

Sistem Kontrol pada AMR (Autonomous Mobile Robot)

Handri Toar^{1*}, Elga Viona Sihombing¹, Rikky Septiawan¹, Fajar Nasrialdi¹, Senanjung Prayoga¹, and Fitriyanti Nakul¹

¹Politeknik Negeri Batam, Batam, Indonesia

*Email: toar@polibatam.ac.id

Abstract— Autonomous mobile robot adalah robot yang mampu bergerak secara mandiri (tidak selalu membutuhkan perintah). Sistem kontrol menjadi bagian terpenting agar autonomous mobile robot dapat bergerak sesuai tujuan yang sudah dirancang secara mandiri. Dalam penelitian ini diterapkan sistem kontrol pada autonomous mobile robot dengan metode web app, PID controller, obstacle avoidance. Sistem kontrol ini dirancang agar autonomous mobile robot mampu memonitoring dan mengontrol pergerakan robot, mengatur kecepatan pada robot, dan mendeteksi objek atau halangan yang berada di sekitar robot. Hasil pengujian menunjukkan bahwa sistem kontrol mampu mengatur gerak autonomous mobile robot dalam mencapai posisi tujuan dengan nilai paramater PID adalah $K_p= 0.2$, $K_d= 0.065$, dan $K_i= 0.02$. Untuk mendeteksi dan menghindari objek dengan nilai error jarak sensor ke objek minimal $2\text{ cm}= 0.01$ dan nilai maksimal $20\text{ cm}= 0.02$. Data yang dihasilkan pada web app berupa data kordinat dan data hasil data waktu terlama bernilai 2.99 detik dan waktu tercepat bernilai 0.68 detik.

Keyword: *Autonomous Mobile Robot, Web App, PID Controller, Sensor Ultrasonik.*

I. PENDAHULUAN

Perkembangan mengacu pada perkembangan suatu benda dari lokasi pertama ke lokasi terakhirnya. Dalam pergerakan yang dapat disertifikasi, ada dua bagian penting: jalur dan jarak. Jalur mengacu pada arah suatu benda akan mengikuti, sedangkan jarak menunjukkan perkembangan sebuah benda dari tahap awal hingga titik penyempurnaan. Robot akan menghadapi berbagai jenis gerakan, termasuk gerakan lurus, bengkok, dan bulat. Roda mecanum merupakan bagian utama pada robot yang memudahkan dalam menggerakkan roda [1]. Otonom Mobile Robot (AMR) memiliki sedikit atau tidak ada

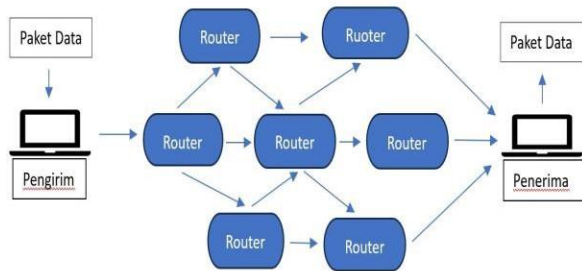
campur tangan manusia pergerakannya dan dirancang untuk berpindah dari tahap awal ke titik tujuan secara mandiri sangat penting bagi robot AMR. Untuk memenuhi hal ini, robot harus dapat mengidentifikasi di sekitarnya dan mengetahui lokasinya tersebut [2]. Untuk mendeteksi lingkungan dan lokasi, *Autonomous Mobile Robot* (AMR) menggunakan sistem kontrol sebagai salah satu alat yang dapat menunjang kompetensi mahasiswa tentang dunia robotika. Pada saat yang sama, jalur yang dihasilkan oleh perencana harus mengikuti batasan yang diberlakukan. Hal ini mungkin disebabkan oleh keterbatasan kemampuan adaptasi robot pada medan tertentu [3]. Selain itu, metode yang digunakan pada robot ini menggunakan invers kinematika yang berfungsi agar pada saat berjalan robot tetap stabil dan metode PID membantu robot saat berjalan dapat diatur kecepatannya, di samping itu sensor ultrasonik membantu agar ketika robot ada halangan di depan maka robot tersebut dapat menghindarinya. Hal ini juga diperjelas dengan mengikuti metode ini, fakta bahwa karya-karya sebelumnya memiliki kekurangan pada saat robot berjalan [4]. Tujuan dari pembuatan penelitian ini adalah untuk mengetahui bagaimana robot ini dapat berjalan sesuai jalur yang sudah dirancang.

