

# Penerapan *Inverse Kinematics* pada Pengendalian Gerak Robot Lego

Riki Ria<sup>1\*</sup>, Senanjung Prayoga<sup>1</sup>

<sup>1</sup>Teknik Mekatronika, Politeknik Negeri Batam, Batam, Indonesia

Email: rikiria@polibatam.ac.id

**Abstrak**—Dalam sebuah kompetisi robot Lego, diperlukan sebuah bentuk robot yang dapat dibangun untuk dapat digunakan menyelesaikan keseluruhan tugas dengan cepat. Penggunaan kinematika robot sangat penting untuk mendefinisikan arah dan kecepatan robot agar robot dapat melakukan pergerakan yang akurat serta sesuai dengan titik koordinat yang telah ditentukan. Permasalahan yang muncul adalah bagaimana persamaan *inverse kinematics* yang ada dapat membuat robot Lego Mindstorm EV3 bergerak mengikuti perencanaan gerak yang telah dibuat. Penelitian ini menggunakan persamaan *inverse kinematics* pada robot *differential wheel* dengan menggunakan kontrol *Proporsional Integral Derivatif* (PID) dan regresi linier. Hasil percobaan menunjukkan bahwa robot memiliki pergerakan yang baik mengikuti jalur terpendek dengan parameter kontrol PID:  $K_p=1$ ,  $K_i=0,3$ , dan  $K_d=1$ . Rata-rata *error* kecepatan yang dihasilkan adalah 1,5795% dan rata-rata *error* posisi adalah 12,22%.

**Kata kunci:** *Lego Mindstorms EV3, differential wheels, inverse kinematics*

## I. PENDAHULUAN

PADA saat pertandingan, pergerakan robot di lapangan atau ruang pertandingan dapat dikendalikan secara otomatis maupun manual. Jika robot dikendalikan secara otomatis, robot harus dapat berjalan sesuai dengan rencana atau masukan yang diberikan. Hal ini menjadi penting apabila robot akan menjalankan tugas-tugas dari satu titik ke titik lain yang telah ditentukan sesuai dengan peraturan pertandingan. Robot harus memiliki suatu sistem yang dapat mengendalikan serta menerjemahkan koordinat posisi maupun masukan lainnya yang diberikan untuk dijalankan. Dalam kit robot Lego Mindstorms, robot hanya memiliki dua roda sehingga lebih sering menggunakan sistem pergerakan *differential drive*. *Differential drive* merupakan salah satu pergerakan robot yang memanfaatkan kecepatan pada roda kiri dan kanannya. Permasalahan utama dari *differential drive* adalah terbatasnya pergerakan robot, yakni tidak mampu bergerak ke segala arah atau biasa disebut robot *nonholonomic*.

Dari beberapa penelitian yang sudah dilakukan, penggunaan

kinematika robot sangat penting untuk mendefinisikan arah dan kecepatan robot. Metode yang digunakan untuk mengendalikan robot yang menggunakan *differential drive* adalah *inverse kinematic*, yang mana metode ini merupakan model matematis dari pergerakan suatu sistem mekanis tanpa mempertimbangkan adanya gaya yang memengaruhi gerakan tersebut. Selain sistem *feedback* yang tepat, robot juga harus memiliki sistem mekanis yang mendukung agar pergerakan robot dapat berjalan dengan baik [1].

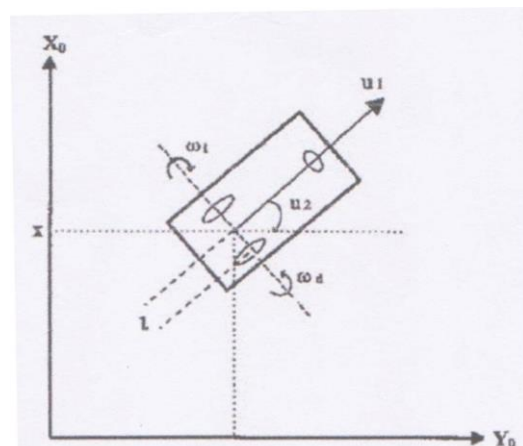
Penelitian ini menggunakan metode *inverse kinematics*. Robot dapat diberi masukan berupa posisi maupun koordinat yang diinginkan sehingga tidak mengikuti jalur sebagaimana robot *line follower*. Dari beberapa titik koordinat tersebut, perencanaan gerak robot dapat dibuat. Kontrol PID dan regresi linier ditambahkan untuk memperhalus pergerakan motor.

