



RANCANG BANGUN KONTROL KECEPATAN ROBOT *THREE OMNI-DIRECTIONAL* MENGGUNAKAN METODE PID (Proportional Integral Derivative)

TUGAS AKHIR

Program Studi

S1 Teknik Komputer

INSTITUT BISNIS
DAN INFORMATIKA

stikom
SURABAYA

Oleh:

DIO ARTA KARNADI

15410200037

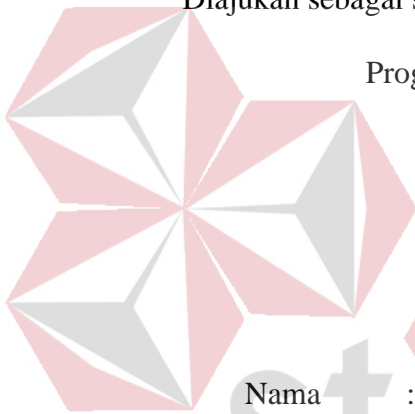
**FAKULTAS TEKNOLOGI DAN INFORMATIKA
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA
2018**

RANCANG BANGUN KONTROL KECEPATAN ROBOT *THREE OMNI-DIRECTIONAL* MENGGUNAKAN METODE PID (Proportional Integral Derivative)

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk menyelesaikan

Program Sarjana Komputer



Oleh:

Nama : DIO ARTA KARNADI

NIM : 15410200037

Program : S1 (Strata Satu)

Jurusan : Teknik Komputer

Fakultas : Teknologi dan Informatika

**FAKULTAS TEKNOLOGI DAN INFORMATIKA
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA**

2018

TUGAS AKHIR

RANCANG BANGUN KONTROL KECEPATAN ROBOT *THREE OMNI-DIRECTIONAL* MENGGUNAKAN METODE PID (Proportional Integral Derivative)

dipersiapkan dan disusun oleh

DIO ARTA KARNADI

NIM : 15410200037

Telah diperiksa, diuji dan disetujui oleh Dewan Pembahas

Pada : 1 April 2018

Susunan Dewan Pembimbing dan Penguji

Pembimbing

I. **Dr. Susijanto Tri Rasmana, S.Kom., M.T.**

NIDN. 0727097302

II. **Musayyanah, S.ST., M.T.**

NIDN. 0730069102

Penguji

I. **Pauladie Susanto, S.Kom., M.T.**

NIDN. 0729047501

Tugas Akhir ini telah diterima sebagai salah satu persyaratan

Untuk memperoleh gelar Sarjana



FAKULTAS TEKNOLOGI
DAN INFORMATIKA

stikom
SURABAYA

Dr. Jusak

FAKULTAS TEKNOLOGI DAN INFORMATIKA

INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA

SURAT PERNYATAAN

PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa Institut Bisnis dan Informatika Stikom Surabaya, saya :

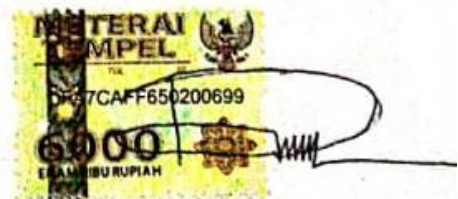
Nama : Dio Arta Karnadi
NIM : 15410200037
Program Studi : S1 Teknik Komputer
Fakultas : Fakultas Teknologi dan Informatika
Jenis Karya : Tugas Akhir
Judul Karya : **RANCANG BANGUN KONTROL KECEPATAN
ROBOT *THREE OMNI-DIRECTIONAL*
MENGUNAKAN METODE PID (*Proportional
Integral Derivative*)**

Menyatakan dengan sesungguhnya bahwa:

1. Demi pengembangan Ilmu Pengetahuan, Teknologi dan Seni, saya menyetujui memberikan kepada Institut Bisnis dan Informatika Stikom Surabaya Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive Royalti Free Right*) atas seluruh isi/ sebagian karya ilmiah saya tersebut di atas untuk disimpan, dialihmediakan dan dikelola dalam bentuk pangkalan data (*database*) untuk selanjutnya didistribusikan atau dipublikasikan demi kepentingan akademis dengan tetap mencantumkan nama saya sebagai penulis atau pencipta dan sebagai pemilik Hak Cipta
2. Karya tersebut di atas adalah karya asli saya, bukan plagiat baik sebagian maupun keseluruhan. Kutipan, karya atau pendapat orang lain yang ada dalam karya ilmiah ini adalah semata hanya rujukan yang dicantumkan dalam Daftar Pustaka saya
3. Apabila dikemudian hari ditemukan dan terbukti terdapat tindakan plagiat pada karya ilmiah ini, maka saya bersedia untuk menerima pencabutan terhadap gelar kesarjanaan yang telah diberikan kepada saya.

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, 1 April 2018



Dio Arta Karnadi
NIM.15410200037



*“ Untuk menjadi orang yang tangguh dalam hidup maka kita perlu
membiasakan diri untuk belajar terpaksa, dipaksa, biasa, bisa, luar biasa”.*

~ Dio Arta Karnadi ~

INSTRUMEN BISNIS
DAN INFORMATIKA
stikom
SURABAYA

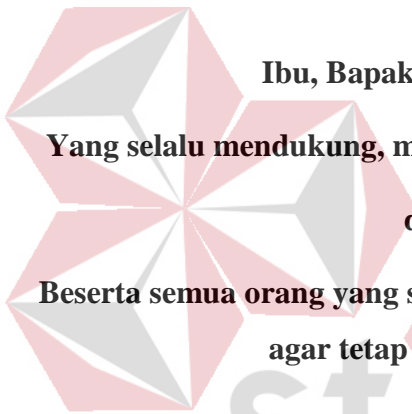
Kupersembahkan Kepada

ALLAH SWT

Ibu, Bapak dan semua keluarga tercinta,

**Yang selalu mendukung, memotivasi dan menyisipkan nama saya dalam
doa-doa terbaiknya.**

**Beserta semua orang yang selalu membantu, mendukung dan memotivasi
agar tetap berusaha menjadi lebih baik.**



stikom
SURABAYA

ABSTRAK

Mobile robot adalah robot yang mampu melakukan perpindahan dari suatu tempat ke tempat lain. Robot dengan jenis ini memiliki ciri khas yaitu robot menggunakan motor dengan roda sebagai aktuatornya. Bentuk roda akan mempengaruhi sistem kerja robot, seperti memilih untuk menggunakan roda *omni-directional* dimana roda jenis ini adalah roda yang bisa bergerak ke segala arah sehingga robot tidak perlu melakukan rotasi terlebih dahulu. Salah satu jenis *mobile robot* yang sering digunakan yaitu jenis robot *three omni directional* yang menggunakan susunan 3 motor membentuk huruf y, Untuk melakukan pergerakan robot memerlukan sistem kendali yang berfungsi untuk mengendalikan posisi dan kecepatan robot. Pada *mobile robot* ini diterapkan sistem kontrol cerdas yaitu menggunakan metode PID (*Proportional Integral Derivatif*).

