

Trajectory Tracking pada Robot *Omni* dengan Metode *Odometry*

Fahmizal¹, Dhiya Uddin Rijalussalam², Ma'un Budiyanto³, Afrizal Mayub⁴

Abstract—This paper presents trajectory tracking on omni robots using odometry method. The odometry system aims to estimate the position relative to the initial position of the omni robot to estimate changes in position over time. The rotary encoder sensor is used to count the movement of the omni robot at the x and y coordinates in the odometry calculation process in this study. Furthermore, using the reverse kinematics on the omni robot, the rotational speed value of each DC motor in the omni robot wheel is obtained. In addition, to obtain the results of a good robot movement on the odometry system, PID control is applied to control the rotational speed of each DC motor on the omni robot wheel. With reverse kinematics and odometry systems, omni robot trajectory designs can be easily built. To test the odometry method performance in trajectory tracking process, there are three types of trajectory testing patterns, namely rectangles, equilateral triangles, and equilateral triangles. From the results of this test, the error value obtained is below 5%.

Intisari—Makalah ini memaparkan *trajectory tracking* pada robot *omni* dengan metode *odometry*. Sistem *odometry* bertujuan untuk memperkirakan posisi relatif terhadap posisi awal robot *omni* untuk memperkirakan perubahan posisi dari waktu ke waktu. Sensor *rotary encoder* digunakan untuk mencacah pergerakan robot *omni* pada koordinat x dan y pada proses perhitungan *odometry*. Selanjutnya, dengan menggunakan kinematika balik pada robot *omni*, nilai kecepatan putar masing-masing motor DC pada roda robot *omni* diperoleh. Selain itu, untuk memperoleh hasil pergerakan robot yang baik pada sistem *odometry*, kendali PID diterapkan untuk mengendalikan kecepatan putar masing-masing motor DC pada roda robot *omni*. Dengan kinematika balik dan sistem *odometry*, desain *trajectory* robot *omni* dapat dengan mudah dibangun. Untuk menguji kinerja metode *odometry* dalam melakukan proses *trajectory tracking*, terdapat tiga jenis pola pengujian *trajectory*, yaitu persegi panjang, segitiga sama sisi, dan segitiga sama kaki. Dari hasil pengujian ini, diperoleh nilai kesalahan di bawah 5%.

Kata Kunci— *Trajectory Tracking, Robot Omni, Odometry.*

I. PENDAHULUAN

Perkembangan ilmu pengetahuan dan teknologi yang sedemikian pesat telah membawa dampak yang cukup besar

^{1,3} Dosen, Departemen Teknik Elektro dan Informatika, Sekolah Vokasi, Universitas Gadjah Mada, Sekip Unit III, Gd. Herman Yohanes, Yogyakarta 55281, (Telp: 0274-561111; Fax.: 0274-542908, e-mail: ¹fahmizal@ugm.ac.id; ³m.budiyanto@ugm.ac.id)

² Mahasiswa, Departemen Teknik Elektro dan Informatika, Sekolah Vokasi, Universitas Gadjah Mada, Sekip Unit III, Gd. Herman Yohanes, Yogyakarta 55281 (Telp: 0274-561111; Fax.: 0274-542908, e-mail: rijalsmp2123@gmail.com)

⁴ Dosen, Pascasarjana S2 Pendidikan IPA FKIP, Universitas Bengkulu, Jl. WR Supratman, Bengkulu 38371 (Telp: 0736-21186; Fax.: 0736-21186, e-mail: afrizalmayub@unib.ac.id)

terhadap kehidupan manusia untuk mempelajari dan mengembangkan ilmu pengetahuan. Salah satunya adalah teknologi robotika. Semakin banyak inovasi robot yang diciptakan baik dalam segi hemat sumber energi maupun efisien waktu. Robot dapat dikatakan sempurna ketika dapat menjangkau segala arah tanpa harus berbelok atau memutar arah. Namun, yang menjadi salah satu masalah dalam *mobile robot* adalah mobilitas robot.

Pada umumnya, navigasi robot yang sering dijumpai bertipe *ackerman steering* dan *differential steering* yang memiliki mobilitas rendah. Keduanya dapat bergerak maju dan berbelok, tetapi robot tidak mampu bergerak ke segala arah atau biasa disebut dalam kategori robot *non-holonomic*. Oleh sebab itu, dikembangkan sebuah robot yang tidak hanya dapat bergerak maju dan mundur, tetapi dapat bergerak ke segala arah dalam bidang kartesian x - y tanpa perlu memutar badannya, yaitu salah satunya adalah *omnidirectional* robot atau robot roda *omni*. Dengan demikian, robot roda *omni* ini akan mempunyai mobilitas yang tinggi [1], [2]. Terdapat dua jenis desain robot roda *omni*, yaitu dengan konfigurasi tiga buah roda dan empat buah roda. Pengaturan posisi roda *omni* sangat memengaruhi pergerakan robot secara signifikan. Semakin jauh jarak roda depan dengan roda belakang, semakin cepat posisi robot dapat diputar.

Dalam menentukan sistem pergerakan robot, dapat digunakan dua jenis media, yaitu luar dan dalam. Contoh media dari luar adalah garis dan dinding [3]. Namun, penggunaan media dari luar kurang begitu efisien. Robot seharusnya dapat menentukan arah pergerakan dari dalam robot itu sendiri. Sensor yang umum digunakan untuk mendapatkan data posisi dari suatu robot adalah *rotary encoder*. Data dari sensor ini dimasukkan dalam perhitungan *odometry* sehingga menghasilkan posisi relatif dari robot tersebut [4]. Metode *odometry* telah banyak digunakan untuk penyelesaian suatu robot dalam proses melakukan sebuah *trajectory tracking*. *Trajectory* merupakan metode untuk menentukan pola gerakan dari sebuah robot dari titik awal menuju titik akhir. Algoritme dari *trajectory tracking* dikategorikan menjadi dua macam, yaitu *off-line path trajectory* dan *on-line trajectory* [5]. Pada makalah ini, digunakan *odometry* pada kasus *off-line tracking* pada robot roda *omni* dengan konfigurasi empat roda.

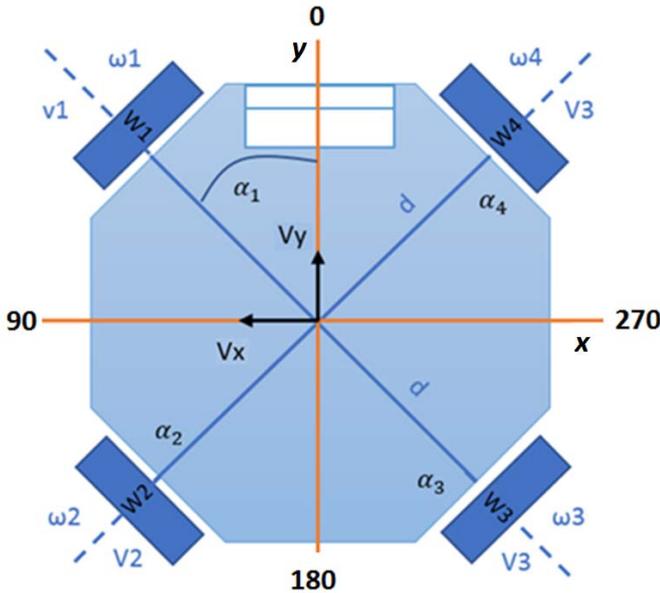
Dalam makalah ini, robot *omni* bergerak secara otomatis sesuai dengan *trajectory* yang diberikan. Untuk menjaga kecepatan putar roda stabil, kendali PID dengan struktur independen diterapkan. Kendali PID digunakan untuk menjaga agar nilai kecepatan motor yang dihasilkan selama proses jalannya robot mengikuti *trajectory* dengan metode *odometry* memiliki tingkat kesalahan yang kecil. Makalah ini disusun sebagai berikut. Pada Bagian II, metodologi dalam makalah ini diusulkan. Bagian III

menjelaskan perancangan perangkat keras robot *omni*. Bagian IV menyajikan prosedur implementasi *odometry* pada robot *omni* dan eksperimen yang menunjukkan keefektifan dan manfaat metode yang diusulkan. Bagian V merupakan kesimpulan.

