar 7, Gambar 8, Gambar 9, Gambar 10 dan Gambar 11.

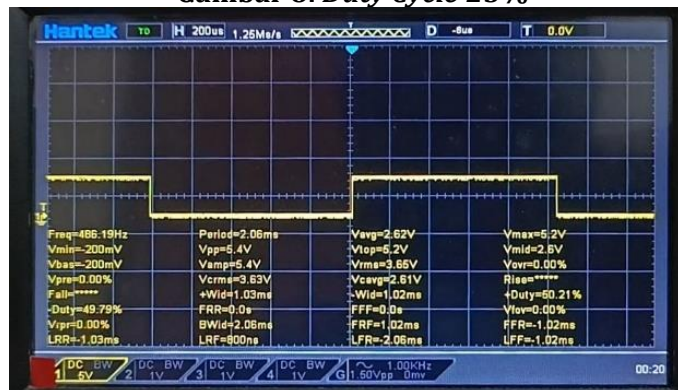
© The Author(s)
Published by Universitas Negeri Padang
This is an open-access article under the: <https://creativecommons.org/licenses/by/4.0/>



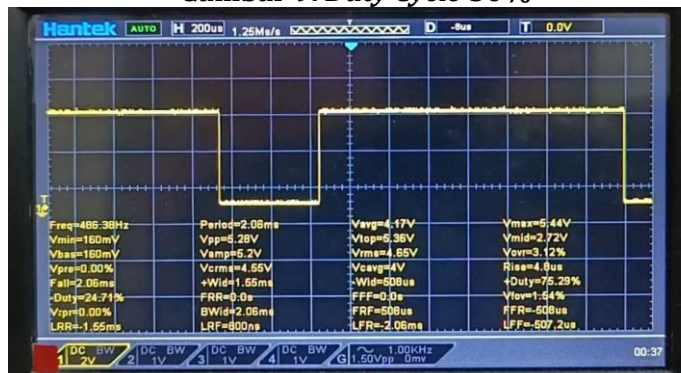
Gambar 7: Duty cycle 0,39%



Gambar 8: Duty Cycle 25%



Gambar 9: Duty Cycle 50%



Gambar 10: Duty Cycle 75%

© The Author(s)
Published by Universitas Negeri Padang
This is an open-access article under the: <https://creativecommons.org/licenses/by/4.0/>



Gambar 11: Duty Cycle 95%

Dengan referensi ke Gambar 7, Gambar 8, Gambar 9, Gambar 10 dan Gambar 11, dapat diperoleh nilai variasi *duty cycle* yang sesuai dengan besar variasi nilai PWM. Hubungan nilai PWM dan tegangan *output* yang diteruskan dari *driver* motor ke motor penggerak dapat dilihat pada Tabel 5.

Tabel 5: Hubungan antara PWM, Duty Cycle dan Tegangan Motor

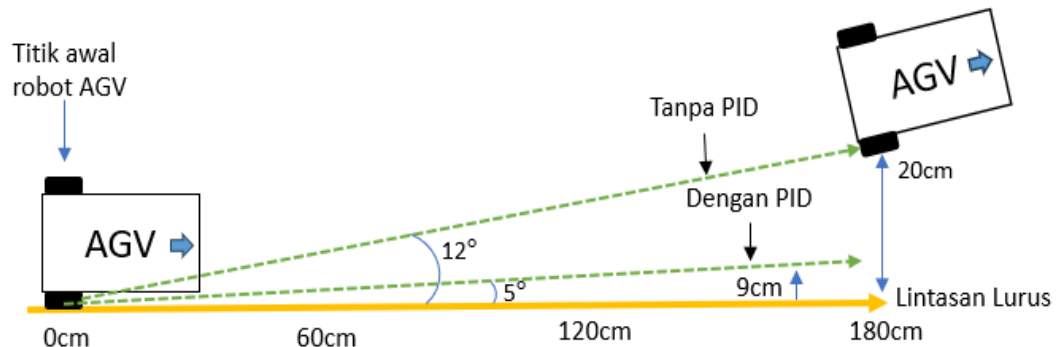
PWM	V_{in}	Duty Cycle	V_{out}
0	24V	0,39%	0V
56	24V	25,1%	6V
112,5	24V	50,21%	12V
168	24V	75,29%	18V
225	24V	95,68%	22,3V

Bentuk gelombang dapat berubah-ubah sesuai besar nilai PWM. Semakin besar nilai PWM semakin besar pula nilai *duty cycle*-nya mengakibatkan semakin cepat putaran motor DC. Sebagaimana, nilai kecepatan motor yang terdeteksi oleh sensor *encoder* diproses oleh PID dan di konversi oleh PWM menjadi tegangan yang sesuai untuk menyesuaikan *setpoint* kecepatan motor oleh *driver* motor. Pengujian PWM berhasil menghasilkan variasi *duty cycle* yang sesuai dengan variasi nilai PWM. Hal ini menunjukkan kemampuan sistem dalam mengontrol kecepatan motor. Selanjutnya diuji kecepatan dua motor penggerak robot AGV dengan *setting*-an yang sama sebagai awalan dari proses *tuning* PID untuk membandingkan pergerakan motor tanpa PID dan dengan menggunakan PID nantinya.