## II. METODE

### A. Persamaan Invers Kinematics

Seperti halnya metode-metode pergerakan dari beberapa jenis robot, untuk robot *differential drive* sendiri memiliki rumus persamaan untuk mendapatkan pergerakan yang diinginkan. *Frame* robot dalam proyeksi sumbu X dan Y dapat dilihat pada Gambar 1.

Persamaan (1), (2) dan (3) dapat digunakan untuk menentukan kecepatan sudut roda kanan ( $\omega_d$ ) dan kecepatan sudut roda kiri ( $\omega_i$ ) dari sebuah robot *differential drive* dengan



Gambar 1. *Frame* robot dalam proyeksi sumbu X dan Y

hanya menentukan  $\dot{x}$ ,  $\dot{y}$ ,  $\dot{\phi}$  dan  $\phi$  [2].

$$\begin{pmatrix} \omega d \\ \omega i \end{pmatrix} = \begin{pmatrix} \frac{-2}{r \sin \phi} & \frac{-1}{\cos \phi} & \frac{b}{2r} \\ \frac{-2}{r \sin \phi} & \frac{-1}{\cos \phi} & \frac{-b}{2r} \end{pmatrix} \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{pmatrix} \quad (1)$$

$$\omega d = -\frac{2}{r \sin \phi} \dot{x} - \frac{1}{r \cos \phi} \dot{y} + \frac{b}{2r} \dot{\phi} \quad (2)$$

$$\omega i = -\frac{2}{r \sin \phi} \dot{x} - \frac{1}{r \cos \phi} \dot{y} - \frac{b}{2r} \dot{\phi} \quad (3)$$

Keterangan :

- $\omega d$  = Rotational velocity (deg/s)
- $\omega i$  = Rotational velocity (deg/s)
- $\dot{x}$  = Position velocity (m/s)
- $\dot{y}$  = Position velocity (m/s)
- $\dot{\phi}$  = Rotational velocity (deg/s)
- $r$  = Wheel radius (m)
- $b$  = Distance between to wheel (m)
- $\phi$  = Robot angle (deg)

**B. Kontrol PID**

Kendali PID merupakan salah satu jenis kontrol yang banyak digunakan terutama dalam industri proses. Dalam suatu sistem kontrol kita mengenal adanya beberapa macam kontrol, di antaranya adalah kontrol proporsional, kontrol integral, dan kontrol derivatif. Masing-masing aksi kontrol ini memiliki kelebihan-kelebihan tertentu, di mana aksi kontrol proporsional memiliki keunggulan *rise time* yang cepat, aksi kontrol integral memiliki kelebihan memperkecil *error*, dan aksi kontrol derivatif mempunyai keunggulan untuk memperkecil *error* dan meredam *overshoot/undershoot*. Untuk itu agar kita dapat menghasilkan *output* dengan *risetime* yang cepat dan *error* yang kecil, maka kita dapat menggabungkan ketiga aksi kontrol ini menjadi kontrol PID [3]. Persamaan (4), (5), (6), dan (7) merupakan rumus umum yang digunakan dalam matematis mencari parameter PID.

$$P_{out} = K_p e(t) \quad (4)$$

$$I_{out} = K_i \int_0^t e(t) dt \quad (5)$$

$$D_{out} = K_d \frac{d}{dt} e(t) \quad (6)$$

$$MV(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t) \quad (7)$$

di mana :  $P_{out}$  = Proporsional       $I_{out}$  = Integral  
 $D_{out}$  = Derivatif               $e$  = Error  
 $K_p$  = konstanta Proporsional  
 $K_i$  = konstanta integral

$K_d$  = konstanta derivatif  
 $t$  = waktu

Tabel I menggambarkan kelebihan dan kekurangan masing-masing setiap aksi kontrol.