II. METODOLOGI

A. Kinematika pada Omnidirectional Robot

Kinematika robot sangat penting untuk mendefinisikan arah dan kecepatan robot. Berdasarkan sifatnya, robot jenis ini termasuk dalam kategori robot *holonomic*. Robot dapat bertranslasi ke sebelah kiri (atau kanan) tanpa terlebih dahulu mengubah orientasi robot [6], dengan menggunakan kinematika balik yang mengubah masukan (*input*) posisi dalam bentuk koordinat (x,y) dengan sudut θ menjadi kecepatan masing-masing roda, yaitu $\omega_1, \omega_2, \omega_3, \omega_4$, dan arah putaran roda. Gbr. 1 dan Tabel I merupakan rancangan robot *omni* dengan bentuk *frame* “X” dengan empat buah roda sehingga sudut yang terbentuk antar sumbu roda dengan sumbu y, $\alpha_1 = 45^\circ, \alpha_2 = 135^\circ, \alpha_3 = 225^\circ, \alpha_4 = 315^\circ$.



Gbr. 1 Skematik susunan roda *omni* dengan konfigurasi empat roda.

Dari nilai sudut yang terbentuk antara roda, maka diperoleh (1).

$$V_i = -\sin(\alpha_i)x + \cos(\alpha_i)y + R\theta \tag{1}$$

dengan $R\theta$ merupakan proses rotasi dari perubahan orientasi robot. Selanjutnya, nilai kecepatan sudut dari masing-masing roda dapat dihitung dengan menggunakan (2).

$$V_i = R\omega_i \tag{2}$$

Dari (1) dan (2), diperoleh (3).

$$\omega_i = \frac{1}{R}(-\sin(\alpha_i)x + \cos(\alpha_i)y + R\theta) \tag{3}$$

TABEL I
SPESIFIKASI ROBOT *OMNI*

No.	Simbol	Keterangan
1	x dan y	Merupakan koordinat posisi robot dari acuan peta global.
2	$\alpha_1, \alpha_2, \alpha_3, \alpha_4$	Merupakan sudut yang terbentuk antara roda dengan sumbu y pada robot.
3	$W1, W2, W3, W4$	Merupakan simbol untuk roda pada setiap sisi robot.
4	d [meter]	Merupakan jarak (meter) antara roda dan pusat roda.
5	v_1, v_2, v_3, v_4 [m/s]	Merupakan kecepatan linear roda robot.
6	$\omega_1, \omega_2, \omega_3, \omega_4$ [rad/s]	Merupakan kecepatan sudut roda.
11	V_x	Merupakan kecepatan terhadap sumbu x robot.
12	V_y	Merupakan kecepatan terhadap sumbu y robot.

Dari (3), bila diuraikan ke masing-masing roda, maka diperoleh persamaan kinematika balik seperti yang disajikan pada (4).

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} = \frac{1}{R} \begin{bmatrix} -\sin(\alpha_1) & \cos(\alpha_1) & R \\ -\sin(\alpha_2) & \cos(\alpha_2) & R \\ -\sin(\alpha_3) & \cos(\alpha_3) & R \\ -\sin(\alpha_4) & \cos(\alpha_4) & R \end{bmatrix} \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \tag{4}$$

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} = \frac{1}{R} \begin{bmatrix} -\sin(45^\circ) & \cos(45^\circ) & R \\ -\sin(135^\circ) & \cos(135^\circ) & R \\ -\sin(225^\circ) & \cos(225^\circ) & R \\ -\sin(315^\circ) & \cos(315^\circ) & R \end{bmatrix} \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$$

Dikarenakan perhitungan *cosinus* dalam satuan radian, (4) diubah menjadi pada (5).

$$\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} = \frac{1}{R} \begin{bmatrix} -\sin(\pi/4) & \cos(\pi/4) & R \\ -\sin(3\pi/4) & \cos(3\pi/4) & R \\ -\sin(5\pi/4) & \cos(5\pi/4) & R \\ -\sin(7\pi/4) & \cos(7\pi/4) & R \end{bmatrix} \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \tag{5}$$

B. Sistem Odometry pada Omnidirectional Robot

Odometry adalah penggunaan data dari pergerakan aktuator untuk memperkirakan perubahan koordinat posisi dari waktu ke waktu. *Odometry* digunakan untuk memperkirakan koordinat posisi relatif terhadap posisi awal [7], [8]. Pada sistem *odometry* robot beroda, sensor yang digunakan adalah *rotary encoder* sebagai pendeteksi jumlah putaran roda. Terdapat tiga buah parameter utama dalam menghitung koordinat posisi robot yaitu diameter roda *free wheel* ($DW_{(i)}$), jumlah resolusi *encoder* ($resolusi_{enc}$), dan jumlah pulsa *rotary encoder* yang dihasilkan (*pulse*). Untuk menghitung keliling roda *free wheel* ($KW_{(i)}$), dapat digunakan (6). Bila digunakan

sumbu gerak Kartesian, maka perubahan koordinat posisi x (x_{tempuh}) dan koordinat posisi y (y_{tempuh}) dapat dihitung menggunakan (7) dan (8).

$$KW_{(i)} = DW_{(i)} \times \pi \tag{6}$$

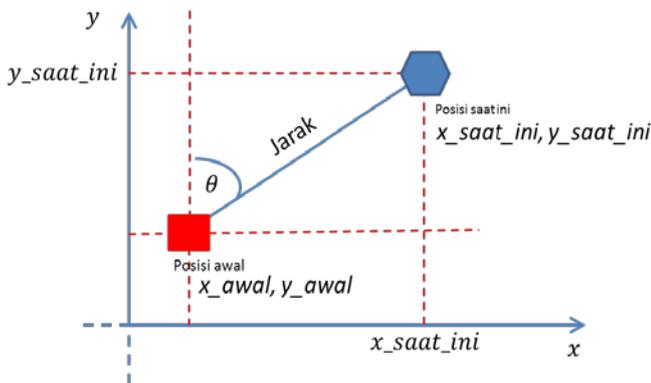
$$x_{tempuh} = \frac{pulse_{(x)}}{resolusi_{enc}} KW_{(x)} \tag{7}$$

$$y_{tempuh} = \frac{pulse_{(y)}}{resolusi_{enc}} KW_{(y)} \tag{8}$$

Untuk mengetahui orientasi arah robot *omni* (θ), koordinat posisi saat ini ($x_{saat_ini}, y_{saat_ini}$) relatif terhadap pergerakan posisi awal (x_{awal}, y_{awal}), seperti yang disajikan pada Gbr. 2, maka orientasi robot (θ) dapat diperoleh dengan (9). Sedangkan panjang jarak tempuh dari posisi saat ini terhadap posisi awal dapat dicari menggunakan (10).

$$\theta = \tan^{-1} \frac{y_{tempuh}}{x_{tempuh}} \tag{9}$$

$$Jarak = \sqrt{(x_{saat_ini} - x_{awal})^2 + (y_{saat_ini} - y_{awal})^2} \tag{10}$$



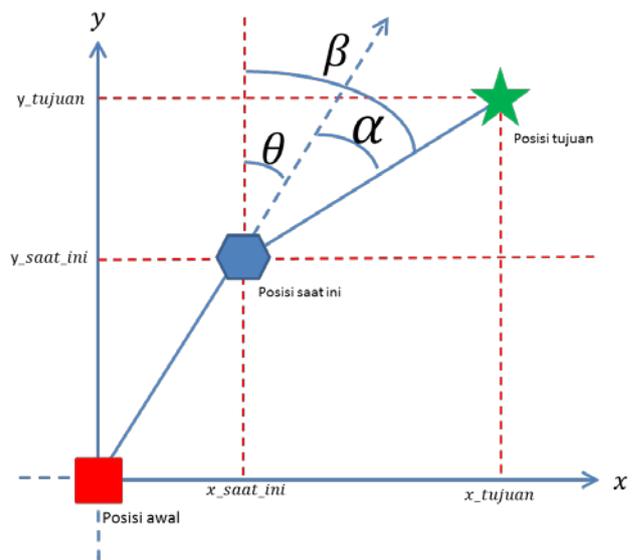
Gbr. 2 Perhitungan jarak tempuh robot pada sistem *odometry*.