II. DASAR TEORI

A. TCP/IP

TCP merupakan protokol yang terletak dilayer transport. Protokol ini menyediakan service yang dikenal sebagai connection oriented, reliable, byte stear service [5]. TCP dan IP adalah konvensi standar yang dimaksudkan untuk melengkapi korespondensi informasi dalam organisasi web. TCP/IP terdiri dari sekumpulan konvensi yang bertanggung jawab atas pelaksanaan eksplisit dalam korespondensi informasi. Dengan aturan ini, pelaksanaan konvensi menjadi jelas dan mendasar, membuatnya mudah untuk dilaksanakan di seluruh peralatan dan program dalam organisasi dan juga mudah untuk menyelesaikan proses penyelidikan dan pemeliharaan. TCP jugak protokol yang bertanggung jawab

untuk pengiriman data yang handal dan teratur. Protokol ini memastikan bahwa data yang dikirimkan dari satu titik ke titik lain di internet tiba dengan benar, dalam urutan yang tepat, dan tanpa kehilangan atau duplikasi. Untuk IP sendiri IP adalah protokol yang bertanggung jawab untuk pengalaman dan routing paket data di jaringan. Setiap perangkat yang terhubung ke Internet diberikan alamat IP yang unik, yang digunakan untuk mengidentifikasi perangkat tersebut di jaringan. IP juga mengatur pengiriman paket data melalui jaringan dengan menggunakan alamat IP tujuan dan informasi routing [6]. Penjelasan cara kerja pada TCP/IP dapat dilihat pada gambar 1 di bawah ini.



Gambar 1. Cara Kerja TCP/IP

B. Invers Kinematik

Model kinematika merepresentasikan hubungan end effector dalam ruang tiga dimensi dengan variabel sendi dalam ruang sendi [7]. Pada bagian ini robot beroda empat mecamum dengan struktur simetris sebagai objek penelitian, seperti terlihat pada gambar 2, dan dibahas hubungan koordinasi gerak masing-masing roda. Memilih pusat simetris geometris sebagai asal sistem O, sistem koordinat persegi panjang XOY yang diperbaiki dengan platform bergerak ditetapkan. Sistem koordinat roda mecamum $X_{wi}O_{wi}Y_{wi}$ ($i = 1, 2, 3, 4$) ditetapkan, mengambil pusat roda sebagai asal mula sistem. $[x \ y \ \theta]^T$ didefinisikan sebagai kecepatan umum platform seluler. Menurut karakteristik geometris robot, untuk roda Mecanum apa pun O_{wi} , $\alpha_i + \beta_i = 0$. Berdasarkan Persamaan (3), dapat diperoleh rumus sebagai berikut.

$$\varphi_i = \frac{-1}{r_v \sin \gamma_i} [\cos \gamma_i \sin \gamma_i / \sin(\beta_i + \gamma_i)] \zeta = \frac{-1}{r_v} [\cot \gamma_i \ 1 \ W - H \cot \gamma_i] [y] \quad (1)$$

di mana sudut rol $\gamma_1 = \gamma_3 = -\gamma = -45^\circ$, $\gamma_2 = \gamma_4 = \gamma = 45^\circ$ maka persamaan invers kinematik robot roda empat Mecanum sebagai berikut:

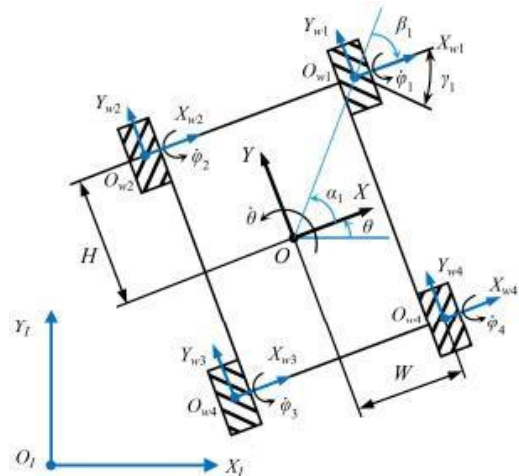
$$\begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \varphi_3 \\ \varphi_4 \end{bmatrix} = -\frac{1}{r_v} \begin{bmatrix} -\cot \gamma & 1 & W + H \cot \gamma \\ \cot \gamma & 1 & -W - H \cot \gamma \\ -\cot \gamma & 1 & -W - H \cot \gamma \\ \cot \gamma & 1 & W + H \cot \gamma \end{bmatrix} [y] = -\frac{1}{r_v} \begin{bmatrix} -1 & 1 & W + H \\ 1 & 1 & -(W + H) \\ -1 & 1 & -(W + H) \\ 1 & 1 & W + H \end{bmatrix} [y] \quad (2)$$

Matriks Jacobian kinematik terbalik J dinyatakan sebagai:

$$J = -\frac{1}{r_v} \begin{bmatrix} -1 & 1 & W + H \\ 1 & 1 & -(W + H) \\ -1 & 1 & -(W + H) \\ 1 & 1 & W + H \end{bmatrix} \quad (3)$$

Dalam sistem koordinat global $X_1O_1Z_1$, persamaan kinematik terbalik robot beroda empat mecamum adalah:

$$\begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \varphi_3 \\ \varphi_4 \end{bmatrix} = JR(\theta) [y_1] = -\frac{1}{r_v} \begin{bmatrix} -1 & 1 & W + H \\ 1 & 1 & -(W + H) \\ -1 & 1 & -(W + H) \\ 1 & 1 & W + H \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & x_1 \\ -\sin \theta & 0 & y_1 \\ 0 & 0 & 1 \end{bmatrix} [\theta] \quad (4)$$



Gambar 2. Pergerakan Invers Kinematika dengan Roda Mecanum [8]