Dari hasil penelitian yang dilakukan, didapatkan hasil respon pergerakan robot dari berbagai nilai variable K_p , K_i , K_d untuk jarak dan arah pergerakan robot yang berbeda-beda. Untuk pergerakan robot maju nilai variable K_p , K_i , K_d dengan nilai eror kurang dari 30 cm dan waktu tempuh 8.02 detik yaitu $K_p=4$, $K_i=5$, $K_d=4$, Sedangkan untuk pergerakan robot ke arah kanan nilai variable K_p , K_i , K_d dengan nilai eror kurang dari 30 cm dan waktu tempuh 08.4 detik yaitu $K_p=4$, $K_i=5$, $K_d=4$, untuk pergerakan diagonal pada robot nilai variable K_p , K_i , K_d dengan nilai eror kurang dari 30 cm dan waktu tempuh pada rentang 8.02 detik yaitu $K_p=4$, $K_i=5$, $K_d=4$.

Kata kunci: *Mobile robot*, *omni-directional*, PID(*Proportional Interral Derivative*).

KATA PENGANTAR

Segala puji syukur kehadirat Allah SWT, karena dengan rahmatnya dan hidayah-Nya penulis dapat menyelesaikan penyusunan Laporan Tugas Akhir yang berjudul “RANCANG BANGUN KONTROL KECEPETAN ROBOT *THREE OMNI-DIRECTIONAL* MENGGUNAKAN METODE PID (Proportional Integral Derivative)”. Laporan Tugas Akhir ini disusun dalam rangka penulisan laporan untuk memperoleh gelar Sarjana Komputer pada program studi S1 Teknik Komputer Stikom Surabaya.

Pada kesempatan ini, penulis ingin mengucapkan rasa terima kasih kepada pihak-pihak yang memberi dukungan dan masukan dalam menyelesaikan laporan Tugas Akhir ini. Oleh karena itu penulis ingin mengucapkan kepada.

1. Allah SWT yang telah memberikan kesehatan sehingga penulis dapat menyelesaikan Laporan Tugas Akhir ini.
2. Orang Tua dan Saudara-saudara saya tercinta yang telah memberikan dorongan dan bantuan baik moral maupun materi sehingga penulis dapat menempuh dan menyelesaikan Tugas Akhir ini.
3. Bapak Dr. Susijanto Tri Rasmana, S.Kom., M.T. dan Ibu Mussayyanah, S.ST., M.T. selaku Dosen Pembimbing yang selalu memberi arahan dan bimbingan dalam menyelesaikan Tugas Akhir beserta laporan ini.
4. Bapak Pauladie Susanto, S.Kom., M.T. Selaku Kepala Program Studi S1 Teknik Komputer yang selalu memberikan dukungannya dalam menyelesaikan Tugas Akhir ini.
5. Rekan-rekan Komunitas Stikom Robotik yang memberikan motivasi serta bantuan dalam penyelesaian Tugas Akhir ini.

6. Teman- teman seperjuangan program studi Teknik Komputer angkatan 2015 dan semua pihak yang terlibat namun tidak dapat penulis sebutkan satu persatu atas bantuan dan dukungannya.

Penulis menyadari bahwa Laporan Tugas Akhir ini jauh dari kata sempurna, masih banyak kekurangan dalam menyusun laporan ini. Oleh karena itu dalam kesempatan ini, penulis meminta maaf apabila dalam Laporan Tugas Akhir ini masih banyak kesalahan baik dalam penulisan maupun Bahasa yang digunakan. Penulis juga memerlukan kritik dan saran dari para pembaca yang sifatnya membangun untuk kesempurnaan laporan yang telah penulis susun.

Surabaya, 1 April 2018



DAFTAR ISI

	Halaman
ABSTRAK	ii
KATA PENGANTAR	iii
DAFTAR GAMBAR	xi
BAB I	xiii
1.1 Latar Belakang	14
1.2 Perumusan Masalah	17
1.3 Batasan Masalah.....	17
1.4 Tujuan	18
1.5 Sistematika Penulisan	18
BAB II.....	20
2.1 Mobile Robot	20
2.2 Robot <i>Holonomic</i>	21
2.3 Roda Omni-Directional.....	22
2.4 Motor DC	23
2.5 External Rotary Encoder	23
2.5.1 Datasheet ES403-100-3-T-24.....	25
2.6 Optocoupler.....	25
2.7 Motor Driver EMS 30A	27
2.8 LCD (Liquid Crystal Display)	31

2.9	Universal Battery Elimination Circuit (UBEC).....	33
2.10	Pulse Width Modulation (PWM).....	34
2.11	PID (Proportional Integral Derivatif).....	36
2.11.1	Proporsional	36
2.11.2	Integral	38
2.11.3	Derivative.....	38
2.12	Arduino IDE.....	40
BAB III		43
3.1	Perancangan sistem.....	43
3.1.1	Perancangan perangkat keras	44
3.1.2	Alokasi pin Arduino due	46
3.1.3	Rangkaian tombol	49
3.1.4	LCD 16x2.....	50
3.1.5	Rangkaian optocoupler switch.....	51
3.1.6	Rangkaian driver dan motor dc.....	52
3.2.	Perancangan perangkat lunak.....	53
3.2.1	Blok diagram sistem.....	55
3.2.2	Kontrol Proportional Integral Derivatif (PID)	56
3.3	Perancangan mekanik.....	58
3.3.1	Perancangan sistem actuator robot.....	58
3.3.2	Pembuatan <i>Frame</i> Robot.....	59

3.3.3	Pembuatan <i>Shaft</i> motor	60
3.3.4	Pembuatan Hub dan As untuk <i>Rotary Encoder</i>	61
BAB IV	62
4.1	Pengujian Arduino DUE	63
4.1.1	Tujuan	63
4.1.2	Alat yang digunakan	63
4.1.3	Prosedur pengujian.....	63
4.1.4	Hasil Pengujian	64
4.2	Pengujian tombol input	64
4.2.1	Tujuan	64
4.2.2	Alat yang digunakan	64
4.2.3	Prosedur pengujian.....	65
4.2.4	Hasil pengujian.....	65
4.3	Pengujian LCD.....	66
4.3.1	Tujuan	66
4.3.2	Alat yang digunakan	66
4.3.3	Prosedur pengujian.....	66
4.3.4	Hasil pengujian.....	67
4.4	Pengujian external rotary encoder.....	67
4.4.1	Tujuan	67
4.4.2	Alat yang digunakan	68

4.4.3	Prosedur pengujian.....	68
4.4.4	Hasil pengujian.....	68
4.5	Pengujian kecepatan motor	70
4.5.1	Tujuan	70
4.5.2	Alat yang digunakan	70
4.5.3	Prosedur pengujian.....	70
4.5.4	Hasil pengujian.....	71
4.6	Pengujian sistem keseluruhan	73
4.6.1	Tujuan	73
4.6.2	Alat yang digunakan	73
4.6.3	Prosedur pengujian.....	74
BAB V	164
5.1	Kesimpulan	164
5.2	Saran.....	165
DAFTAR PUSTAKA	166
LAMPIRAN	Error! Bookmark not defined.