**C. Perancangan Pergerakan Robot**

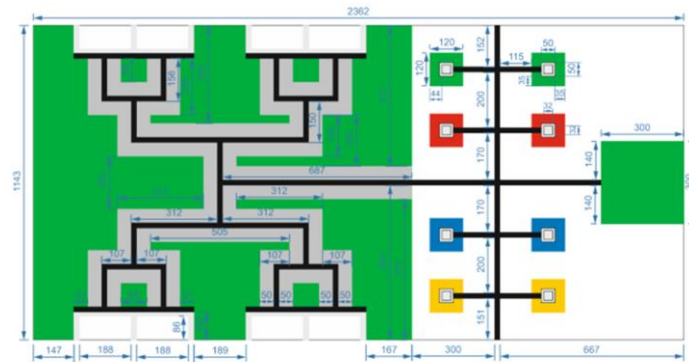
Pada bagian ini akan menjelaskan rencana yang akan dibuat untuk pergerakan dari robot sehingga dapat menyelesaikan misi dengan cepat tanpa harus mengikuti garis (*line follower*). Ukuran dari lapangan yang akan dilewati oleh robot dapat dilihat pada Gambar 2, sedangkan rencana pergerakan robot dijelaskan pada Gambar 3.

**D. Perancangan Hardware**

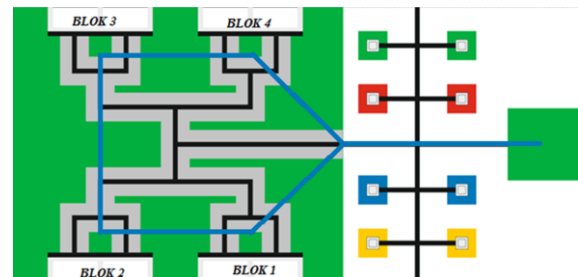
Robot *Lego differential wheel* dapat menggunakan dua motor DC yang berukuran besar dengan jarak antara kedua rodanya 0.102 m dengan diameter roda 56 mm. Sedangkan ukuran panjang, dan tinggi dari robot adalah masing-masing 20 cm, 18 cm, dan 15 cm. Perancangan robot dan peletakan motor penggerak dan *brick* pada robot tersebut dapat dilihat

TABEL I  
 KARAKTERISTIK ANTARA PROPORSIONAL, INTEGRAL, DAN DERIVATIF

Tanggapan Loop Tertutup	Waktu Naik	Overshoot	Waktu Turun	Kesalahan Keadaan Tunak
Proportional (Kp)	Menurun	Meningkat	Perubahan Kecil	Menurun
Integral (Ki)	Menurun	Meningkat	Meningkat	Hilang
Derivative (Kd)	Perubahan Kecil	Menurun	Menurun	Perubahan Kecil



Gambar 2. Dimensi lapangan



Gambar 3. Rencana pergerakan dari robot

pada Gambar 4.

E. Diagram Blok Sistem

Gambar 5 merupakan diagram blok dari sistem yang dibuat. Interface komputer dengan controller menggunakan Bluetooth. Setelah program di-upload dalam controller, controller tersebut akan mengirim data ke motor penggerak 1 dan 2 yang memiliki feedback dari sensor encoder yang ada pada setiap motor untuk mengetahui pergerakan dari motor. Dengan demikian, motor dapat dikontrol pergerakannya.