C. Implementasi PID

PID adalah cara klasik teori kendali otomatis dan mode kendali percobaan umum. Sistem kendali PID dibuat dalam bentuk algoritma program (software) yang nantinya diinput kedalam mikrokontroler [9]. Posisi dapat dikontrol oleh pengontrol PID. Pengontrol PID menyuplai sinyal kontrol yang mempunyai komponen yang sebanding dengan kesalahan pelacakan suatu sistem, komponen yang sebanding dengan akumulasi kesalahan ini sepanjang waktu, dan komponen yang sebanding dengan laju perubahan kesalahan ini terhadap waktu. Modul ini akan mencakup komponen-komponen yang berbeda dan beberapa kombinasi berbeda yang dapat digunakan untuk tujuan kontrol. Dalam aplikasi teknik, pengontrol muncul dalam berbagai bentuk: sebagai pengontrol yang berdiri sendiri, sebagai bagian dari sistem kontrol terdistribusi yang hierarkis, yang dibangun ke dalam komponen tertanam. Pengontrol proporsional-integral-turunan (pengontrol PID) banyak digunakan dalam sistem kendali industri. Ini adalah mekanisme loop kontrol umpan balik umum dan digunakan sebagai pengontrol umpan balik.

Loop PID menentukan berapa banyak energi yang harus diupayakan ke motor setiap saat selama bergerak. Hal ini berdasarkan di mana sepeda motor tersebut berada dan diharapkan berada di mana. Ada empat bagian dalam persamaan dan ini menentukan beban dimana tiga komponen

utama disebut sebagai P, I, D dan komponen gesekan kecil disebut sebagai K.

P menentukan reaksi terhadap kesalahan arus, I menentukan reaksi terhadap kesalahan arus. jumlah kesalahan yang muncul baru-baru ini, dan D menentukan reaksi berdasarkan laju perubahan kesalahan. Jumlah dari ketiga bagian tersebut berkontribusi pada mekanisme kendali seperti kendali kecepatan motor dimana nilai P bergantung pada kesalahan saat ini, I juga bergantung pada akumulasi kesalahan sebelumnya dan D memprediksi kesalahan di masa depan berdasarkan laju perubahan saat ini [10]. Sinyal PWM dibuat oleh mikrokontroler untuk selanjutnya dipakai mengendalikan kecepatan motor [11].

$$U_I(t) = K_I \int_0^t e(t) dt \quad (5)$$

$$U_D(t) = K_D \frac{d}{dt} e(t) \quad (6)$$

$$U(t) = K_p \cdot e(t) + K_I \int_0^t e(t) dt + K_D \frac{d}{dt} e(t) \quad (7)$$

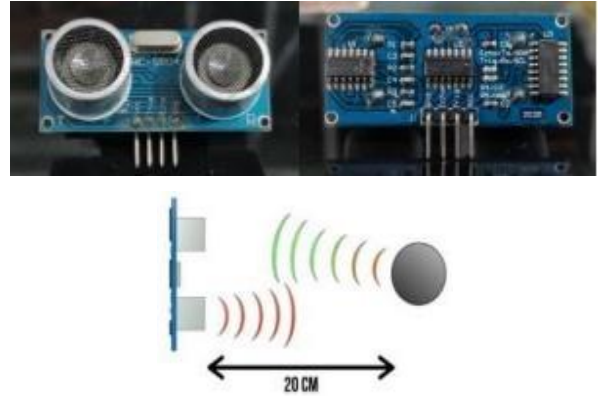
Tabel 1. Tabel Aturan tuning Ziegler-Nichols berdasarkan penguatan kritis K_{cr} dan periode kritis P_{cr} (metode kedua).

Tippe Kontroller	Kp	Ki	Kd
P	$0.5K_{cr}$	∞	0
Pi	$0.45K_{cr}$	$1/1.2P_{cr}$	0
PID	$0.6K_{cr}$	$0.5P_{cr}$	$0.125P_{cr}$

$$G(s) = K \left(1 + \frac{1}{p} + T_{is} \right) \quad (8)$$

D. Sensor Jarak

Sensor jarak adalah modul pengukur jarak ultrasonik. Sensor jarak yang digunakan cukup kecil, berukuran 2,1 cm kali 4,5 cm, dan dapat mengukur jarak 3 cm hingga 300 cm. Sensor ini sangat mudah digunakan pada mikrokontroler karena memanfaatkan empat pin pada sensor, dua pin catu daya untuk sensor ultrasonik dan satu pin trigger dan satu pin echo sebagai input output dan hasil dari sensor ke Arduino. Suara ultrasonik dengan frekuensi 40 KHz akan ditransmisikan sebesar 200uS. Suara ini akan melewati udara dengan kecepatan 344,424m/detik, kemudian dipantulkan kembali ke sensor[12]. Pulsa akan dihasilkan oleh sensor sambil menunggu pantulan. Detak jantung ini akan berhenti ketika suara pantulan dikenali oleh sensor. Hasilnya, jarak antara sensor dan objek dapat direpresentasikan dengan lebar pulsa. Untuk menghitung jarak sebenarnya dengan menggunakan sensor jarak dapat menyesuaikan setiap sensor yang akan digunakan dikarenakan kecenderungannya setiap sensor jarak mempunyai hasil jarak yang berbeda-beda.



Gambar 3. Sensor Ultrasonik [14].