DAFTAR TABEL

	Halaman
Tabel 2. 1 Pin Interface Header.....	28
Tabel 2. 2 Pin Power & Motor Conector	30
Tabel 2. 3 Konfigurasi pin LCD 16x2	31
Tabel 3. 1 alokasi pin Arduino due	47
Tabel 4. 1 Hasil pengujian motor	71
Tabel 4. 2 Tabel pembagian kategori	79
Tabel 4. 3 Tabel pengujian menggunakan nilai Kp=4, Ki=5, Kd=4.....	86
Tabel 4. 4 Tabel pengujian menggunakan nilai Kp=4, Ki=9, Kd=4.....	88
Tabel 4. 5 Tabel pengujian menggunakan nilai Kp=7, Ki=5, Kd=2.....	91
Tabel 4. 6 Tabel pengujian menggunakan nilai Kp=6, Ki=5, Kd=2.....	94
Tabel 4. 7 Tabel pengujian menggunakan nilai Kp=4, Ki=5, Kd=4.....	98
Tabel 4. 8 Tabel pengujian menggunakan nilai Kp=4, Ki=9, Kd=4.....	100
Tabel 4. 9 Tabel pengujian menggunakan nilai Kp=6, Ki=8, Kd=3.....	102
Tabel 4. 10 Tabel pengujian menggunakan nilai Kp=4, Ki=5, Kd=4.....	105
Tabel 4. 11 Tabel pengujian menggunakan nilai Kp=4, Ki=9, Kd=4.....	106
Tabel 4. 12 Tabel pengujian menggunakan nilai Kp=5, Ki=8, Kd=2.....	108
Tabel 4. 13 Tabel pengujian menggunakan nilai Kp=6, Ki=5, Kd=3.....	110
Tabel 4. 14 Tabel pengujian menggunakan nilai Kp=4, Ki=5, Kd=4.....	113
Tabel 4. 15 Tabel pengujian menggunakan nilai Kp=4, Ki=9, Kd=4.....	115
Tabel 4. 16 Tabel pengujian menggunakan nilai Kp=6, Ki=3, Kd=6.....	117
Tabel 4. 17 Tabel pengujian menggunakan nilai Kp=8, Ki=1, Kd=8.....	121
Tabel 4. 18 Tabel pengujian menggunakan nilai Kp=4, Ki=5, Kd=4.....	124

Tabel 4. 19 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$	125
Tabel 4. 20 Tabel pengujian menggunakan nilai $K_p=7$, $K_i=8$, $K_d=2$	128
Tabel 4. 21 Tabel pengujian menggunakan nilai $K_p=5$, $K_i=8$, $K_d=7$	130
Tabel 4. 22 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$	132
Tabel 4. 23 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$	133
Tabel 4. 24 Tabel pengujian menggunakan nilai $K_p=5$, $K_i=8$, $K_d=7$	135
Tabel 4. 25 Tabel pengujian menggunakan nilai $K_p=7$, $K_i=8$, $K_d=2$	137
Tabel 4. 26 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$	139
Tabel 4. 27 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$	141
Tabel 4. 28 Tabel pengujian menggunakan nilai $K_p=6$, $K_i=3$, $K_d=6$	143
Tabel 4. 29 Tabel pengujian menggunakan nilai $K_p=7$, $K_i=3$, $K_d=3$	145
Tabel 4. 30 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$	147
Tabel 4. 31 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$	149
Tabel 4. 32 Tabel pengujian menggunakan nilai $K_p=5$, $K_i=3$, $K_d=7$	150
Tabel 4. 33 Tabel pengujian menggunakan nilai $K_p=7$, $K_i=8$, $K_d=2$	152
Tabel 4. 34 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$	154
Tabel 4. 35 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$	155
Tabel 4. 36 Tabel pengujian menggunakan nilai $K_p=5$, $K_i=3$, $K_d=6$	157
Tabel 4. 37 Tabel pengujian menggunakan nilai $K_p=6$, $K_i=6$, $K_d=3$	158
Tabel 4. 38 Pemilihan nilai K_p , K_i , K_d gerak maju.....	161
Tabel 4. 39 Pemilihan nilai K_p , K_i , K_d pergerakan ke arah kanan.....	162
Tabel 4. 40 Pemilihan nilai K_p , K_i , K_d pergerakan ke arah kanan.....	163

DAFTAR GAMBAR

	Halaman
Gambar 2. 1 Mobile Robot	20
Gambar 2. 2 Robot Holonomic	21
Gambar 2. 3 Roda Omni-Directional	22
Gambar 2. 4 Motor DC	23
Gambar 2. 5 External Rotary Encoder	24
Gambar 2. 6 Datasheet ES403-100-3-T-24.....	25
Gambar 2. 7 Keterangan ES403-100-3-T-24	25
Gambar 2. 8 Tipe kabel ES403-100-3-T-24	25
Gambar 2. 9 Optocoupler	26
Gambar 2. 10 Motor Driver EMS 30A	28
Gambar 2. 11 Pin Motor Driver EMS 30A.....	31
Gambar 2. 12 LCD 16x2 (Liquid Crystal Display).....	32
Gambar 2. 13 UBEC – Universal Battery Elimination Circuit.....	33
Gambar 2. 14 PWM (Pulse Width Modulation)	34
Gambar 2. 15 Blok Diagram PID.....	36
Gambar 2. 16 Arduino IDE.....	40
Gambar 2. 17 Interface Arduino IDE.....	41
Gambar 3. 1 Blok Diagram perangkat keras	44
Gambar 3. 2 Pin Arduino DUE	47
Gambar 3. 3 Rangkaian Tombol	49
Gambar 3. 4 Rangkaian LCD 16x2.....	50
Gambar 3. 5 Rangkain optocoupler switch	51

Gambar 3. 6 Rangkaian driver dan motor dc	52
Gambar 3. 7 flowchart keseluruhan	53
Gambar 3. 8 Algoritma kondisi idle (siaga).....	54
Gambar 3. 9 Algoritma kondisi bergerak.....	54
Gambar 3. 10 blok diagram sistem	55
Gambar 3. 11 Diagram blok kontrol PID.....	57
Gambar 3. 12 Desain frame pada sketup 2016	58
Gambar 3. 13 Bahan pembuatan frame robot	59
Gambar 3. 14 Frame Robot yang sudah dirangkai.....	60
Gambar 3. 15 Pemasangan Shaft roda dan AS roda	61
Gambar 3. 16 Pemasangan Roda Omni dengan Rotary Encoder.....	61
Gambar 4. 1 done uploading	64
Gambar 4. 2 program sederhana tombol pada arduino	65
Gambar 4. 3 Tampilan pada serial monitor.....	66
Gambar 4. 4 Program LCD	67
Gambar 4. 5 Hasil pengujian LCD.....	67
Gambar 4. 6 Program sederhana external rotary.....	69
Gambar 4. 7 Hasil pengujian external rotary	69
Gambar 4. 8 Grafik PWM dan RPM Motor.....	72
Gambar 4. 9 Tabel pengamatan	80
Gambar 4. 10 Gerak robot maju.....	76
Gambar 4. 11 Gerak robot kearah kanan	77
Gambar 4. 12 Gerak robot serong kearah kanan.....	78

DAFTAR LAMPIRAN

Lampiran 1 Program robot **Error! Bookmark not defined.**



BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan perkembangan zaman dan kebutuhan manusia juga meningkat, sistem industrialisasi berbasis teknologi yang semakin maju membuat banyak orang berinovasi untuk membangun alat yang memiliki sistem cerdas dan efisien tinggi salah satunya dalam bidang robotika. Dalam melakukan suatu pembangunan robot banyak sekali tahapan-tahapan yang harus dilakukan salah satunya melakukan pemilihan dimensi robot, sensor, aktuator dan mikrokontroler. Robot adalah perangkat yang digunakan sebagai alat bantu manusia dalam menangani berbagai masalah yang telah ditemui sehingga manusia dapat menyelesaikan masalah tersebut secara efisien. Dalam sistem kerjanya robot tergolong menjadi dua kategori yaitu robot manual yang bergerak secara manual yaitu dengan cara dikendalikan oleh manusia melalui remot kontrol atau robot otomatis yang bergerak secara otomatis berdasarkan sensor dan perintah (program) yang ditanamkan pada mikrokontroler robot tersebut.