Secara lengkap, perubahan koordinat dan orientasi pada sistem *odometry* tiap waktu ke waktu disajikan pada Gbr. 3. Nilai perubahan tersebut terdiri atas perubahan posisi orientasi dan koordinat robot terhadap koordinat global robot, dengan θ merupakan arah orientasi robot dan target *bearing* (β) yaitu sudut antara posisi robot saat ini terhadap titik tujuan. Untuk menentukan nilai β dan *heading error* robot (α), digunakan (11) dan (12). Selanjutnya, jarak tujuan dapat dihitung menggunakan (13).

$$\beta = 90^\circ - \left[\tan^{-1} \frac{y_{tujuan} - y_{saat_ini}}{x_{tujuan} - x_{saat_ini}} \right] \tag{11}$$

$$\alpha = \beta - \theta \tag{12}$$

$$Jarak = \sqrt{(x_{tujuan} - x_{saat_ini})^2 + (y_{tujuan} - y_{saat_ini})^2} \tag{13}$$

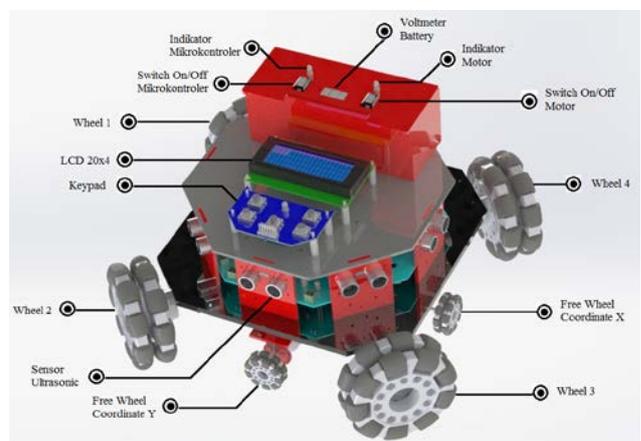


Gbr. 3 Ilustrasi posisi robot pada sumbu Kartesian pada sistem *odometry*.

III. PERANCANGAN ROBOT *OMNI*

A. Perancangan Mekanik

Mekanik robot *omni* dirancang menggunakan enam buah roda yang terdiri atas empat buah roda penggerak dan dua buah roda *free wheel* yang berfungsi sebagai penghitung proses *odometry*. Gbr. 4 menampilkan tampak keseluruhan dari rancangan robot *omni*. Lebih jauh, Gbr. 5 menyajikan tampilan *base* bawah dari robot *omni*. Terlihat jelas bahwa konfigurasi roda *free wheel* untuk sumbu gerak x dan y saling tegak lurus. Penjelasan spesifikasi ukuran desain robot *omni* ini disajikan pada Tabel II.

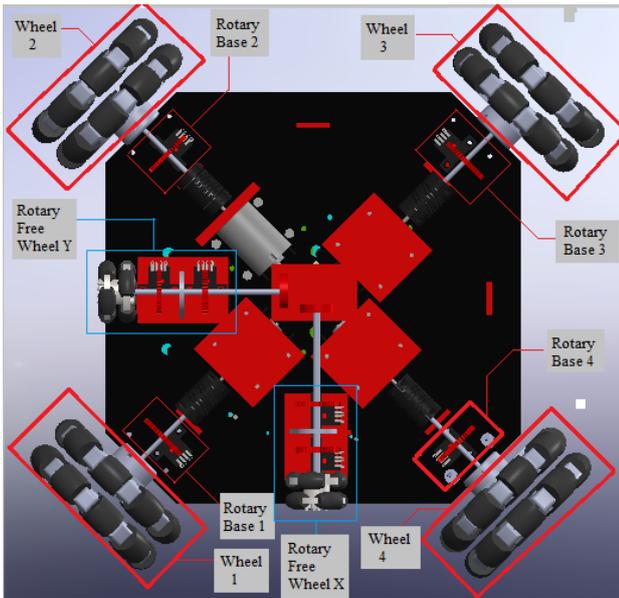


Gbr. 4 Desain keseluruhan robot *omni*.

B. Perancangan Rotary Encoder

Pada makalah ini digunakan enam buah *rotary encoder* dengan konfigurasi empat *rotary* terpasang satu *shaft* dengan *base wheel*, disebut *rotary base wheel*, dan dua *rotary* terpasang satu *shaft* pada *free wheel*, atau disebut *rotary free wheel*. Implementasi *rotary encoder* pada robot ini menggunakan sebuah piringan *encoder* dan sebuah *optocoupler* berbentuk “U” seperti yang disajikan pada Gbr. 6.

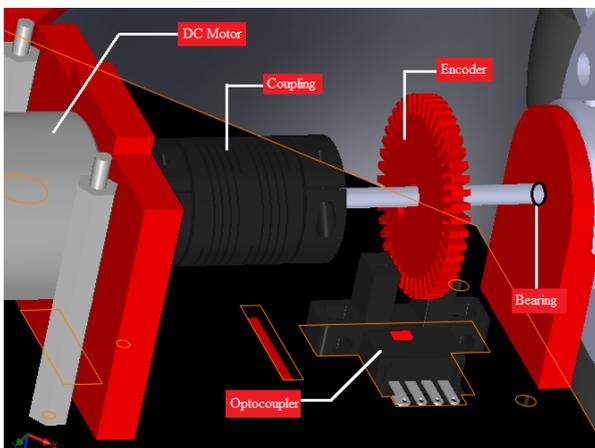
Rotary base wheel digunakan sebagai umpan balik dalam hal mengatur kecepatan putar motor DC pada masing-masing *wheel* robot *omni*.



Gbr. 5 Desain *frame base* bawah robot *omni*.