Pada gambar 3, prinsip kerja sensor ultrasonik HC-SR04 diantaranya menggunakan 10 trigger dengan minimal 10us sinyal high, modul juga secara otomatis mengirimkan 8 kali 40 KHz dan mendeteksi memiliki sinyal balik atau tidak jika terdapat sinyal balik maka durasi waktu dari output high adalah waktu dari pengiriman dan penerimaan ultrasonik [13].

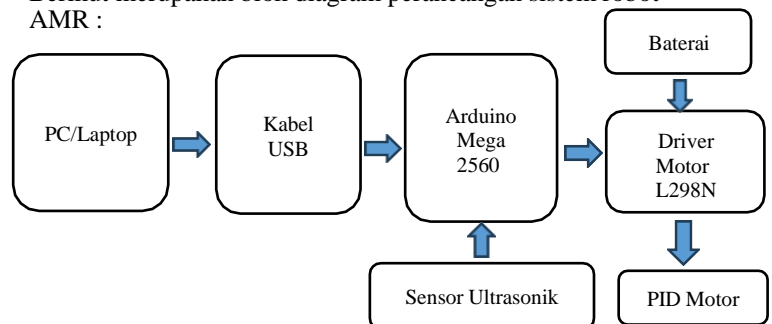
E. Sistem Obstacle Avoidance Robot AMR

Sistem *obstacle avoidance robot* dapat disusun menjadi tiga bagian, yaitu sensor khusus, motor dan kontrol motor menggunakan aplikasi Arduino untuk memasukkan perintah ke dalam robot sesuai kebutuhan. Robot ini dimodifikasi untuk menghindari rintangan, misalnya tembok atau benda lain yang dapat dibedakan oleh sensor sebagai penghalang. AMR (Autonomous Mobile Robot) melibatkan empat sensor ultrasonik sebagai indikator pencegah, yaitu sensor depan, sensor kiri, sensor kanan, sensor belakang. Aturan penting dari AMR (Autonomous Mobile Robot) adalah robot dapat membedakan benda atau rintangan yang dekat dengan robot. Robot akan mengenali suatu benda bilamana robot tersebut dihalangi dan akan berbelok ke arah jalan dimana robot tidak dapat membedakan benda atau hambatan [15].

III. METODE PENELITIAN

A. Blok Diagram Perancangan Sistem Robot AMR

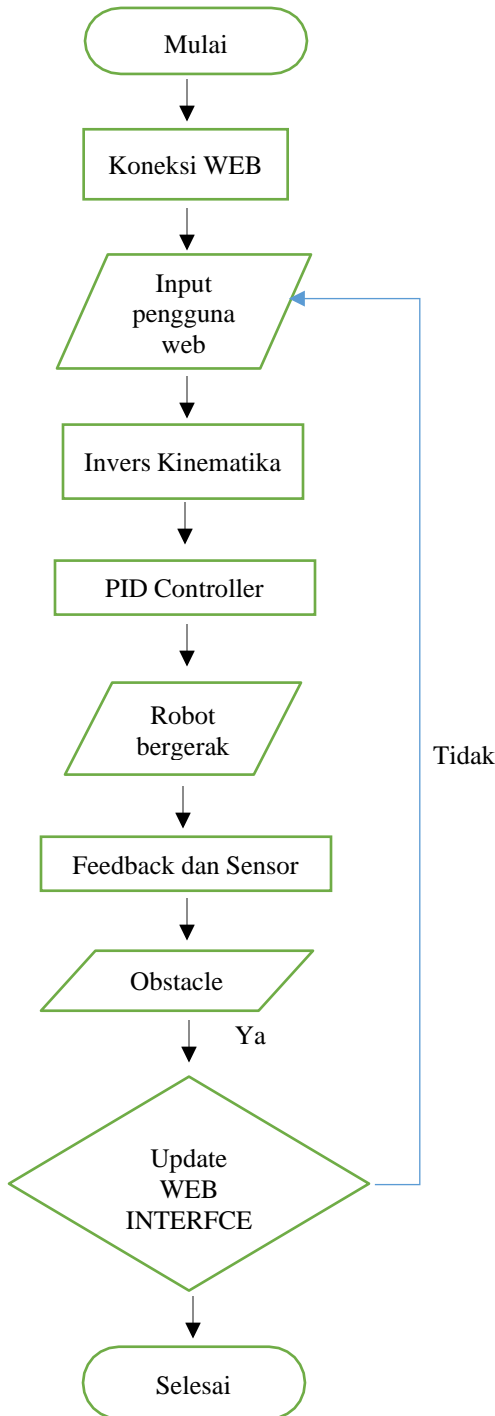
Berikut merupakan blok diagram perancangan sistem robot AMR :