Dalam Penelitian ini, robot diujikan bergerak maju sepanjang lorong. Tujuan pengujian ini untuk melihat apakah robot bisa bergerak lurus maju dengan stabil menggunakan roda *differentiation drive*. Robot dengan penggerak diferensial adalah robot yang memanfaatkan dua penggerak yang terpasang pada sisi kiri dan sisi kanan dengan sejajar dan dapat bergerak secara mandiri. Setiap roda dapat dikontrol untuk bergerak dengan kecepatan dan arah putar yang berbeda, memungkinkan robot bergerak baik lurus maupun dengan jalur melingkar. Saat kedua roda penggerak dengan kecepatan yang sama, robot akan bergerak secara sejajar atau lurus. Namun, apabila pergerakan roda tidak sama robot akan bergerak membentuk sebuah kurva [17]. Maka menggunakan metode PID diharapkan kecepatan motor penggerak robot bisa sama dan lebih stabil untuk mencapai tujuan pergerakan lurus robot yang baik.

Pengujian menggunakan metode standar deviasi untuk mengevaluasi variasi pergerakan robot saat bergerak lurus maju dari titik awal terhadap lintasan lurus. Penyimpangan gerak lurus robot diukur untuk menentukan standar deviasi. Jarak yang ditempuh oleh robot dibandingkan dengan rencana jarak, diuji tiga kali dengan pengaturan kecepatan menggunakan PWM motor dan tiga kali dengan kontrol PID bersama sensor *encoder*. Pada Gambar 13, ujicoba standar deviasi dilakukan untuk memperbandingkan

pergerakan robot AGV dengan dan tanpa penggunaan PID. Sebelum menjalankan robot pada jarak yang lebih jauh, uji coba dilakukan terlebih dahulu pada jarak yang lebih pendek sebesar 180 cm untuk mengevaluasi besarnya standar deviasi pergerakan robot. Bentuk deviasi gerak robot AGV diilustrasikan pada Gambar 12.



Gambar 12: Deviasi Gerak Robot AGV

Dengan parameter PID yang disesuaikan secara otomatis, grafik pergerakan robot AGV pada Gambar 12 menunjukkan konsistensi dan presisi yang ditingkatkan. Setelah dilakukan beberapa kali ujicoba gerak lurus dan dilakukan pengukuran didapati besaran nilai standar deviasi seperti yang terdapat pada tabel sebagai berikut:

Tabel 6: Hasil Pengujian Jarak Tempuh

Jarak Tempuh (cm)	Teknik	Hasil jarak tempuh (cm)			Rata-rata (cm)	Standar Deviasi (cm)
		tes 1	tes 2	tes 3		
60	Set pwm	61	61	60,8	60,8	0,56
120	Set pwm	123	121,5	121,8	121,8	1,27
180	Set pwm	184	183	182	183	2,12
60	PID + Encoder	60	60,5	60,5	60,3	0,21
120	PID + Encoder	120,5	120	120,5	120,6	0,42
180	PID + Encoder	181	181,5	181	181,1	0,77

Dari Tabel 6, dapat dilihat bahwa semakin besar jarak tempuh robot, semakin besar perbedaan antara rata-rata jarak tempuh yang direncanakan dan yang benar-benar terjadi. Selain itu, nilai standar deviasi juga meningkat seiring dengan peningkatan jarak, menunjukkan bahwa hasil pengujian semakin bervariasi ketika jaraknya lebih jauh. Seperti pada hasil uji coba pada jarak 180 cm memiliki kesalahan sebesar 4 cm dengan nilai deviasi tertinggi sebesar 2,12.

Algoritma PID digunakan untuk mengoreksi posisi robot berdasarkan *error*, memastikan stabilitas putaran motor, dengan selisih kesalahan maksimal 1,5 cm dan standar deviasi 0,77 pada jarak tempuh 180 cm. Sebaliknya, teknik set PWM menunjukkan selisih kesalahan maksimal 3 cm dan standar deviasi 2,12 pada jarak yang sama. Pengujian kelurusan gerakan robot dilakukan dalam tiga percobaan berbeda dengan variasi jarak tempuh, menunjukkan hasil dalam Tabel 7 dan Gambar 12.