F. Perencanaan Software

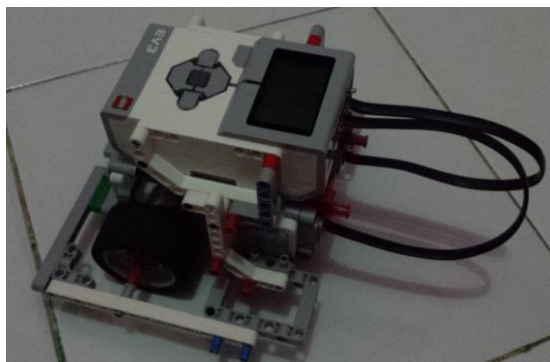
Gambar 6 merupakan alur dari pemrograman yang berjalan pada robot. Proses pertama inialisasi keseluruhan proses, kemudian dilanjutkan proses *inverse kinematics*, kontrol PID, jarak setiap *check point*, dan regresi linier. Proses program berjalan berapa kali untuk menyelesaikan semua *check point*. Ketika mendapat kecepatan, masing-masing motor dari proses *invers kinematics* akan langsung masuk kedalam proses regresi linier, mengubah kecepatan (deg/s) menjadi satuan persentase dari power masing-masing motor. Nilai hasil dari proses regresi linier ini akan menjadi set poin dari masing-masing proses kontrol PID. Gambar 7 dan Gambar 8 menjelaskan program dari proses *invers Kinematics* dan proses dari kontrol PID.

III. HASIL DAN ANALISIS

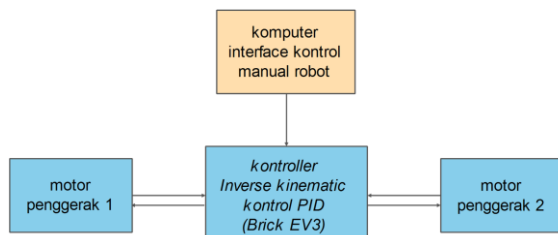
Pengujian dilakukan meliputi pengujian *driver* motor, pengujian parameter kontrol PID, dan pengujian *invers kinematics*.

A. Pengujian Driver Motor

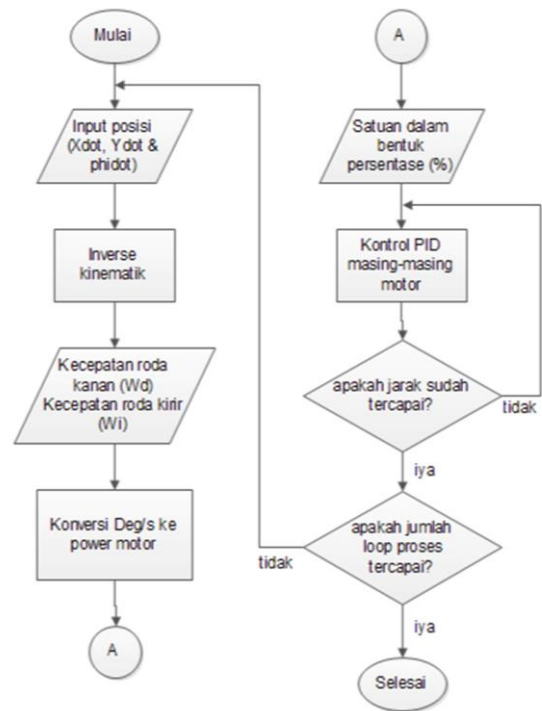
Pada tahap ini dilakukan pengujian terhadap masing-masing motor. Motor diberi masukan *power* 2% - 99% dan dilihat kecepatan putar yang dihasilkan (deg/s), setelah itu dibuat



Gambar 4. Desain Robot Lego



Gambar 5. Diagram blok



Gambar 6. Blok diagram perangkat lunak

persamaannya. Gambar 9 dan Gambar 10 merupakan grafik masing-masing motor di mana masukan persentase power motor dari 2% - 99% (sumbu X) terhadap kecepatan motor berputar dalam satuan *deg/s* (sumbu Y). Sehingga untuk grafik motor pertama didapat sebuah persamaan polinomial sebagai berikut :

$$y = -0,0000000001846x^4 + 0,0000003196589x^3 - 0,0001447279198x^2 + 0,1237670032253x - 1,7$$

Sedangkan untuk persamaan polinomial dari motor kedua adalah sebagai berikut:

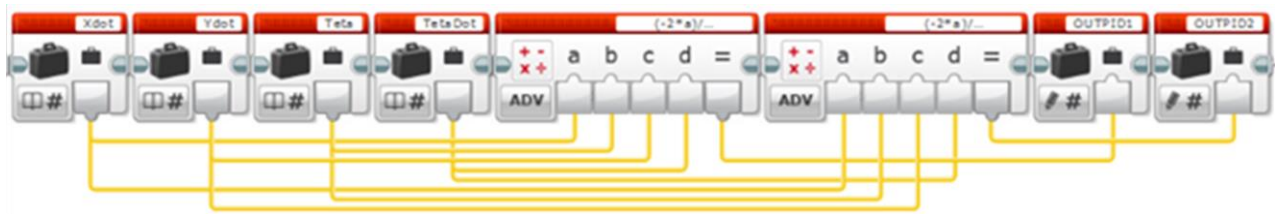
$$y = -0,000000000006668x^5 + 0,000000013x^4 - 0,000000873x^3 + 0,00027x^2 + 0,068x + 0,26$$

Ada perbedaan antara hasil pengukuran dengan perhitungan. Perbedaan ini terjadi karena nilai-nilai komponen yang digunakan dalam driver motor memiliki nilai toleransi.

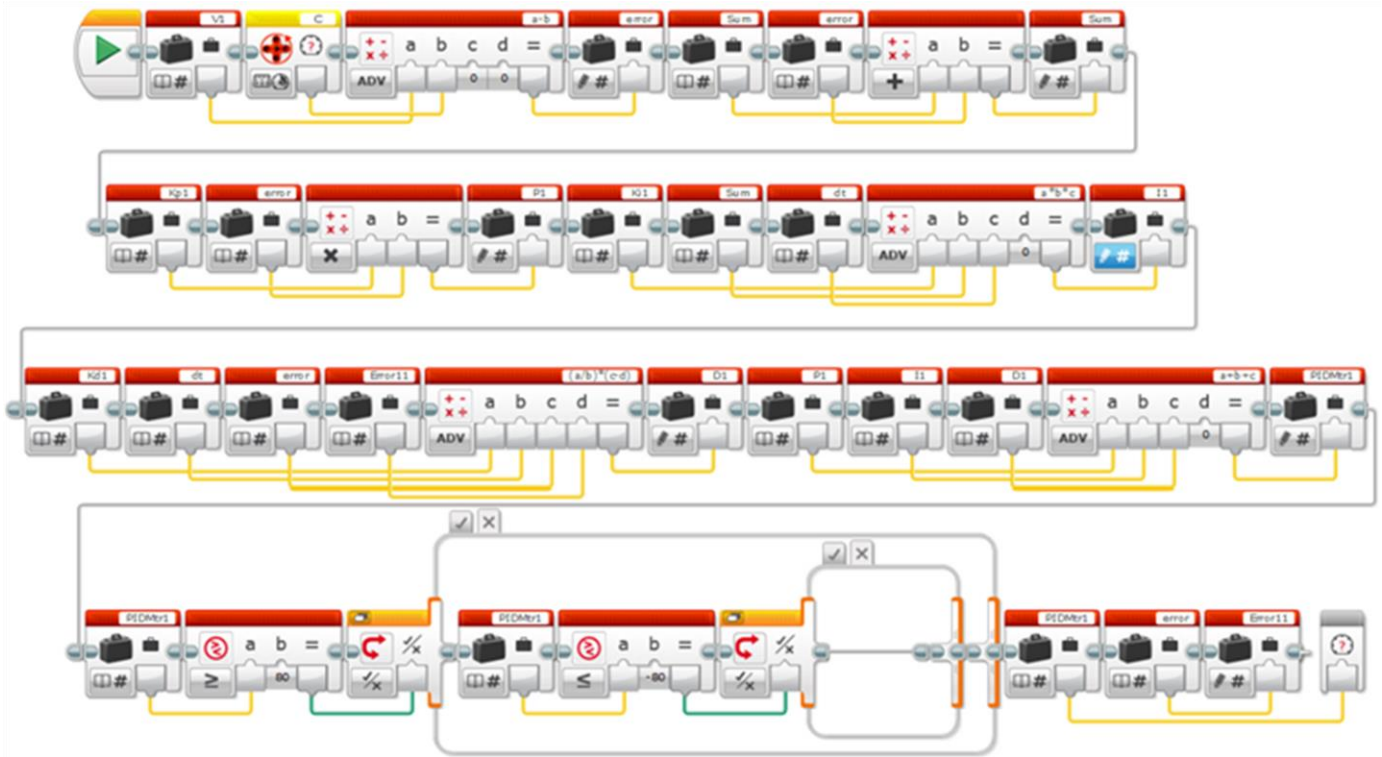
B. Pengujian Parameter PID

Tahap ini menguji parameter kontrol PID dari masing-masing motor. Pengujian ini menggunakan metode *tuning* manual. Pengujian yang telah dilakukan menghasilkan grafik dari hasil kendali PID yang ditunjukkan pada Gambar 11 dan Gambar 12. Kedua gambar tersebut memperlihatkan bahwa *overshoot* yang dihasilkan kecil dan *steady state* yang dihasilkan pun relatif stabil. Meskipun masih mamiliki *rise time* yang cukup besar, namun nilai *error* yang dihasilkan tidak terlalu besar.