Gambar 4. Blok diagram robot AMR

B. Flowchart Perancangan Sistem Robot AMR

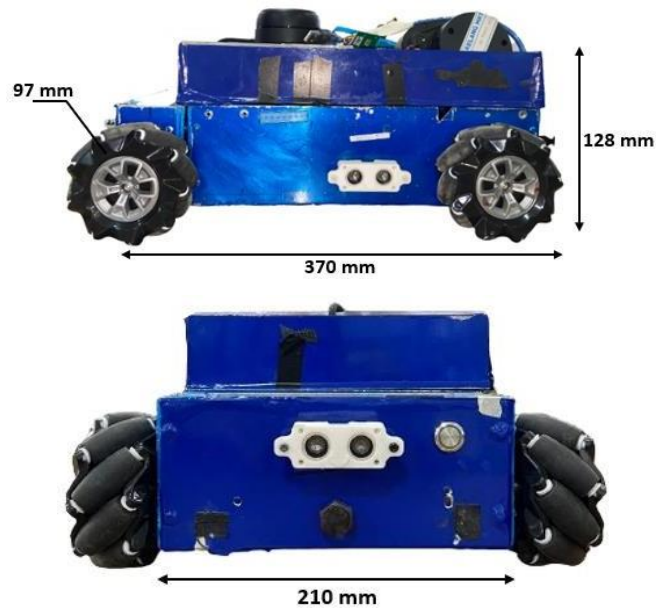
Berikut merupakan flowchart perancangan sistem robot AMR :



Gambar 5. Flowchart robot AMR.

C. Desain Mekanikal Robot AMR

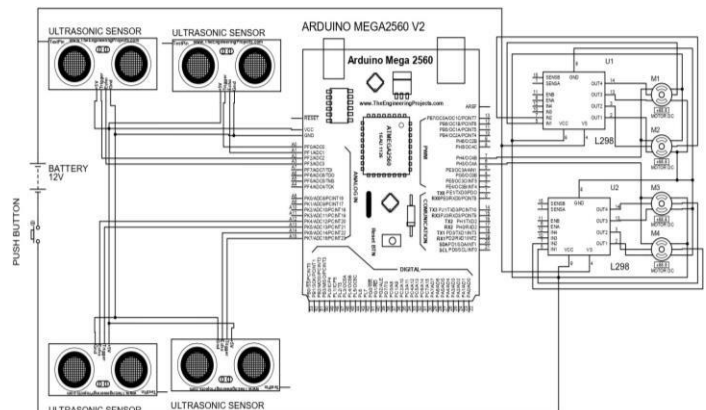
Di bawah ini adalah desain dan ukuran pada robot AMR menggunakan plat bahan aluminium sebagai material utama.



Gambar 6. Tampilan Ukuran Robot AMR

D. Desain Skematik Elektrikal Robot AMR

Di bawah ini adalah desain skematik elektrikal pada robot AMR :



Gambar 7. Desain skematik elektrikal AMR

Keterangan :

1. Baterai 11.1 V
2. Motor DC
3. Arduino mega 2560
4. Motor driver L298N
5. Push button
6. Sensor ultrasonik

IV. HASIL PENGUJIAN

A. Pengujian waktu pengiriman data antar pc

Pada pengujian ini waktu data pengiriman yang diambil menggunakan 2 PC yang saling berkomunikasi untuk mengirim data maupun menerima data yang dapat kita lihat melalui stopwatch yang berada di dalam tampilan web.

Tabel 2 . Tampilan tabel pengujian waktu pengiriman data

Percobaan	PC 1 Kirim (Sec)	Delay Time	PC 2 Kirim (Sec)	Delay Time	Keterangan
1	2.67	0	2.99	0	Terkirim
2	1.63	0	1.13	0	Terkirim
3	1.77	0	0.68	0	Terkirim
4	2.47	0	2.88	0	Terkirim
5	2.64	0	1.05	0	Terkirim
6	2.86	0	1.64	0	Terkirim
7	2.60	0	2.92	0	Terkirim
8	2.17	0	1.3	0	Terkirim
9	1.03	0	1.89	0	Terkirim
10	1.78	0	2.22	0	Terkirim
Waktu Tertinggi	2.86	0	2.99	0	
Waktu Terendah	1.03	0	0.68	0	
Rata - Rata	2.162	0	1.87	0	

Dari uraian tabel 2 diatas dapat kita lihat hasil pengujian dari pengiriman data antara PC 1 dan PC 2 tidak ada mengalami delay time sehingga data dapat terkirim dengan baik. Selain itu waktu terlama mengirim data ialah 2,99 detik dan waktu tercepat ialah 0,68 detik.

B. Pengujian Gerak Robot dengan Invers Kinematika

Metode pada Pengujian ini robot akan bergerak dari titik A ke titik B dengan jarak 5 meter dengan berjalan lurus ke depan, kemudian akan diukur nilai error dari garis yang ditentukan.



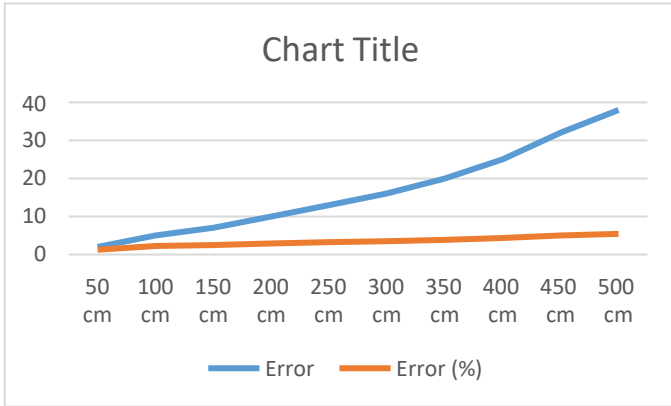
Gambar 8. Titik awal pengujian robot jalan lurus

Pada gambar 8, kondisi awal ketika robot sebelum mulai bergerak untuk mengikuti garis yang telah ditentukan kemudian robot akan berjalan dengan melakukan metode PID motor. namun ketika melewati garis robot mulai keluar dari jalur yang ditentukan. Dibawah ini adalah hasil pengujian nilai error pada robot.