Pada umumnya robot digunakan oleh bidang industri untuk membantu melakukan pekerjaan manusia yang cenderung kurang efisien karena keterbatasan penglihatan, pergerakan maupun pemilihan pada suatu benda. Fungsi robot adalah untuk memaksimalkan kinerja yang dilakukan oleh manusia dalam sebuah sistem industri sebagai gantinya manusia bertugas untuk mengawasi kinerja dari robot tersebut. Salah satu faktor yang dapat mempengaruhi perkembangan robot di Indonesia ialah dengan adanya penyelenggaraan kompetisi robot Indonesia dengan

berbagai macam tema salah satunya adalah Kontes Robot Sepak Bola Beroda Indonesia. Untuk pertandingan sepak bola beroda masih dikategorikan kompetisi baru karena baru diadakan dua kali yaitu pada tahun 2017 dan 2018, kompetisi robot sepak bola beroda berada dalam kategori Kontes Robot Sepak Bola – Beroda atau biasa disingkat dengan KRSBI-B (Kementerian Riset, 2018). Pada bidang ini yaitu Robot Sepak Bola Beroda yang menerapkan beberapa aspek seperti visualisasi robot (pengindraan robot melalui kamera), komunikasi jaringan setiap robot, *positioning* robot melalui jalur pemetaan. Robot sepak bola berjalan secara otomatis sesuai dengan segala peraturan yang telah ditetapkan dengan cara meniru sistem kerja manusia saat bermain sepak bola yang sesungguhnya. Salah satu bagian terpenting dalam sistem kendali robot sepak bola beroda adalah pemosisian robot, yang digunakan untuk mengatur formasi pertandingan dan menentukan pemetaan. Pemosisian pada robot meliputi 2 hal penting yaitu lokalisasi robot dan kontrol kecepatan posisi robot, dalam melakukan kontrol kecepatan robot ada beberapa aspek yang perlu dipertimbangkan seperti aktuator, sensor, metode yang dipilih dan kondisi saat robot sedang aktif.

Kontrol kecepatan posisi robot sangat diperlukan untuk model robot otomatis seperti robot sepakbola beroda karena sebagian besar gerakannya dilakukan secara otomatis seperti robot berjalan menuju titik tengah lapangan untuk mengambil bola, robot dalam kondisi menyerang dan bertahan, robot melakukan pergerakan tanpa bola, robot melakukan perebutan bola. Menurut gerakan yang akan dilakukan oleh robot tersebut perlu menggunakan sistem yang stabil yang dapat meningkatkan efektifitas kinerja aktuator dan sumber daya robot serta menjaga kondisi robot agar tidak mengalami kerusakan fisik pada saat sedang berhadapan dengan robot lain

dilapangan seperti pada saat robot sedang melakukan perebutan bola dengan robot lain.

Penelitian yang digunakan untuk mengendalikan pergerakan *mobile robot* yang diinginkan yaitu robot dapat dengan cepat merespon posisi keberadaan bola dan juga dapat dengan cepat bergerak menuju bola tersebut, maka dari itu pada kendali PID ini menggunakan nilai $K_p = 2.8$, $K_i = 5$ dan $K_d = 0.8$. Karena dengan menggunakan nilai K_p , K_i , K_d tersebut didapatkan respon pergerakan robot yang cepat dan nilai *overshoot* tidak terlalu tinggi, tetapi nilai *steady state* yang lama (PRASTYAWAN, 2017) Penelitian yang dilakukan menggunakan nilai PID dengan nilai $K_p = 2.8$, $K_i = 5$ dan $K_d = 0.8$ sebagai nilai terbaik yang dapat dengan cepat merespon dan bergerak menuju bola dengan jarak robot dan bola selalu tetap. Pertama pada pembahasan tersebut masih memiliki beberapa kekurangan seperti penggunaan kamera sebagai media untuk pengukuran jarak namun hasil yang diperoleh adalah jarak pixel kamera bukan jarak sebenarnya yang bisa diukur. Kedua kordinat pengujian yang digunakan adalah satu dimensi yaitu kordinat x sedangkan robot sepak bola menggunakan lapangan pertandingan yang memiliki dua dimensi yaitu kordinat x dan y. Ketiga tidak ada informasi mengenai penggunaan nilai PID terbaik untuk jarak dan sudut yang berbeda pada robot karena nilai yang digunakan pada penelitian tersebut adalah satu nilai terbaik dalam pengujian.

Penulis berinovasi untuk membuat sebuah prototipe robot sepak bola beroda menggunakan metode PID menggunakan umpan balik *external rotary encoder* untuk mendapat nilai PID terbaik untuk masing-masing nilai dari jarak dan arah prototipe robot karena robot akan melakukan pergerakan lurus ke arah maju,

bergerak lurus diagonal kanan, bergerak lurus ke arah kanan sehingga mendapatkan pergerakan yang stabil dengan respon yang baik serta kemampuan robot dalam berakselerasi tinggi maka perlu dilakukan pemilihan nilai PID yang tepat untuk masing-masing jarak dan arah pergerakan pada robot.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah diuraikan diatas, diperoleh rumusan permasalahan sebagai berikut :

1. Bagaimana merancang dan mengimplementasikan *hardware* untuk kontrol kecepatan roda pada robot *Three Omni-directional* ?
2. Bagaimana merancang dan mengimplementasikan *software* untuk kontrol kecepatan roda menggunakan metode PID ?
3. Bagaimana menentukan nilai PID untuk jarak dan sudut gerakan yang ditentukan oleh prototipe robot *Three Omni-directional* ?

1.3 Batasan Masalah

Dalam sistem ini, agar tidak menyimpang dari tujuan yang nantinya akan dicapai maka pembahasan masalah dibatasi pada hal – hal sebagai berikut:

1. Prototipe robot menggunakan ukuran maksimal 50cm x 50cm.
2. Berat Prototipe robot kurang dari 25 kg.
3. Menggunakan sensor *external rotary encoder* sebagai umpan balik PID.
4. Pergerakan Prototipe robot hanya dilakukan pada bidang datar dengan jarak 4 meter.
5. Prototipe robot memiliki dua kondisi yaitu keadaan siaga dan keadaan siap.

6. Dalam keadaan siaga menandakan prototipe robot menunggu perintah masukan nilai PID dan Prototipe robot tidak bisa *running*.
7. Dalam keadaan siap maka robot siap *running* dan dalam sekali proses *running* hanya bisa dilakukan sekali.

1.4 Tujuan

Tujuan dari tugas akhir ini adalah sebagai berikut:

1. Dapat merancang *hardware* yang mampu mengontrol kecepatan roda pada prototipe robot sepak bola beroda.
2. Dapat merancang *software* yang mampu menyimulasikan kontrol kecepatan roda menggunakan metode PID.
3. Mendapatkan nilai PID untuk masing-masing jarak dan gerakan sudut pada prototipe robot pada area yang sudah ditentukan.