TABEL II
SPESIFIKASI FRAME ROBOT *OMNI*

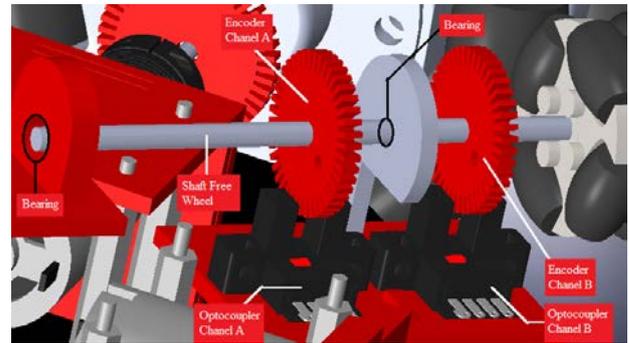
Panjang Diameter	250,00 mm
Panjang Diagonal	300,63 mm
Luas <i>Frame</i>	58.901,08 mm ²
Jarak Roda <i>Base</i> dengan pusat <i>frame</i>	182,63 mm
Jarak Roda <i>Odometry</i> dengan pusat <i>frame</i>	135,30 mm
Panjang <i>Shaft</i> dari Motor ke Roda	88,42 mm



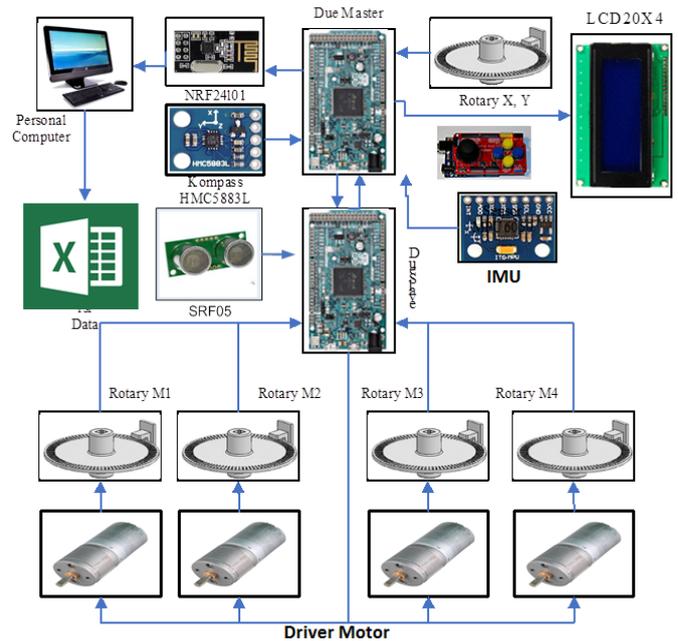
Gbr. 6 Perancangan *rotary base wheel*.

Rotary free wheel merupakan sensor *rotary* yang berfungsi untuk menghitung nilai koordinat *x* dan *y* terhadap koordinat global robot. *Rotary free wheel* terdiri atas dua buah *encoder* dan dua buah *optocoupler*. *Rotary free wheel* ini hanya terhubung satu *shaft* dengan *free wheel*. Oleh karena itu, untuk mengetahui arah putarannya, harus menggunakan lebih dari satu buah *encoder*, yaitu dengan cara mengetahui *channel*

yang *leading* terhadap *channel* satunya. Gbr. 7 menunjukkan perancangan *rotary free wheel*. Hasil pembacaan data dari *rotary free wheel* selanjutnya digunakan pada proses *odometry* robot *omni* dalam melakukan proses *trajectory tracking* sebuah lintasan.



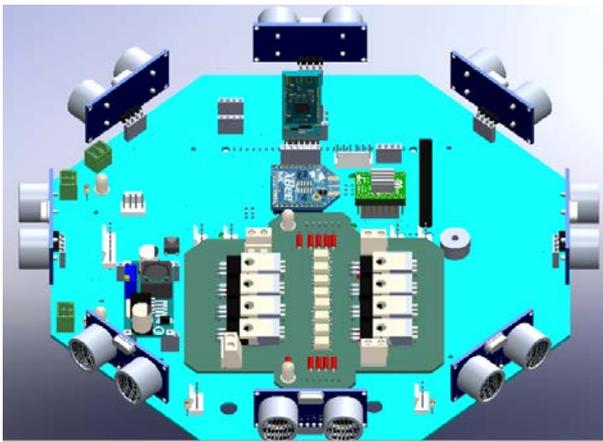
Gbr. 7 Perancangan *rotary free wheel*.



Gbr. 8 Diagram skematik perangkat elektronik robot *omni*.

C. Perancangan Elektronik

Sistem elektronik robot *omni* menggunakan dua buah Arduino Due yang dikonfigurasi sebagai *master* dan *slave*. Arduino Due *slave* digunakan untuk menangani seluruh pembacaan *rotary encoder* yang berjumlah enam buah, sedangkan Arduino Due *master* sebagai pengolah algoritme utama dari proses *odometry* robot *omni*. Arduino Due *master* juga dikonfigurasi untuk dapat berkomunikasi secara *wireless* dengan *personal computer* (PC) menggunakan NRF2401. Sensor kompas HMC5883L dan IMU MPU 6050 juga dikonfigurasi menggunakan komunikasi I2C pada Arduino Due *master*. Adapun diagram blok konfigurasi sistem elektronik dari robot *omni* secara keseluruhan disajikan pada Gbr. 8. Lebih jauh, hasil rancangan *printed circuit board* (PCB) dari robot *omni* ini disajikan pada Gbr. 9.



Gbr. 9 Desain PCB perangkat elektronik robot omni.

IV. IMPLEMENTASI DAN PENGUJIAN SISTEM ODOMETRY PADA ROBOT OMNI

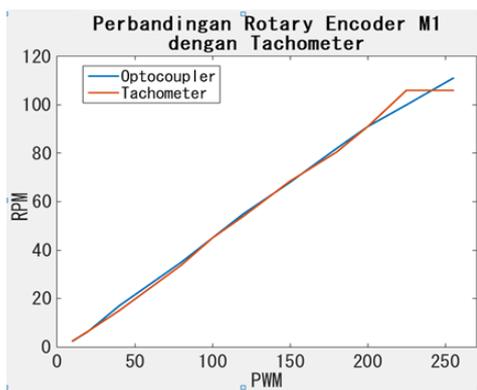
A. Pengujian Rotary Encoder sebagai Kendali Kecepatan Motor DC dari Roda Robot Omni

Pada makalah ini diterapkan kendali PID untuk mengatur kecepatan putar motor DC dari masing-masing roda robot omni. Namun, sebelum kendali PID ini diterapkan, dilakukan terlebih dahulu proses kalibrasi pembacaan rotary encoder

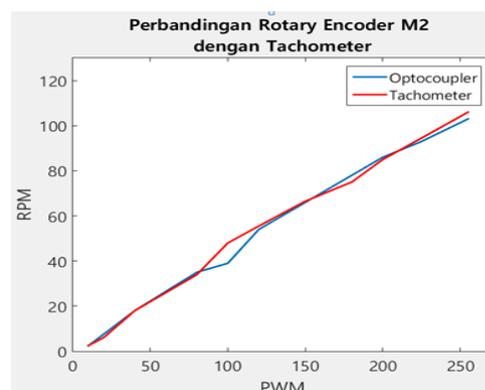
pada masing-masing roda. Pengujian ini dilakukan untuk mengetahui sensor rotary encoder yang digunakan dapat bekerja dengan baik atau tidak dalam mengukur kecepatan putar motor DC. Pengujian dilakukan dengan membandingkan nilai yang terukur dari rotary encoder setelah diproses oleh Arduino Due dengan nilai pembacaan tachometer seperti yang disajikan pada Tabel III.