Tabel 3. Tabel Pergerakan untuk Robot

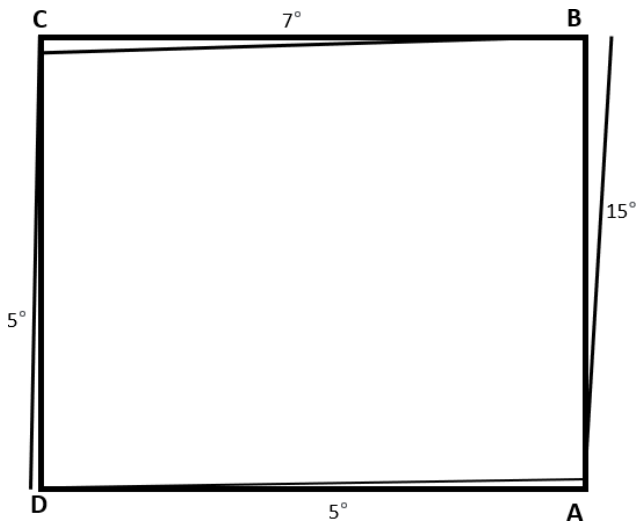
No	A-B	Error (cm)	Error (%)
1	50 cm	2	1,25
2	100 cm	5	2,27
3	150 cm	7	2,5
4	200 cm	10	2,94
5	250 cm	13	3,25
6	300 cm	16	3,47
7	350 cm	20	3,84
8	400 cm	25	4,31
9	450 cm	32	5
10	500 cm	38	5,42
Rata -Rata %			3,425

Pada tabel 3 di atas dapat dilihat bahwasannya eror dari pergerakan sumbu dimana pergerakan robot bergerak dari titik A menuju ke titik B dapat menghasilkan error semakin besar.



Gambar 9. Grafik pengujian rata rata

Hasil grafik di atas terdapat error ketika robot AMR berjalan maka error semakin membesar dengan rata rata total 3,425%.



Gambar 10. Lintasan kotak pada robot AMR

Pada gambar 10 terdapat lintasan kotak pada robot AMR. Lintasan ini terdiri dari 4 point. Titik point A ke titik point B, titik point B ke titik point C, titik point C ke titik point D. Setelah itu kembali ke titik awal.

C. Pengujian pada kendali PID

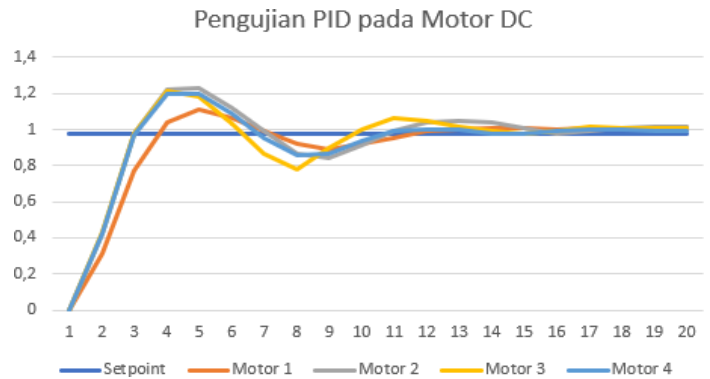
Kontrol PID diharapkan untuk mendapatkan reaksi kerangka kerja. Penyesuaian nilai parameter PID Kp, Ki, dan Kd diperlukan agar diperoleh respon dari sistem. Dalam pengujian ini, strategi tuning Ziegler Nichols 2 diputuskan untuk menentukan batasan akomodasi PID dalam analisis.

Berikut adalah parameter PID yang akan menentukan kecepatan pada motor DC.

Tabel 5. Tabel konstanta parameter PID

Tipe Kontrol	Kp	Ki	Kd
P	0.2	0	0
PID	3.5	1.0	0.065

Langkah selanjutnya adalah memilih jenis kontrol dengan melihat hasil grafik respon sistem setelah parameter PID ditentukan. Setelah menentukan batasan PID, tahap selanjutnya adalah memilih tipe kontrol dengan memperhatikan efek dari diagram reaksi kerangka.



Gambar 11. Nilai grafik PID pada motor

Dari hasil pada gambar dapat dilihat tipe kontrol pada tiap motor memiliki kecepatan yang sangat berbeda.