1.5 Sistematika Penulisan

Untuk memudahkan pembaca dalam memahami persoalan dan pembahasannya, maka penulisan laporan tugas akhir ini dibuat dengan sistematika sebagai berikut.

BAB I PENDAHULUAN

Pada bab ini membahas tentang latar belakang masalah dan penjelasan permasalahan secara umum, perumusan masalah serta batasan masalah yang dibuat, tujuan dari pembuatan tugas akhir dan sistematika penulisan buku ini.

BAB II LANDASAN TEORI

Pada bab ini membahas teori – teori yang berhubungan dan mendukung dalam pembuatan tugas akhir seperti mobile robot,

arduino, kinematika, PID dan literatur yang menunjang dalam pembuatan Tugas Akhir ini.

BAB III ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini membahas tentang perancangan sistem baik pada bagian perangkat keras, maupun perangkat lunak pada penerapan sistem PID ini.

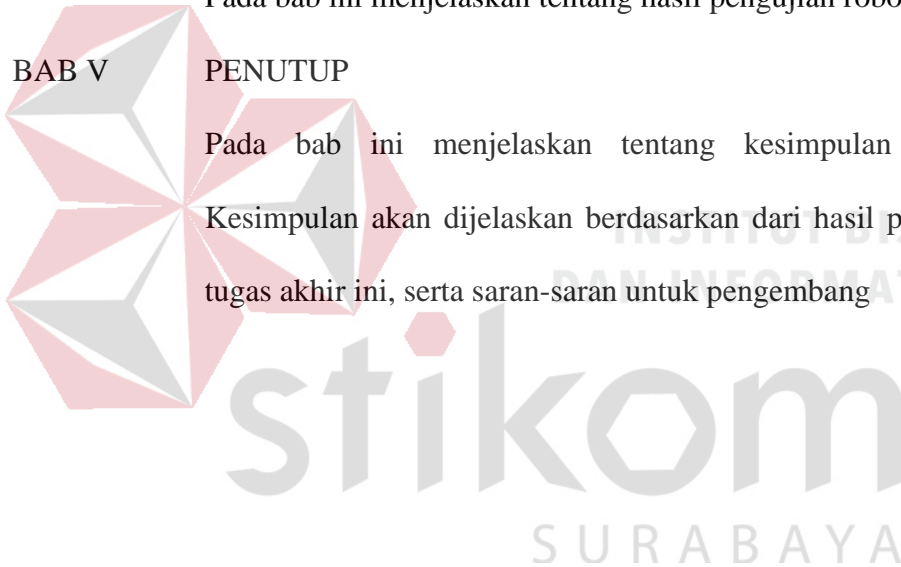
BAB IV PENGUJIAN

Pada bab ini menjelaskan tentang hasil pengujian robot

BAB V PENUTUP

Pada bab ini menjelaskan tentang kesimpulan dan saran.

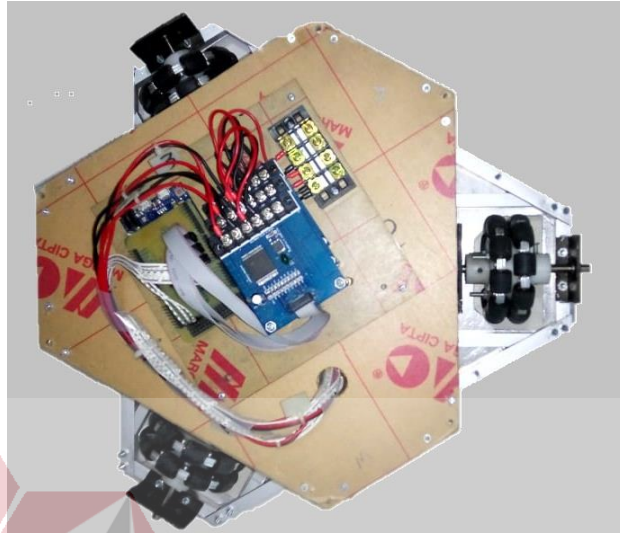
Kesimpulan akan dijelaskan berdasarkan dari hasil pengujian alat tugas akhir ini, serta saran-saran untuk pengembang



BAB II

LANDASAN TEORI

2.1 Mobile Robot

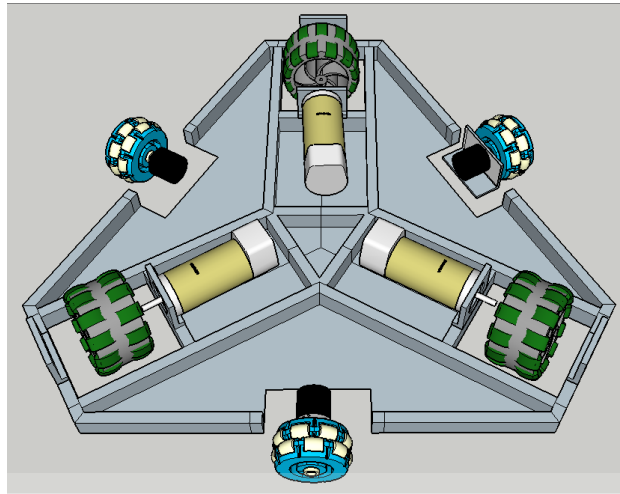


Gambar 2. 1 Mobile Robot

Mobile robot adalah robot yang memiliki ciri khas mempunyai roda sebagai aktuatornya, yang memungkinkan robot untuk melakukan pergerakan. Pada tugas akhir ini robot yang digunakan adalah robot yang telah dibuat sebagai prototipe mobile robot yaitu robot sepak bola beroda dengan spesifikasi berikut:

1. Suplai tegangan sebesar 12-24V untuk motor penggerak
2. Motor penggerak sebanyak 3 buah dengan tipe Planetary Gear 36
3. Rotary encoder internal pada motor dengan 7 pulse/rotation
4. Roda omni dengan diameter 100mm
5. Motor driver EMS 30
6. Controller menggunakan Arduino Due
7. Menggunakan 3 External rotary encoder
8. Dimensi robot 50x50 cm
9. Berat 28kg

2.2 Robot *Holonomic*



Gambar 2. 2 Robot Holonomic

Robot holonomik adalah robot yang dapat bergerak ke segala arah tanpa melakukan rotasi sehingga robot memiliki ruang gerak bebas dalam melakukan pergerakan. Robot memungkinkan untuk melakukan pergerakan dengan banyak kombinasi seperti bergerak maju/mundur, geser kanan/kiri dan rotasi secara bersamaan, hal ini dikarenakan dengan adanya dukungan roda yang digunakan pada robot ini, roda yang digunakan adalah roda omni-directional, roda khusus ini mempunyai roda kecil tambahan yang berporos tegak lurus pada roda inti, sehingga roda dapat bergerak segala arah. Sehingga robot dengan metode holonomic ini dapat melakukan pergerakan/mobilisasi robot lebih cepat, karena tanpa bermanuver saat belok. Omni-directional robot adalah robot dengan sistem pergerakan yang secara langsung dapat bergerak kesegala arah dengan konfigurasi apapun

2.3 Roda Omni-Directional

Roda Omnidirectional telah menjadi populer untuk digunakan pada mobile robot, karena roda ini memungkinkan robot untuk berjalan dengan arah yang sama tanpa harus memutar terlebih dahulu jika robot bergerak maju maupun bergeser. Selain itu, gerakan translasi sepanjang jalur yang diinginkan bisa dikombinasikan dengan rotasi, sehingga robot itu tiba tujuan pada sudut yang benar.