TABEL III
PENGUJIAN OPTOCOUPLER SEBAGAI SENSOR KECEPATAN MOTOR DC

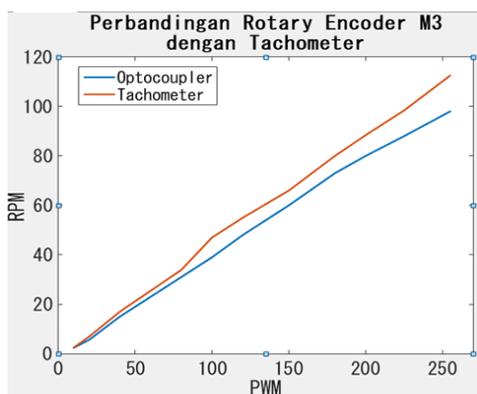
PWM	Rotary Encoder (Optocoupler)				Tachometer				Tegangan (V)
	M1	M2	M3	M4	M1	M2	M3	M4	
	Revolution per Minutes (RPM)								
10	2,4	2,4	2,4	2,4	2,5	2,5	2,5	2,5	0,4706
20	6,4	7,6	5,6	6,5	6,5	6,1	6,9	6,8	0,9412
40	17	18	15	16	15	18	17	16	1,8824
80	35	35	31	33	34	34	33,9	33,9	3,7647
100	45	39	39	39	45	48	47	40	4,7059
120	55	54	48	51	54	55,5	55	54,5	5,6471
150	68	66	60	64	69	66,5	66	65,5	7,0588
180	82	78	73	76	81	75	80	74,6	8,4706
200	91	86	80	84	91	85	88,4	83,4	9,4118
225	100	93	88	94	106	94,5	98,4	93,5	10,588
255	111	103	98	106	106	106	113	106	12



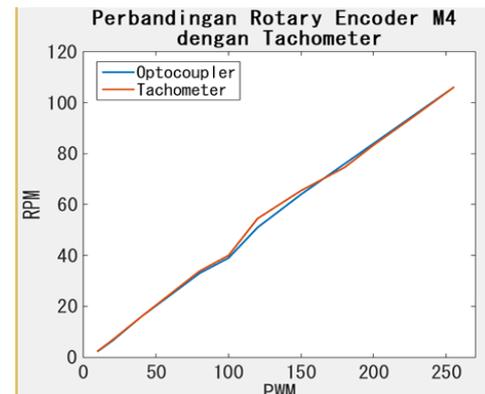
(a)



(b)



(c)



(d)

Gbr. 10 Grafik perbandingan pengukuran kecepatan rotary encoder dengan tachometer pada masing-masing motor.

TABEL IV
PENGUJIAN BERBEBAN UNTUK PWM MOTOR DC KE-1 (M1)

No.	PWM	M1 (RPM)	RPM/PWM	PWM/RPM	Perubahan Speed
1	40	11	0	0	0
2	60	24	0,275	0	0
3	80	33	0,4	2,5	13
4	100	46	0,4125	2,424242424	9
5	120	55	0,46	2,173913043	13
6	140	64	0,458333333	2,181818182	9
7	160	71	0,457142857	2,1875	9
8	180	77	0,44375	2,253521127	7
9	200	85	0,427777778	2,337662338	6
10	220	91	0,425	2,352941176	8
11	255	97	0,413636364	2,417582418	6
Rata-rata			0,427853249	2,345804669	8,6
Persamaan Regresi PWM			PWM = -0,911 + 2,39 * RPM		
Persamaan Regresi RPM			RPM = 1,95 + 0,41 * PWM		

TABEL V
PENGUJIAN BERBEBAN UNTUK PWM MOTOR DC KE-2 (M2)

No.	PWM	M2 (RPM)	RPM/PWM	PWM/RPM	Perubahan Speed
1	10	0	0	0	0
2	20	0	0	0	0
3	40	12	0,3	0	0
4	60	24	0,4	2,5	12
5	80	33	0,4125	2,424242424	9
6	100	46	0,46	2,173913043	13
7	120	55	0,458333333	2,181818182	9
8	140	67	0,478571429	2,089552239	12
9	160	73	0,45625	2,191780822	6
10	180	79	0,438888889	2,278481013	6
11	200	84	0,42	2,380952381	5
12	220	90	0,409090909	2,444444444	6
13	255	99	0,388235294	2,575757576	9
Rata-rata			0,295276356	3,464999869	8,6489362
Persamaan Regresi PWM			PWM = -1,6 + 2,38 * RPM		
Persamaan Regresi RPM			RPM = 2,37 + 0,40 * PWM		

Dari Tabel III, dapat dibuat grafik hubungan hasil pembacaan *rotary encoder* dengan *tachometer*, seperti yang disajikan pada Gbr. 10.

Setelah itu, pengujian kecepatan masing-masing motor DC dengan beban dilakukan. Tujuannya untuk mendapatkan data karakteristik masing-masing motor DC pada saat dijalankan. Selain itu, langkah pengujian ini digunakan untuk menentukan nilai regresi dari masing-masing motor DC yang digunakan. Pengujian dilakukan dengan cara memberikan nilai *pulse width modulation* (PWM) dan mengamati respons dari pembacaan sensor kecepatan pada *rotary encoder*. Tabel IV menyajikan hasil pengujian berbeban untuk PWM motor DC ke-1 (M1), Tabel V menyajikan hasil pengujian berbeban untuk PWM motor DC ke-2 (M2), Tabel VI menyajikan hasil pengujian berbeban untuk PWM motor DC ke-3 (M3), dan Tabel VII menyajikan hasil pengujian berbeban untuk PWM motor DC ke-4 (M4).

TABEL VI
PENGUJIAN BERBEBAN UNTUK PWM MOTOR DC KE-3 (M3)

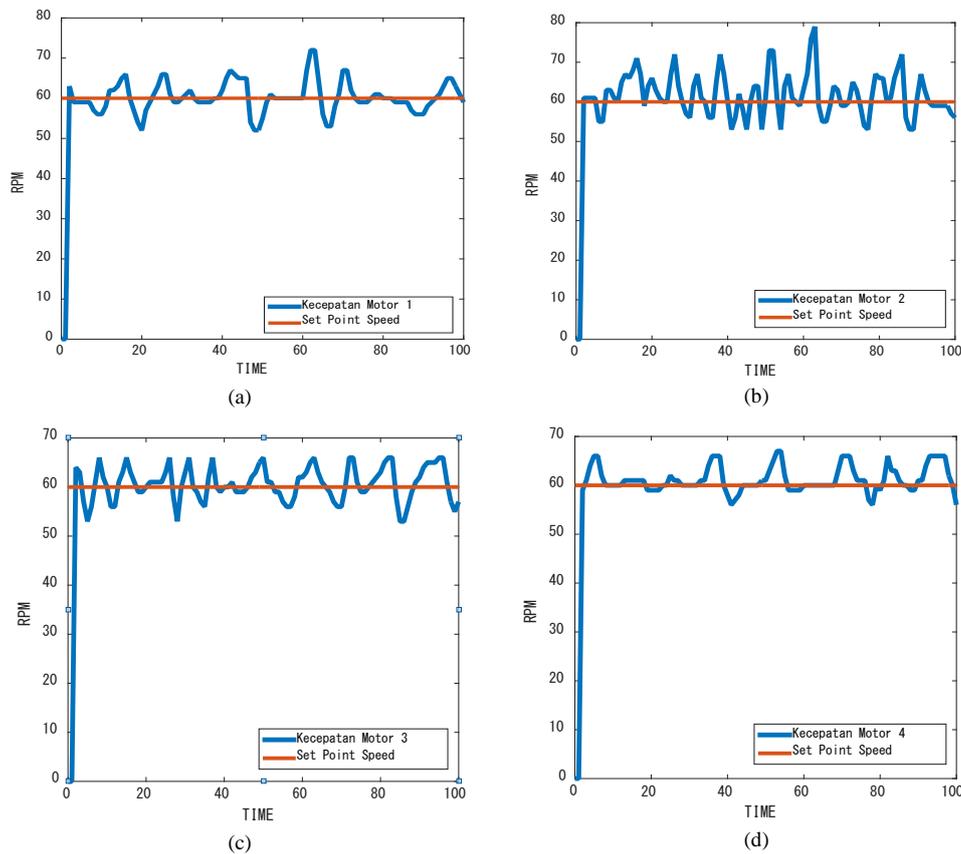
No.	PWM	M3 (RPM)	RPM/PWM	PWM / RPM	Perubahan Speed
1	10	0	0	0	0
2	20	0	0	0	0
3	40	11	0,275	0	0
4	60	26	0,433333333	2,307692308	15
5	80	35	0,4375	2,285714286	9
6	100	43	0,43	2,325581395	8
7	120	52	0,433333333	2,307692308	9
8	140	58	0,414285714	2,413793103	6
9	160	63	0,39375	2,53968254	5
10	180	70	0,388888889	2,571428571	7
11	200	78	0,39	2,564102564	8
12	220	83	0,377272727	2,65060241	5
13	255	89	0,349019608	2,865168539	6
Rata-rata			0,30876053	3,257595649	0,4047384
Persamaan Regresi PWM			PWM = -10,47 + 2,75* RPM		
Persamaan Regresi RPM			RPM = 5,12 + 0,35 * PWM		