C. Pengujian Persamaan Inverse Kinematics pada Robot



Gambar 7. Program *invers kinematics*



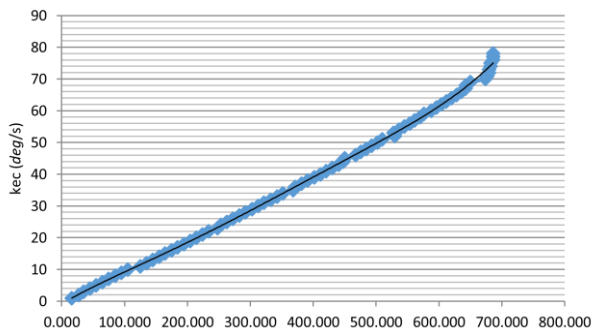
Gambar 8. Program kontrol PID

Pengujian ini merupakan pengujian keseluruhan program untuk menguji apakah robot mengikuti pergerakan sesuai dengan perencanaan yang dijelaskan pada Gambar 3. Hasil dari pengujian pergerakan robot ini ditunjukkan pada Gambar 13 dan Tabel II. Pengukuran nilai *error* diukur setiap kelipatan 10 cm.

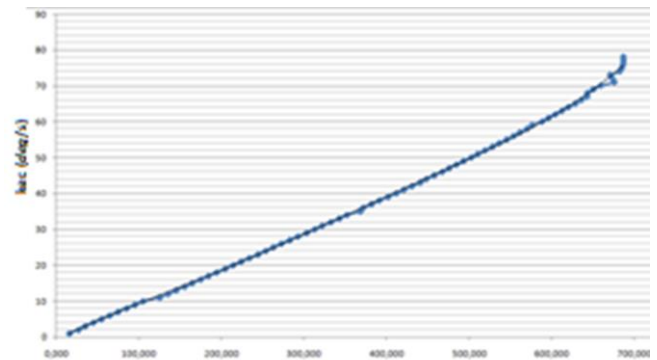
Berdasarkan hasil dan analisis maka dapat disimpulkan adalah sebagai berikut:

1. Berdasarkan hasil pengujian output persamaan linier pada motor, rata-rata *error* pada kedua motor DC adalah  $\pm 0,782\%$ .