Gambar 2. 3 Roda Omni-Directional

Roda Omnidirectional sebagian besar didasarkan prinsip yang sama, sementara roda menyediakan traksi dalam arah Seperti biasa pada sumbu motor, roda dapat meluncur tanpa geseran arah sumbu motor. Untuk mencapai ini, roda dibangun menggunakan roda yang lebih kecil yang terpasang di sepanjang pinggiran roda utama menggunakan rol dengan arah rotasi yang tidak paralel atau tegak lurus sumbu motor. Gambar 2.3 menunjukkan contoh jenis roda yang digunakan pada Tugas Akhir ini yaitu roda omni yang menggunakan diameter 100ml.

2.4 Motor DC

Motor dc yang digunakan adalah Planetary Gear 36 dapat dilihat pada gambar F.3, dengan spesifikasi torsi 13kgfcm, kecepatan 870Rpm, memiliki rotary internal dengan pulse 7ppr, dan di suplai dengan tegangan 12-24Vdc



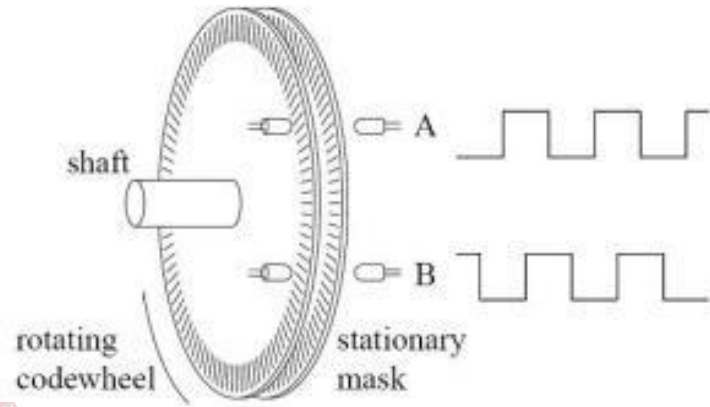
Gambar 2. 4 Motor DC

Rotary internal yang sudah termasuk pada motor sebenarnya tergolong rotary dengan pulse rendah namun rotary tersebut terpasang langsung pada putaran inti dinamo dan dengan adanya gearbox untuk output putaran sehingga rotary dapat lebih detil dalam melakukan pembacaan putaran.

2.5 External Rotary Encoder

Rotary Encoder (ES403-100-3-T-24) merupakan suatu perangkat sensor yang biasa digunakan untuk menghitung putaran poros, sehingga biasanya banyak digunakan untuk mendeteksi pergerakan/perpindahan posisi. terdiri dari *double track* atau *single track* dan dua sensor yang disebut *channel A* dan *B* (Gambar 2.1). Ketika poros berputar, deretan pulsa akan muncul di masing-masing *channel* pada frekuensi yang proporsional dengan kecepatan putar sedangkan hubungan fasa antara *channel A* dan *B* menghasilkan arah putaran. Dengan menghitung jumlah pulsa yang terjadi terhadap resolusi piringan maka putaran dapat diukur. Untuk

mengetahui arah putaran, dengan mengetahui *channel* mana yang *leading* terhadap *channel* satunya dapat kita tentukan arah putaran yang terjadi karena kedua *channel* tersebut akan selalu berbeda fasa seperempat putaran



Gambar 2. 5 External Rotary Encoder

Pulsa yang telah dihasilkan *rotary encoder* akan dikonversi menjadi sudut dengan menggunakan persamaan dibawah :

$$K_r = 2 \times \pi \times r \quad (2.1)$$

$$K_{enc} = \left(\frac{pulsa}{tickRev} \right) \times K_r \quad (2.2)$$

Keterangan :

R : Jari – jari roda

K_r : Keliling roda

K_{enc} : Keliling roda terbaca *rotary encoder*

2.5.1 Datasheet ES403-100-3-T-24



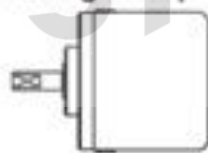
Gambar 2. 6 Datasheet ES403-100-3-T-24

E30S	4	3000	3	N	24	
Series	Shaft diameter	Pulse/1Revolution	Output phase	Output	Power supply	Cable
Diameter ø30mm, shaft type	ø4mm	Refer to resolution	3 : A, B, Z 6 : A, \bar{A} , B, \bar{B} , Z, \bar{Z}	T: Totem pole output N: NPN open collector output V: Voltage output L: Line driver output(※)	5 : 5VDC \pm 5% 24: 12-24VDC \pm 5%	No mark: Cable type C: Connector cable type(※)

Gambar 2. 7 Keterangan ES403-100-3-T-24

◎ Cable type

- Totem pole output / NPN open collector output / Voltage output



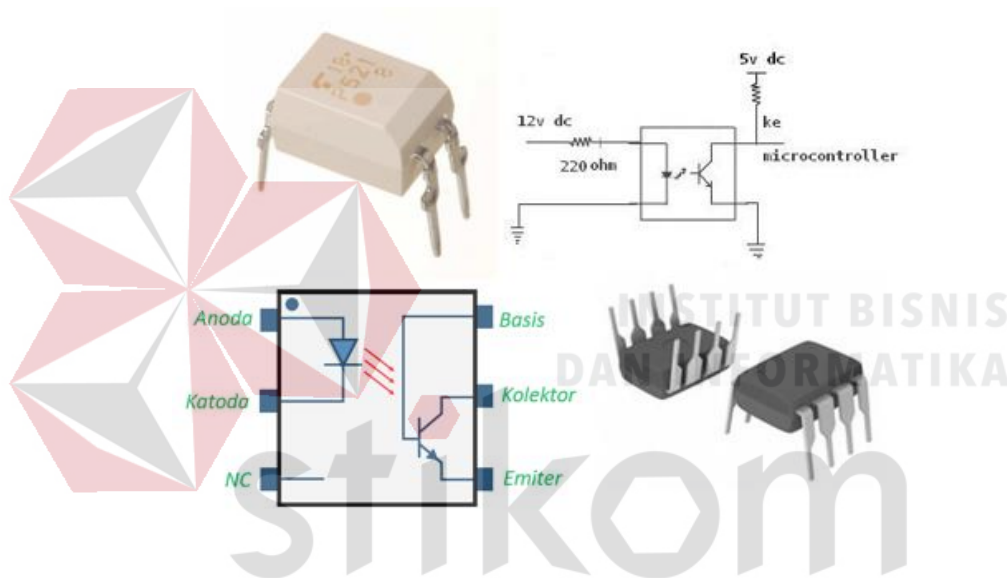
Black: OUT A
White: OUT B
Orange: OUT Z
Brown: +V (5VDC, 12-25VDC \pm 5%)
Blue: GND (0V)
Shield: F.G.