TABEL VII
PENGUJIAN BERBEBAN UNTUK PWM MOTOR DC KE-4 (M4)

No.	PWM	M4 (RPM)	RPM/PWM	PWM/ RPM	Perubahan Speed
1	20	0	0	0	0
2	40	12	0,3	0	0
3	60	22	0,366666667	2,727272727	10
4	80	30	0,375	2,666666667	8
5	100	39	0,39	2,564102564	9
6	120	47	0,391666667	2,553191489	8
7	140	53	0,378571429	2,641509434	6
8	160	60	0,375	2,666666667	7
9	180	67	0,372222222	2,686567164	7
10	200	74	0,37	2,702702703	7
11	220	80	0,363636364	2,75	6
12	255	100	0,392156863	2,55	20
Rata-rata			0,224701697	4,48574564	0,377492
Rumus Regresi PWM			PWM = 4,13 + 2,58 * RPM		
Persamaan Regresi RPM			RPM = -1,30 + 0,38 * PWM		

Dari data-data pada Tabel IV hingga Tabel VII, diperoleh nilai regresi dari setiap motor DC. Nilai regresi ini kemudian digunakan sebagai acuan untuk mengonversi nilai *revolution per menit* (RPM) menjadi PWM, dengan nilai *set point* kecepatan yang diinginkan dalam satuan RPM. Selanjutnya, untuk mendapatkan respons kecepatan masing-masing motor DC sesuai dengan RPM yang diinginkan, diterapkan kendali PID. Untuk mendapatkan parameter konstanta PID (K_p , K_i , dan K_d), dilakukan *tuning* eksperimen. Langkah ini dilakukan dengan menggunakan kendali proporsional saja pada awalnya, seperti yang disajikan pada Gbr. 11.

Respons dengan kendali proporsional belum memberikan hasil yang baik. Selanjutnya, penambahan kendali integral dan diferensial dilakukan. Gbr. 12 menyajikan hasil respons masing-masing motor DC setelah penambahan kendali integral dan diferensial.



Gbr. 11 Kendali proporsional pada masing-masing motor DC; (a) $K_p=2, K_i=0, K_d=0$; (b) $K_p=2, K_i=0, K_d=0$; (c) $K_p=2, K_i=0, K_d=0$; (d) $K_p=2, K_i=0, K_d=0$.

Setelah parameter kendali PID diperoleh, langkah selanjutnya adalah menguji *rotary encoder* pada roda *free wheel* yang digunakan pada sistem *odometry* robot *omni*.

B. Pengujian Rotary Encoder pada Free Wheel sebagai Sistem Odometry Robot Omni

Pengujian ini dilakukan untuk memperoleh data aktual dari *rotary encoder* pada *free wheel* berupa jarak tempuh pergerakan pada sumbu *x* dan *y*. Proses pengujian dilakukan dengan cara membandingkan nilai pembacaan dari *rotary encoder* dengan panjang penggaris (*mistar*). Pengujian dilakukan dengan variasi panjang penggaris hingga 1.000 mm. Tabel VIII menyajikan hasil pengujian pergerakan robot *omni* pada sumbu *x* dan Tabel IX menyajikan hasil pengujian pergerakan robot *omni* pada sumbu *y*.

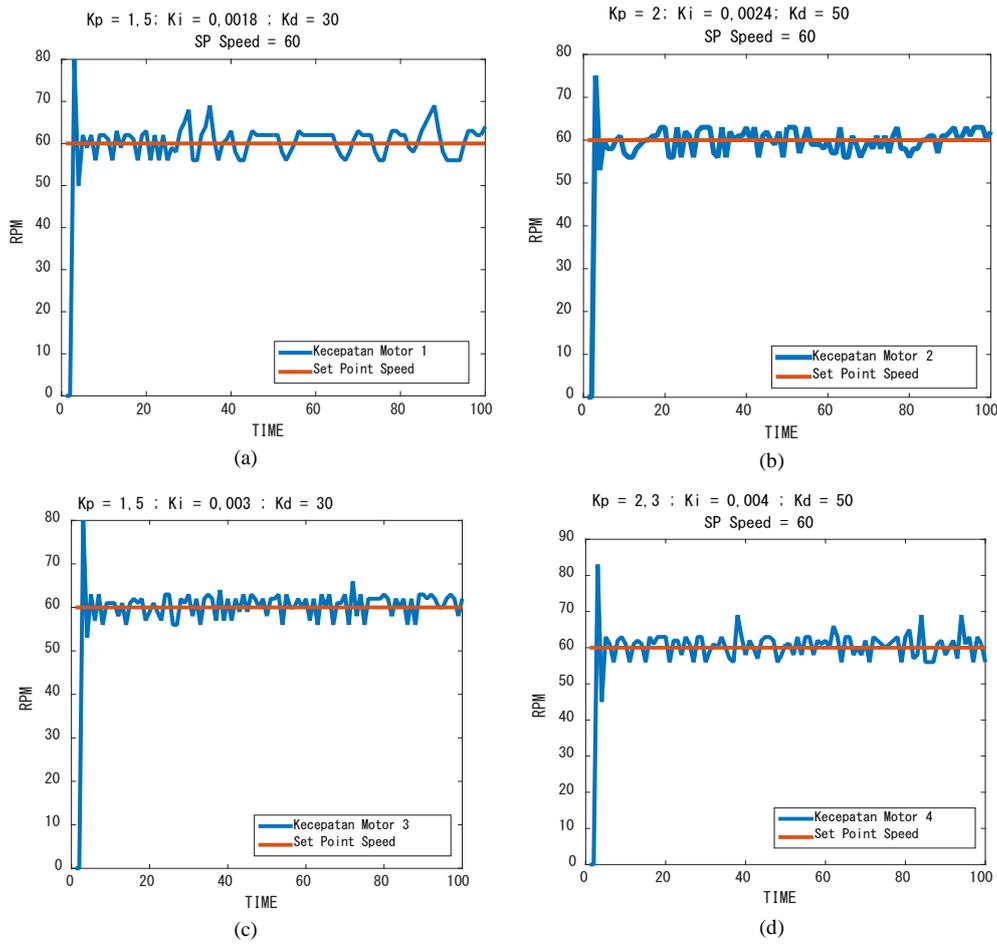
Dari data pada Tabel VIII dan Tabel IX diperoleh hasil bahwa pembacaan pergerakan robot, baik pada sumbu *x* dan sumbu *y*, memiliki tingkat kesalahan rata-rata di bawah 5% pada rentang jarak 1.000 mm. Dengan mengacu pada data tersebut, maka dapat dikatakan pembacaan jarak tempuh robot menggunakan sensor *rotary encoder* dapat bekerja dengan baik. Setelah pengujian *rotary encoder* pada kecepatan masing-masing motor DC robot *omni* dan *rotary encoder* pada *free wheel* sistem *odometry* robot *omni*, maka langkah selanjutnya ialah menguji robot *omni* untuk dapat mengikuti sebuah *trajectory* yang diberikan. Dilakukan pengujian sebanyak tiga variasi *trajectory*, yaitu membentuk pola persegi panjang, segitiga sama sisi, dan segitiga sama kaki.