Tabel 6. Tabel Hasil Simulasi PID pada motor

Tuning Method	Kp	Ki	Kd	Maximum Overshoot	Rise Time	Settling Time
ZN Method	0.2	0.02	0.065	1,23%	0.4	2

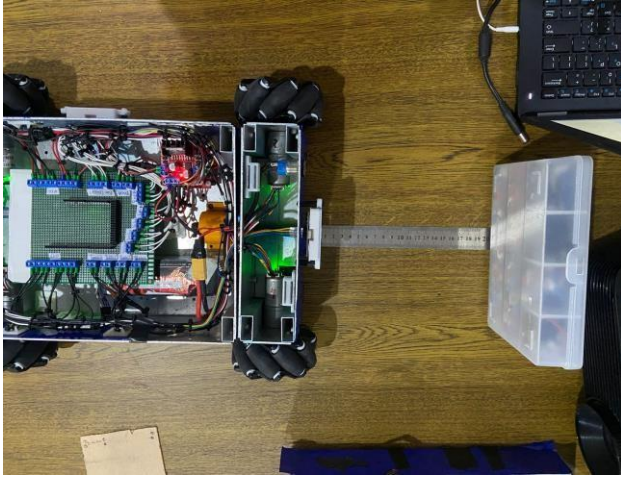
Pada tabel 6 terdapat dari hasil dari simulasi dengan tiap pada motor dc mendapatkan hasil overshoot sebesar 1.23%, waktu rise time 0.4 dan dengan settling time 2.

D. Analisa dan Pengujian Sensor Ultrasonik

Sensor ultrasonik mempunyai aturan fungsi dalam melihat gelombang suara. Tanda yang dibuat akan dikeluarkan dari pemancar ultrasonik. Ketika tanda tersebut mengenai suatu benda atau rintangan, maka tanda tersebut akan dipantulkan kembali dan diterima oleh rangkaian mikrokontroler yang kemudian akan diproses untuk memastikan jarak ke benda di depannya. Pengujian dilakukan untuk mengetahui ketepatan sensor ultrasonik yang akan digunakan dan untuk menyesuaikan sensor apabila sensor ultrasonik tidak sesuai dengan yang diinginkan.

Dengan menggunakan sensor ultrasonik, pengujian pada robot mobil otonom beroda empat membuahkan hasil dari sudut pandang depan, kanan, kiri, dan belakang. Dari segi hasil pengujian alat, tahap awal diawali dengan pembacaan sensor. Kondisi ketika sensor tidak membaca suatu benda dalam jarak 20 cm, maka informasi tersebut akan dimasukkan ke dalam tampilan Arduino dan selanjutnya informasi tersebut akan

diproses. Selama tidak ada benda yang dapat dibedakan, peralatan akan bergerak hingga perangkat tersebut mengidentifikasi objek penghalang berikut dalam sistem pengujian.



Gambar 12. Pengujian sensor ultrasonik

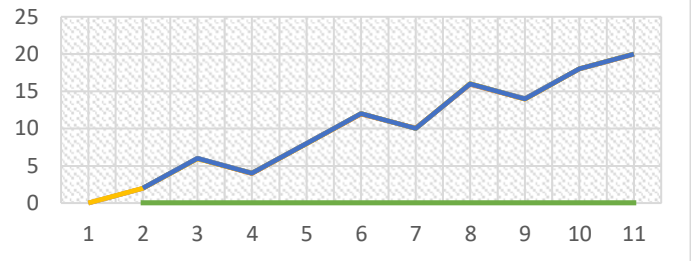
Pada gambar 12, pengujian dilakukan dengan cara membandingkan nilai yang terbaca dari sensor ultrasonik dengan jarak sebenarnya, yaitu dengan meletakkan penghalang di depan sensor ultrasonik maka nilai yang terbaca akan ditampilkan pada aplikasi Arduino.

Tabel 7. Pembacaan Sensor Ultrasonik

Jarak Real (cm)	Percobaan 1	Percobaan 2	Percobaan 3	Rata Rata Hasil Percobaan	Error Jarak (%)
2	1,99	2	1,98	1,99	0,01
4	3,96	4,03	4	3,99	0,03
6	5,99	6,03	5,95	5,99	0,04
8	8,01	8,02	7,95	7,99	0,03
10	10,05	9,99	10,02	10,02	0,03
12	11,96	12,01	12	11,99	0,02
14	13,95	14,05	13,94	13,98	0,06
16	15,95	15,99	16,01	15,98	0,03
18	18,01	18,02	18	18,01	0,01
20	19,95	19,99	20	19,98	0,02

Tabel 7 menampilkan hasil pengujian dan pengukuran yang dilakukan. Secara khusus, pengukuran jarak sebenarnya tidak selalu sesuai dengan pengukuran sensor ultrasonik.

Grafik Pengujian Sensor Ultrasonik



Gambar 13. Grafik pengujian sensor ultrasonik

Dari data yang diambil, pada gambar 13 sensor ultrasonik dapat mengukur jarak dengan presisi yang sangat baik, meskipun masih terdapat kesalahan pada saat memperkirakan objek yang jaraknya sangat jauh dengan sensor ultrasonik.