Gambar 2. 8 Tipe kabel ES403-100-3-T-24

2.6 Optocoupler

Optocoupler adalah komponen yang digunakan untuk menghubungkan sebuah jalur rangkaian elektronika berdasarkan cahaya optic, tidak langsung terhubung dengan cahaya optik melainkan terhubung melalui cahaya inframerah yang dikirim oleh transmitter ke receiver. Optocoupler berfungsi untuk memedakan

jalur rangkaian tegangan 5v atau 12/24v seperti dalam penggunaan external rotary encoder yang terhubung dengan tegangan 24v sedangkan mikrokontroler Arduino terhubung dengan tegangan 5v sehingga untuk menghindari sinyal feedback besar yang dihasilkan oleh sensor rotary encoder jika terhubung secara langsung dengan mikrokontroler Arduino maka perlu dilakukan cara menambahkan sebuah rangkaian optocoupler switch sebagai media penghubung kedua komponen tersebut tanpa menghasilkan feedback tegangan dari sensor rotary encoder.



Gambar 2. 9 Optocoupler

Sistem kerja dari optocoupler tipe TLP521 dengan 4 kaki yang terdiri dari dua bagian yaitu transmitter dan receiver. Pada bagian transmitter terdapat 2 kaki yaitu kaki positif dan negative dari komponen LED inframerah, untuk bagian receiver terdiri dari 2 kaki positif dan negatif dari komponen phototransistor. Kedua komponen ini sangat padu karena LED infrared menghasilkan panas dan cahaya sedangkan phototransistor sangat peka terhadap cahaya inframerah dan panas.

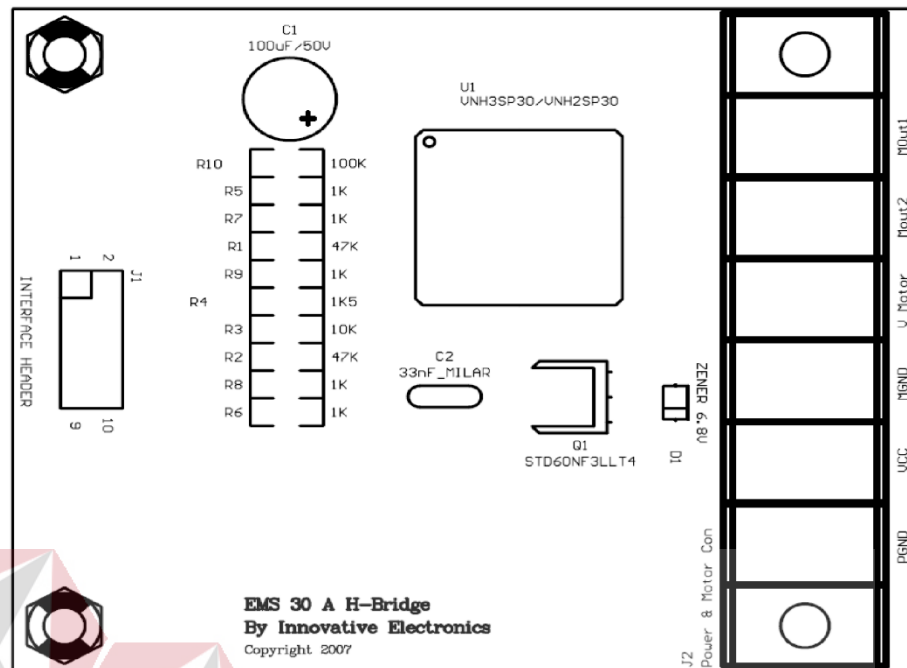
2.7 Motor Driver EMS 30A

Embedded Module Series (EMS) 30 A H-Bridge merupakan *driver* H-Bridge yang didisain untuk menghasilkan *drive* 2 arah dengan arus kontinyu sampai dengan 30A pada tegangan 5,5 Volt sampai 16 Volt. Modul ini dilengkapi dengan rangkaian sensor arus beban yang dapat digunakan sebagai umpan balik ke pengendali. Modul ini mampu men-*drive* beban-beban induktif seperti misalnya relay, solenoida, motor DC, motor stepper, dan berbagai macam beban lainnya.

Spesifikasi:

- Terdiri dari 1 *driver full* H-Bridge beserta rangkaian *current sense*.
- Mampu melewatkan arus kontinyu 30 A.
- *Range* tegangan output untuk beban: 5,5 V sampai 16 V.
- Input kompatibel dengan level tegangan TTL dan CMOS.
- Jalur catu daya input (VCC) terpisah dari jalur catu daya untuk beban (V Mot).
- Output tri-state.
- Dilengkapi dengan dioda eksternal untuk pengaman beban induktif.
- Frekuensi PWM sampai dengan 20 KHz
- *Fault Detection*.
- Proteksi hubungan singkat.
- Proteksi *overtemperature*.
- *Undervoltage* dan *Overvoltage Shutdown*.
- *Reverse Battery Protection*

Tata Letak Komponen:



Gambar 2. 10 Motor Driver EMS 30A

Modul H-Bridge memiliki 1 set *header* (J1) dan 1 set terminal konektor (J2). Pada bagian ini akan dijelaskan deskripsi dan fungsi dari masing-masing *header* dan konektor tersebut. **Interface Header** (J1) berfungsi sebagai input untuk antarmuka dengan input-output digital serta output analog dari modul H-Bridge. Seperti pada tabel 2.1 berikut :

Tabel 2. 1 Pin Interface Header

No. pin	Nama	I / O	Fungsi
1	MIN1	I	Pin input untuk menentukan output MOUT1
2	MIN2	I	Pin input untuk menentukan output MOUT2

3	MEN1	I/O	<p>Pin <i>enable</i> untuk output MOUT1 Diberi logika High untuk mengaktifkan <i>half</i> HBridge 1, diberi logika Low secara eksternal untuk menonaktifkan <i>half</i> H-Bridge 1 Jika terjadi kondisi <i>Fault</i> (<i>thermal shutdown</i>, <i>undervoltage</i>, <i>overvoltage</i>, dsb.), maka pin ini akan ditarik Low secara internal oleh modul HBridge untuk melaporkan adanya kondisi <i>Fault</i></p>
4	MEN2	I/O	<p>Pin <i>enable</i> untuk output MOUT2 Diberi logika High untuk mengaktifkan <i>half</i> HBridge 2, diberi logika Low secara eksternal untuk menonaktifkan <i>half</i> H-Bridge 2 Jika terjadi kondisi <i>Fault</i> (<i>thermal shutdown</i>, <i>undervoltage</i>, <i>overvoltage</i>, dsb.), maka pin ini akan ditarik Low secara internal oleh modul HBridge untuk melaporkan adanya kondisi <i>Fault</i></p>
5	MCS	O	<p>Output tegangan analog yang berbanding lurus dengan arus beban (<i>Range</i> output 0 – 5 Volt)</p>

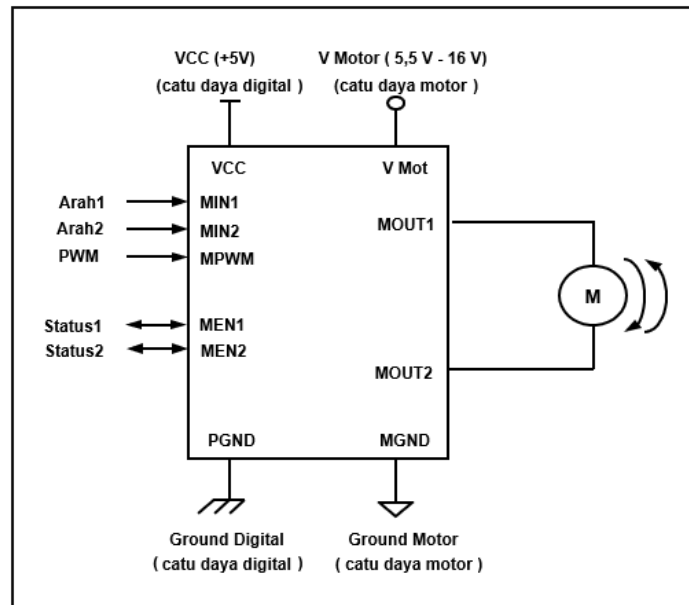
6	MPWM	I	Pin input untuk mengatur kerja modul H-Bridge secara PWM
7,9	VCC	-	Terhubung ke catu daya untuk input (5 Volt)
8,10	PGND	-	Titik referensi untuk catu daya input

Power & Motor Con (J2) berfungsi sebagai konektor untuk catu daya dan beban.