TABEL VIII
HASIL PENGUJIAN PERGERAKAN ROBOT OMNI PADA SUMBU X

Data pengujian Rotary Encoder sumbu x						
No	Koordinat x					
	Mistar	Rotary Encoder	Pulse	Error	Perubahan	Tingkat Kesalahan
	(mm)			(mm)	Error (mm)	(%)
1	100	100,57	30	0,57	0	1%
2	200	199,97	60	0,03	0,54	0%
3	300	307,77	91	7,77	7,74	3%
4	400	415	123	15	7,23	4%
5	500	501,49	150	1,49	13,51	0%
6	600	605,41	181	5,41	3,92	1%
7	700	708,33	211	8,33	2,92	1%
8	800	810,75	242	10,75	2,42	1%
9	900	905,61	270	5,61	5,14	1%
10	1.000	1.009,53	301	9,53	3,92	1%
Rata-rata				6,449	4,734	1%

C. Pengujian Trajectory Tracking Robot Omni

Bergeraknya robot *omni* dalam mengikuti lintasan (*trajectory tracking*) dapat dilakukan dengan cara menghitung kinematika balik (*inverse kinematic*) robot *omni* seperti yang disajikan pada (5). Dengan memberi nilai vektor kecepatan (v_x, v_y , dan/atau posisi (x,y)), diperoleh nilai kecepatan masing-masing roda beserta dengan arah putar dari roda tersebut. Pada



Gbr. 12 Kendali PID pada masing-masing motor DC; (a) Respons kendali PID motor 1; (b) Respons kendali PID motor 2; (c) Respons kendali PID motor 3; (d) Respons kendali PID motor 4.

makalah ini, *trajectory* didesain dengan memberikan nilai koordinat atau posisi robot *omni* yang dituju pada sumbu Kartesian, yaitu sumbu *x* dan *y*, seperti yang disajikan pada diagram blok pada Gbr. 13. Percobaan pertama berupa persegi panjang dengan koordinat seperti yang dijelaskan pada (14) yang berupa *array* posisi robot *omni* dalam hal mengikuti lintasan persegi panjang. Pada *array* pertama, diawali pada posisi $(x,y)=(0,0)$ dan diikuti *array* selanjutnya. Persamaan (15) digunakan untuk membuat pola *trajectory* berbentuk segitiga sama kaki dan (16) digunakan untuk membuat pola *trajectory* berbentuk segitiga sama sisi.

$$x = \{0, 800, 800, 0, 0\} \tag{14}$$

$$y = \{0, 0, -800, -800, 0\}$$

$$x = \{0, 800, 800, 0, 0\} \tag{15}$$

$$y = \{0, 0, -800, -800, 0\}$$

$$x = \{0, 800, 800, 0, 0\} \tag{16}$$

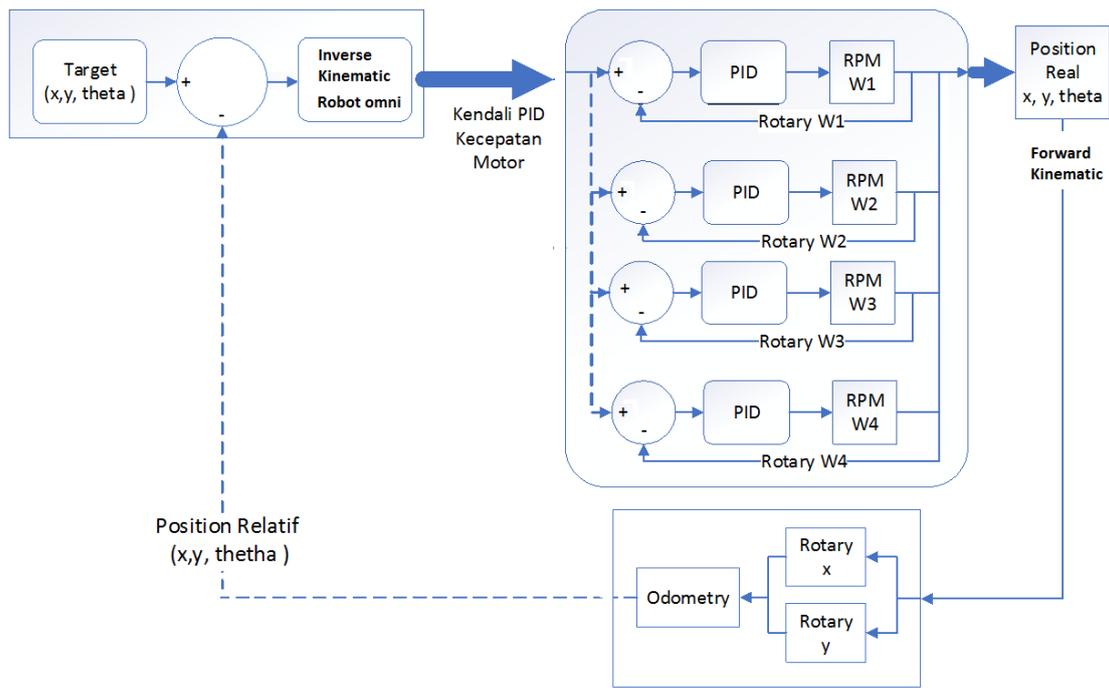
$$y = \{0, 0, -800, -800, 0\}$$

Gbr. 14 dan Tabel X menyajikan hasil pengujian robot *omni* saat *trajectory* berupa persegi panjang. Pada hasil

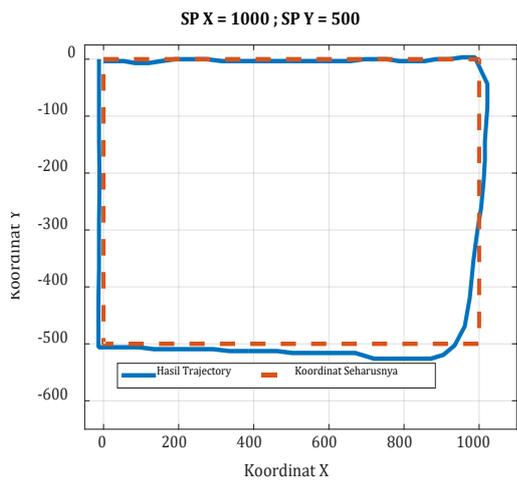
pengujian *trajectory* persegi panjang, hasil yang didapatkan yaitu pergerakan robot menunjukkan *error* maksimum sebesar 50 mm pada koordinat *x*, dan *error* sebesar 70 mm pada koordinat *y*.

TABEL IX
HASIL PENGUJIAN PERGERAKAN ROBOT *OMNI* PADA SUMBU *Y*

Data pengujian <i>Rotary Encoder</i> sumbu <i>y</i>						
No	Koordinat <i>y</i>					
	Mistar	<i>Rotary Encoder</i>	Pulse	Error	Perubahan	Tingkat Kesalahan
	(mm)			(mm)	Error (mm)	(%)
1	100	107,28	32	7,28	3,92	7%
2	200	211,2	63	11,2	5,5	6%
3	300	305,7	91	5,7	3,41	2%
4	400	402,29	120	2,29	0,57	1%
5	500	502,86	150	2,86	2,04	1%
6	600	600,82	181	0,82	6,53	0%
7	700	707,35	210	7,35	3,93	1%
8	800	811,28	242	11,28	13,98	1%
9	900	925,26	276	25,26	7,27	3%
10	1.000	1.032,53	307	32,53	0	3%
Rata-rata				10,66	4,715	2%



Gbr. 13 Diagram blok trajectory tracking robot omni.