V. KESIMPULAN

Adapun kesimpulan dari hasil penelitian yang dibuat, yaitu :

1. Web app dapat berkomunikasi dan mengirimkan data antar 2 PC dengan data yang terkirim tidak mengalami delay time sehingga data terkirim dengan baik.
2. Robot sudah dapat melakukan dengan 2 kali percobaan dimana dengan lintasan jalur lurus dan jalur kotak. Hasil pengujian jalan lurus rata – rata error telah mencapai 3,425% dan pengujian lintasan jalur kotak rata – rata error telah mencapai 0,051%.
3. Berdasarkan hasil pengujian parameter kontrol PID yang digunakan masing – masing parameter motor adalah $K_p = 0.2$, $K_i = 0.02$, dan $K_d = 0,065$
4. *Autonomous Mobile Robot* dapat mendeteksi dan menghindari halangan atau objek dengan jarak 20 cm yang berada di depan, kanan, kiri, dan belakang. Alat sudah berfungsi secara optimal, hanya saja sensor tidak dapat mendeteksi objek yang berada di atas dan bawah robot. Untuk hasil pengujian dan analisis data pengukuran menunjukkan bahwa semakin jauh jarak objek pengukuran maka akurasi pengukurannya cenderung menurun.

Referensi

- [1] R. S. Wijaya, A. Kaputra, N. A. Prasetyo, dan H. Soebhakti, "Omni-directional Movement on the MRT PURVI Ship Robot," vol. 7, no. 2, hal. 1–8, 2023.
- [2] A. S. Taufik, "Sistem Navigasi Waypoint pada Autonomous Mobile Robot," *J. Mhs. TEUB*, vol. 1, no. 1, hal. 1–6, 2013, [Daring]. Tersedia pada: <http://elektro.studentjournal.ub.ac.id/index.php/teub/article/view/7>
- [3] M. Wahdan, "Motion planning for autonomous mobile

- robots,” *AEJ - Alexandria Eng. J.*, vol. 44, no. 1, hal. 51–57, 2005.
- [4] M. B. Alatise dan G. P. Hancke, “A Review on Challenges of Autonomous Mobile Robot and Sensor Fusion Methods,” *IEEE Access*, vol. 8, hal. 39830–39846, 2020, doi: 10.1109/ACCESS.2020.2975643.
- [5] M. Yunus dan M. As’ad, “Protokol Tcp/Ip Sebagai Sarana Dalam Proses Transfer Data,” *J. Teknol. Inf.*, vol. 3, no. 1, hal. 75–92, 2012, doi: 10.36382/jti-tki.v3i1.121.
- [6] I. Putu, A. E. Pratama, P. Adhika Dharmesta, dan T. Informasi, “IMPLEMENTASI WIRESHARK DALAM MELAKUKAN PEMANTAUAN PROTOCOL JARINGAN (Studi Kasus : Intranet Jurusan Teknologi Informasi Universitas Udayana),” *J. Mantik Penusa*, vol. 3, no. 1, hal. 94, 2019.
- [7] S. Setiawan, Firdaus, B. Rahmadya, dan Derisma, “Penerapan Invers Kinematika Untuk Pergerakan Kaki Robot Biped,” no. November, hal. 1–9, 2015.
- [8] Y. Li *et al.*, “Kinematic modeling of a combined system of multiple mecanum-wheeled robots with velocity compensation,” *Sensors (Switzerland)*, vol. 20, no. 1, 2020, doi: 10.3390/s20010075.
- [9] P. Seminar, N. Nciety, dan N. Conference, “Sistem Kontrol Proportional Integral Derivative (Pid) Untuk Mengatur Kecepatan Motor Dc Menggunakan Mikrokontroler,” *Pros. Semin. Nas. NCIET*, vol. 1, no. 1, hal. 528–534, 2020, doi: 10.32497/nciet.v1i1.172.
- [10] W. W. Naing, K. Z. Aung, dan A. Thike, “Position Control of 3-DOF Articulated Robot Arm using PID Controller,” vol. 7, no. 09, hal. 254–259, 2018.
- [11] R. T. Yunardi, A. Muchadin, K. L. Priyanti, dan D. Arifianto, “Inverse Kinematics and Pid Controller Implementation of Hexapod Robot for Wall Follower Navigation,” *Ina. Indones. J. Electr. Eletronics Eng.*, vol. 2, no. 2, hal. 57–62, 2019, doi: 10.26740/inajeee.v2n2.p23-28.
- [12] M. Z. Lubis, W. Anurogo, dan P. Novanto, “Desain dan Uji Coba Sederhana Pada Obstacle Avoiding Robot Menggunakan Desain dan Uji Coba Sederhana Pada Obstacle Avoiding Robot Menggunakan Mikrokontroler Arduino,” vol. 3, no. 2, hal. 47–53, 2018.
- [13] A. N. A. Setiawan dan C. Bella, “Rancangan Bangun Prototipe Program Cerdas Perparkiran Mobil Memanfaatkan TCRT5000 Berbasis Arduino Dengan Kendali Terpusat,” *J. PORTAL DATA*, vol. 1, no. 3, hal. 1–20, 2021.
- [14] A. M. Sudiro, M. A. Putra, dan E. Yadie, “Rancang Bangun Prototipe Follower Cargo Cart with Obstacle Avoider Berbasis Arduino Uno,” *PoliGrid*, vol. 3, no. 2, hal. 63, 2022, doi: 10.46964/poligrid.v3i2.1717.
- [15] M. R. R. Jusman, A. Mashinton, F. Jilan Aqilah, M. Nur, dan P. Bosowa, “Rancang Bangun Robot AVOIDER Berbasis Arduino Uno Menggunakan Satu Sensor Ultrasonik,” *Mechatronics J. Prof. Entrep.*, vol. 2, no. 2, hal. 43–47, 2020.