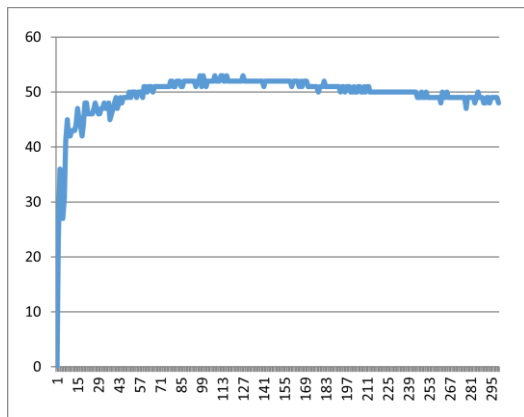
IV. KESIMPULAN



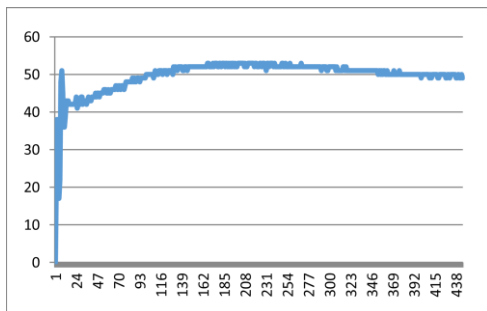
Gambar 9. Grafik persamaan motor pertama



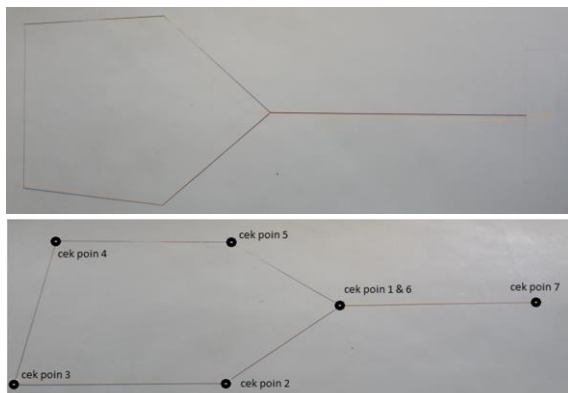
Gambar 10. Grafik persamaan motor kedua



Gambar 11. Grafik PID Motor Pertama Dengan  $K_p = 1$ ;  $K_i = 0.3$ ; dan  $K_d = 1$



Gambar 12. Grafik PID Motor Kedua Dengan  $K_p = 1$ ;  $K_i = 0.3$ ; dan  $K_d = 1$



Gambar 13. Hasil Pergerakan Dari Robot

TABEL II  
PENGUJIAN PERGERAKAN ROBOT

Titik <i>check point</i>	Lintasan ( <i>deg</i> )	Hasil pergerakan robot ( <i>deg</i> )	<i>Error (deg)</i>	<i>Error (%)</i>
1	90	100	10	11,11
2	45	57	12	13,33
3	90	105	15	16,67
4	180	189	9	10,00
5	90	95	5	5,56
6	125	135	10	11,11
7	90	106	16	17,78
Rata-rata <i>error (%)</i>				12,22

- Berdasarkan hasil pengujian parameter kontrol PID yang digunakan masing-masing parameter motor adalah  $K_p = 1$  ;  $K_i = 0,3$  dan  $K_d = 1$ .
- Robot sudah dapat melakukan pergerakan sesuai dengan perencanaan gerak dengan menggunakan *inverse kinematics* dan *control PID* rata-rata *error* robot dalam mencapai posisi yang diinginkan adalah 12,22%.

REFERENSI

- Nova.S Rezky. "Perencanaan Gerak Robot Mecanum *Wheel* Menggunakan Persamaan *Inverse Kinematics*.", Politeknik Negeri Batam, Batam, 2015.
- Giraldi. "Implementasi persamaan *inverse Kinematics* pada robot *differential wheel*.", Politeknik Negeri Batam, Batam, 2015.
- Svendsen Alexander. "RoboMind." Master's Thesis, The Arctic University, Norway, 2014.
- Wijaya.K Johan. "Penerapan *Invers Kinematics* Terhadap Pergerakan Kaki Pada Robot *Hexapoda*." Sarjana Teknik Informatika, STMIK GI MDP, Palembang, 2013.
- Yufan. H Danang. "Penerapan *Invers Kinematics* Pada Pengendalian Gerak Robot." Sarjana Teknik Eletro, Institut Teknologi Sepuluh Nopember, 2012.
- Andy. "Percangan Model Industrial Robot Secara *Kinematics*." Sarjana Ilmu Komputer, Universitas Bina Nusantara, 2009.