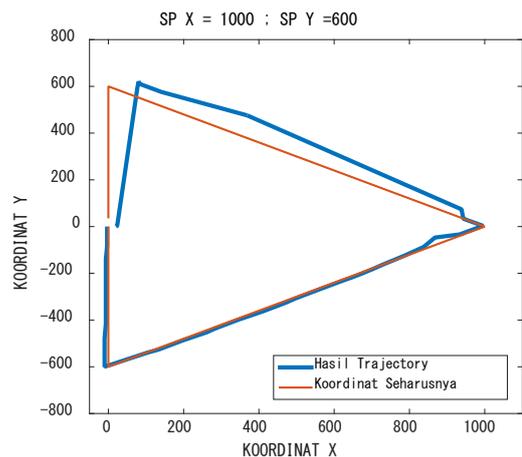


Gbr. 14 Hasil pengujian robot omni saat trajectory berupa persegi panjang.

TABEL X
DATA PENGUJIAN ROBOT OMNI SAAT TRAJECTORY BERUPA PERSEGI PANJANG

Koordinat posisi (mm)				Absolute Error (mm)	
Target		Real			
x	y	x	y	x	y
0	0	0	0	0	0
1.000	0	996	2	4	2
1.000	-500	974	-517	26	17
0	-500	-7	-503	7	3
0	0	1	2	1	2

Gbr. 15 dan Tabel XI menyajikan hasil pengujian robot omni saat trajectory berupa segitiga sama kaki. Pada hasil pengujian trajectory segitiga sama kaki, hasil yang didapatkan

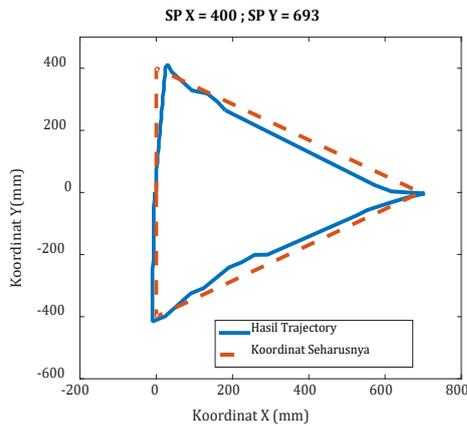


Gbr. 15 Hasil pengujian robot omni saat trajectory berupa segitiga sama kaki.

TABEL XI
DATA PENGUJIAN ROBOT OMNI SAAT TRAJECTORY BERUPA SEGITIGA SAMA KAKI

Koordinat posisi (mm)				Absolute Error (mm)	
Target		Real			
x	y	x	y	x	y
0	0	0	0	0	0
0	-600	10	-599	10	1
1.000	0	995	3	5	3
0	600	77	630	77	30
0	0	20	0	20	0

yaitu pergerakan robot menunjukkan error maksimum sebesar 50 mm pada koordinat x, dan error sebesar 70 mm pada koordinat y.



Gbr. 16 Hasil pengujian robot omni saat trajectory berupa segitiga sama sisi.

TABEL XII

DATA PENGUJIAN ROBOT OMNI SAAT TRAJECTORY BERUPA SEGITIGA SAMA SISI

Koordinat posisi (mm)				Absolute Error (mm)	
Target		Real			
x	y	x	y	x	y
0	0	0	0	0	0
0	-400	10	-410	10	10
693	0	710	2	17	2
0	400	20	405	20	5
0	0	3	3	3	3

Gbr. 16 dan Tabel XII menyajikan hasil pengujian robot omni saat trajectory berupa segitiga sama sisi. Pada hasil pengujian trajectory segitiga sama sisi, hasil yang didapatkan yaitu pergerakan robot menunjukkan error maksimum sebesar 50 mm pada koordinat x , dan error sebesar 70 mm pada koordinat y .

V. KESIMPULAN

Dari hasil pengamatan, pengujian, dan analisis, dapat ditarik beberapa kesimpulan. Makalah ini menyajikan perancangan robot omni baik secara perangkat keras maupun perangkat lunak dengan tujuan utama penerapan odometry robot omni dalam hal trajectory tracking. Dari hasil percobaan, diperoleh tingkat error rata-rata koordinat x sebesar 1,75% dan koordinat y sebesar 4,71% saat robot omni mengikuti lintasan trajectory berupa persegi panjang dengan ukuran panjang 1.000 mm dan lebar 500 mm. Selain itu, makalah ini juga bertujuan untuk mencari nilai regresi kecepatan masing-masing motor pada robot omni. Untuk menjaga kestabilan kecepatan putar motor pada robot omni, kendali PID telah diterapkan. Dari pengujian yang telah dilakukan, nilai konstanta PID yang digunakan pada motor 1 adalah $K_p = 1,5$, $K_i = 0,0018$, dan $K_d = 30$. Pada motor 2 $K_p = 2$, $K_i = 0,0024$, dan $K_d = 30$. Pada motor 3, $K_p = 1,5$, $K_i = 0,003$, dan $K_d = 30$. Pada motor 4, $K_p = 2,3$, $K_i = 0,004$, dan $K_d = 50$. Hasil pengujian odometry dari robot omni ini telah dipublikasikan pada kanal video YouTube pada alamat [9].

UCAPAN TERIMA KASIH

Diucapkan terima kasih kepada Departemen Teknik Elektro dan Informatika Sekolah Vokasi Universitas Gadjah Mada atas dukungan yang telah diberikan.

REFERENSI

- [1] R. Rojas dan A.G. Förster, "Holonomic Control of a Robot with an Omnidirectional Drive," *KI-Künstliche Intelligenz*, Vol. 20, No. 2, hal. 12-17, 2006.
- [2] S. Chamberland, É. Beaudry, L. Clavier, F. Kabanza, F. Michaud, dan M. Lauriay, "Motion Planning for an Omnidirectional Robot with Steering Constraints," *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, hal. 4305-4310.
- [3] Fahmizal, M. Arrofiq, dan A. Mayub. "Identifikasi Pemodelan Matematis Robot Wall Following," *Jurnal Nasional Teknik Elektro dan Teknologi Informasi (JNTETI)*, Vol. 7, No. 1, hal. 79-88, 2018.
- [4] K. Ohno, T. Tsubouchi, B. Shigematsu, S. Maeyama, dan S. Yuta. "Outdoor Navigation of a Mobile Robot Between Buildings Based on DGPS and Odometry Data Fusion," *Proc. of 2003 IEEE International Conference on Robotics and Automation*, 2003, hal. 1978-1984.
- [5] J.P. Laumond, S. Sekhavat, dan F. Lamiraux. "Guidelines in Nonholonomic Motion Planning for Mobile Robots," in *Robot Motion Planning and Control*, J.P. Laumond, Ed. Berlin, Germany: Springer, 1998, hal. 1-53.
- [6] L. Huang, Y.S. Lim, D. Lee, dan C.E. Teoh, "Design and Analysis of a Four-wheel Omnidirectional Mobile Robot," *2nd International Conference on Autonomous Robots and Agents*, 2004, hal. 425-428.
- [7] K.S. Chong, dan L. Kleeman, "Accurate Odometry and Error Modelling for a Mobile Robot," *Proc. of the 1997 IEEE International Conference on Robotics and Automation*, 1997, Vol. 4, hal. 2783-2788.
- [8] Borenstein, Johann, and Liqiang Feng. "Measurement and Correction of Systematic Odometry Errors in Mobile Robot," *IEEE Transactions on Robotics and Automation*, Vol. 12, No. 6, hal. 869-880, 1996.
- [9] F. Fahmi (2018) Video implementation Trajectory Tracking Robot Omni, [Online] <https://youtu.be/7qi14OtCou4>, tanggal akses: 20-Jul-2018.