Tabel 7: Hasil Pengujian Penyimpangan

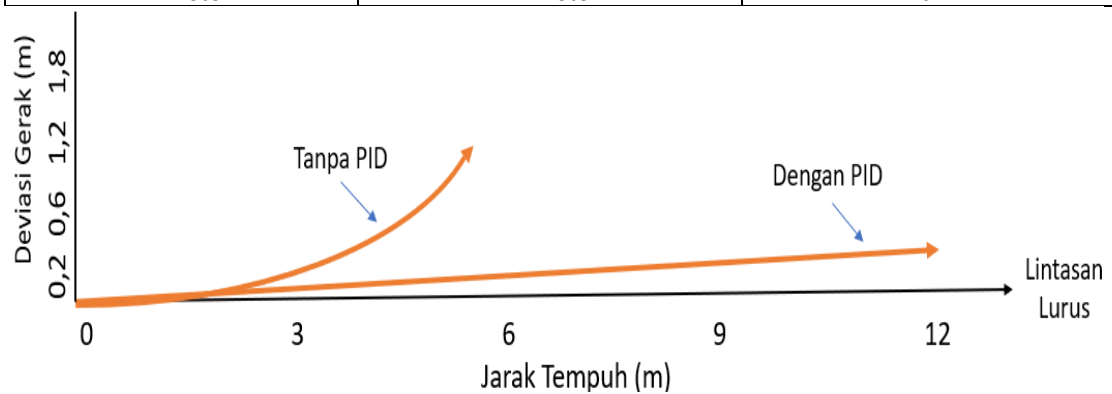
Jarak Tempuh (cm)	Teknik	Penyimpangan dari garis lurus (cm)			Rata-rata (cm)	Standar Deviasi (cm)
		tes 1	tes 2	tes 3		
60	Set pwm	2	2,5	5	3,1	1,47
120	Set pwm	8,3	4	9	7,1	2,41
180	Set pwm	20	13	11,5	14,8	3,72

60	PID + Encoder	0	0,2	1	0,4	0,43
120	PID + Encoder	2,5	2	5	3,1	1,31
180	PID + Encoder	7	6	9	7,3	1,24

Dari Tabel 7 dan Gambar 12, dapat diamati bahwa semakin jauh jarak yang ditempuh oleh robot tanpa PID, posisi robot cenderung melenceng dari lintasan lurus tujuan. Selisih posisi garis lurus mencapai 20 cm dengan standar deviasi tertinggi 3,72. Penerapan Algoritma PID menghasilkan pergerakan robot yang konsisten, baik pada jarak tempuh jauh maupun dekat, dengan perbedaan posisi dari garis lurus terbesar hanya 9 cm dan standar deviasi terbesar 1,31. Algoritma PID sukses menghasilkan gerak lurus robot yang lebih stabil, dengan melakukan koreksi pada setiap nilai putaran roda yang tidak tepat, sehingga mencapai nilai pergerakan yang sama pada kedua motor. Ujicoba gerak lurus yang lebih panjang juga dilakukan untuk mengkonfirmasi keberhasilan robot mencapai pergerakan lurus yang optimal. Ilustrasi pergerakan lurus robot AGV dapat dilihat pada Gambar 13.

Tabel 8: Respon Deviasi pada Gerak Lurus Robot AGV

Panjang lintasan	Deviasi gerak lurus	Akurasi
12 Meter	>2 Meter	Kurang
12 Meter	1-2 Meter	Cukup
12 Meter	<1 Meter	Baik



Gambar 13: Pergerakan Aktual Robot AGV

Berdasarkan hasil pengujian, terlihat bahwa penggunaan kontroler PID secara signifikan meningkatkan stabilitas pergerakan robot AGV. Grafik pergerakan robot menunjukkan bahwa posisinya mendekati lintasan lurus dengan deviasi hanya sebesar 0,6 Meter pada titik 12 Meter, mengindikasikan performa yang baik. Selain itu, hasil pengujian menegaskan bahwa metode Ziegler-Nichols I berhasil diterapkan dalam parameter PID untuk robot AGV, memberikan keyakinan dalam efektivitas dan aplikabilitas pengaturan kontroler untuk mencapai kinerja yang optimal.

4. Conclusion

Berdasarkan hasil penelitian, penerapan pemrograman PID pada robot *Automatic Guided Vehicle* (AGV) mampu mengatasi ketidakseimbangan kecepatan antara Motor 1 dan Motor 2, serta mengurangi deviasi terhadap lintasan lurus. Pengujian kinerja sistem menggunakan kontroler PID beserta sensor *Rotary Encoder* telah berhasil diimplementasikan, dengan parameter PID yang ditemukan untuk Motor 1 adalah $K_p=0,45$, $T_i=0,3$, dan $K_i=1,4$, sedangkan untuk Motor 2 adalah $K_p=0,5$, $T_i=0,15$, dan $K_i=3,1$. Hal ini terbukti melalui respons yang stabil dan kecepatan yang sama pada kedua roda, sehingga memungkinkan pergerakan gerak lurus robot yang optimal dengan deviasi yang minimal, yaitu sekitar 0,6 Meter pada jarak 12 Meter, tidak lebih dari satu meter. Dengan demikian, implementasi PID pada AGV telah membuktikan efektivitasnya dalam meningkatkan kinerja sistem dan memperbaiki akurasi pergerakan robot.

References

- [1] D. Clayton, "Automated Guided Vehicles,," pp. 257–262, 1983, doi: 10.4018/978-1-5225-3064-0.ch002.
- [2] A. A. Wardani and Z. Iklima, "Rancang Bangun Automated Guided Vehicle Menggunakan Metode PID Zeigler Nichols," *J. Teknol. Elektro*, vol. 13, no. 1, p. 6, 2022, doi: 10.22441/jte.2022.v13i1.002.
- [3] T. Kurniawan, "Pemetaan 2D Indoor menggunakan Automated Guided Vehicle (AGV) dengan LiDAR," vol. 4, no. 1, pp. 169–178, 2023.
- [4] U. Sutisna and Zaenurrohman, "Perancangan Sistem Kontrol Wireless pada Mobile Robot Manipulator Berbasis Mikrokontroler ATmega8," *Jnteti*, vol. 3, no. 1, pp. 69–75, 2014.
- [5] H. T. Saputra and A. Muhaimin, "Robot Pemindah Benda dengan Kendali Joystick PS2 Wireless Berbasis Wemos," *J. Ilmu Komput.*, vol. 11, no. 2, pp. 80–85, 2022.
- [6] G. W. Kurniawan, N. Setyawan, and E. A. Hakim, "PID Trajectory Tracking Control 4 Omni-Wheel Robot," *SinarFe7*, vol. 2, no. 1, pp. 345–350, 2019.
- [7] M. S. Zuhrie, P. W. Rusimamto, and N. Kholis, "Rancang Bangun PID Controller Dengan Tuning Ziegler Nicholas Untuk Pengendalian Posisi Sudut Motor DC," *J. Tek. Elektro*, vol. 10, no. 2, pp. 537–545, 2021.
- [8] T. Pujiati and R. Risfendra, "Penerapan Kontroler PID Pada Sistem Kendali Level cairan Dengan Metode Ziegler-Nichols Berbasis Arduino," *JTEIN J. Tek. Elektro Indones.*, vol. 2, no. 1, pp. 55–60, 2021, doi: 10.24036/jtein.v2i1.123.
- [9] K. Ogata, *Modern Control Engineering [Paperback]*. 2009.
- [10] E. H. Muktafin, K. Kusriani, and E. T. Luthfi, "Analisis Sistem Kendali Robot USMAN untuk Sterilisasi Lantai Masjid dengan Algoritma Proportional Integral Derivative," *J. Eksplora Inform.*, vol. 10, no. 2, pp. 80–91, 2021, doi: 10.30864/eksplora.v10i2.468.
- [11] R. Ria, "Penerapan Inverse Kinematics pada Pengendalian Gerak Robot Lego," *J. Appl. Electr. Eng.*, vol. 2, no. 1, pp. 1–5, 2018, doi: 10.30871/jae.v2i1.1075.
- [12] J. Arifin, L. N. Zulita, and H. Hermawansyah, "Perancangan Murottal Otomatis Menggunakan Mikrokontroler Arduino Mega 2560," *J. Media Infotama*, vol. 12, no. 1, pp. 89–98, 2016, doi: 10.37676/jmi.v12i1.276.
- [13] Y. D. Widiarto, M. E. I. Najoan, and M. D. Putro, "Sistem Penggerak Robot Beroda Vacuum Cleaner Berbasis Mini Computer Raspberry pi," *J. Tek. Elektro dan Komput.*, vol. 7, no. 1, pp. 25–32, 2018.
- [14] Indrajani, "Pengertian Flowchart," *It.Jurnal.Com*, pp. 5–22, 2011.
- [15] S. Nugraha, T. A. Putra, and T. Suhendra, "106153-34885-1-Pb," *Sist. Kendali Navig. Robot Man.*, no. 1, pp. 91–95, 2019.
- [16] Royan and L. A, "Aplikasi Motor Dc-Shunt Untuk Laboratory Shaker Menggunakan Metode Pwm (Pulse Widthmodulation) Berbasis Mikrokontroler Atmega 32," *AMedia Elektr.*, vol. 8, no. 1, pp. 32–50, 2015, [Online]. Available: <http://repository.unimus.ac.id/2905/>
- [17] E. Wahyuni, E. Ningrum, and R. Hakkun, "Rancang Bangun Program Visualisasi Pergerakan Differential Drive Mobile Robot," *J. Tek. Inform.*, 2010.