Tabel 2. 2 Pin Power & Motor Conector

Nama	Fungsi
PGND	Titik referensi untuk catu daya input
VCC	Terhubung ke catu daya untuk input (5 Volt)
MGND	Titik referensi untuk catu daya output ke beban
V MOTOR	Terhubung ke catu daya untuk output ke beban
MOUT2	Output ke beban dari <i>half</i> H-Bridge kedua
MOUT1	Output ke beban dari <i>half</i> H-Bridge pertama

Sebuah modul H-Bridge 30A dapat digunakan untuk mengatur kerja 1 buah motor DC secara 2 arah. Contoh koneksinya dapat dilihat pada gambar berikut ini:



Gambar 2. 11 Pin Motor Driver EMS 30A

2.8 LCD (Liquid Crystal Display)

LCD adalah sebuah jenis media tampilan yang menggunakan kristal cair sebagai penampil utama, selain itu LCD juga dapat digunakan untuk menampilkan karakter atau gambar. Dalam modul LCD terdapat mikrokontroler yang berfungsi sebagai pengendali tampilan karakter LCD. Mikrokontroler pada sebuah LCD dilengkapi dengan memori dan *register*, memori yang digunakan, berikut ini merupakan konfigurasi *pin* LCD 16x2.

Tabel 2. 3 Konfigurasi pin LCD 16x2

<i>PIN</i>	Nama	Fungsi
1	V_{SS}	<i>Ground</i>
2	V_{CC}	+5V
3	V_{EE}	<i>Contrast Voltage</i>
4	RS	<i>Register Select,</i> <i>0 = Instruction Register, 1 = Data</i>

<i>PIN</i>	<i>Nama</i>	<i>Fungsi</i>
		<i>Register</i>
5	R/W	<i>Read/Write, to choose read or write mode, 0 = Write Mode, 1 = Read Mode</i>
6	E	<i>Enable, 0 = Start do latch data do LCD Character, 1 = Disable</i>
7	DB0	<i>Data Bus (LSB)</i>
8	DB1	<i>Data Bus</i>
9	DB2	<i>Data Bus</i>
10	DB3	<i>Data Bus</i>
11	DB4	<i>Data Bus</i>
12	DB5	<i>Data Bus</i>
13	DB6	<i>Data Bus</i>
14	DB7	<i>Data Bus (MSB)</i>
15	BPL (+)	<i>Back Plane Light (+)</i>
16	BPL (-) / GND	<i>Back Plane Light (-) / Ground Voltage</i>



Gambar 2. 12 LCD 16x2 (Liquid Crystal Display)

2.9 Universal Battery Elimination Circuit (UBEC)

Mengubah tegangan, tinggi ke rendah atau sebaliknya, memerlukan rangkaian yang tepat, agar daya dapat di-deliver dengan tingkat efisiensi setinggi mungkin. Menurunkan tegangan dengan menggunakan IC regulator seperti 7805, sangat umum digunakan. Regulator ini memiliki kemampuan menangani arus hingga 1A, dengan V_{in} minimal sama dengan 7V, untuk menghasilkan output 5V. Dengan perhitungan sederhana, bila $V_{in} = 9V$, maka disipasi daya ~ 4 Watt, satu nilai yang cukup besar (panas), atau menggunakan regulator linier tipe LDO, seperti 2940, yang juga memiliki kemampuan menangani arus hingga 1A, dengan V_{in} minimal sama dengan 5.5V, untuk menghasilkan output 5V.

Pilihan lain adalah regulator switching. Untuk kebutuhan mencatu motor servo atau rangkaian lain yang bekerja pada tingkat tegangan 5V – 6V, dapat menggunakan UBEC. UBEC – Universal Battery Elimination Circuit adalah rangkaian elektronik yang mengambil daya dari battery pack atau sumber DC lainnya, dan menurunkannya ke level tegangan 5V atau 6V. Tegangan input maksimum tergantung pada spesifikasi UBEC



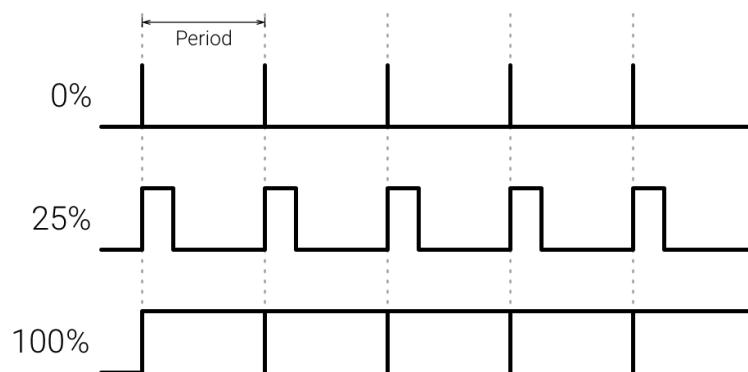
Gambar 2. 13 UBEC – Universal Battery Elimination Circuit

UBEC biasanya digunakan pada aplikasi yang memerlukan arus lebih tinggi, dan divais mampu men-deliver daya dengan efisiensi hingga 92%. Ketika

memilih UBEC, pastikan model UBEC yang dipilih memiliki rating arus yang sesuai dengan kebutuhan (beban). Rangkaian lain yang juga sering dibutuhkan adalah DC-DC Booster. Sebagai contoh, satu produk DC-DC, mampu menghasilkan output 3.7V – 34V dengan input 3.7V – 34V. Artinya, dengan tegangan input minimum 3.7V dapat dihasilkan output maksimum 34V, dengan arus input maksimum 3A, serta mampu men-deliver daya dengan tingkat efisiensi hingga 90%.

2.10 Pulse Width Modulation (PWM)

PWM adalah salah satu teknik modulasi dengan mengubah lebar pulsa (*duty cycle*) dengan nilai amplitudo dan frekuensi yang tetap. Satu siklus pulsa merupakan kondisi high kemudian berada di zona transisi ke kondisi low. Lebar pulsa PWM berbanding lurus dengan amplitudo sinyal asli yang belum termodulasi. *Duty Cycle* merupakan representasi dari kondisi logika high dalam suatu periode sinyal dan dinyatakan dalam bentuk (%) dengan *range* 0% sampai 100%, sebagai contoh jika sinyal berada dalam kondisi high terus menerus artinya memiliki *duty cycle* sebesar 100%. Jika waktu sinyal keadaan high sama dengan keadaan low maka sinyal mempunyai *duty cycle* sebesar 50%.



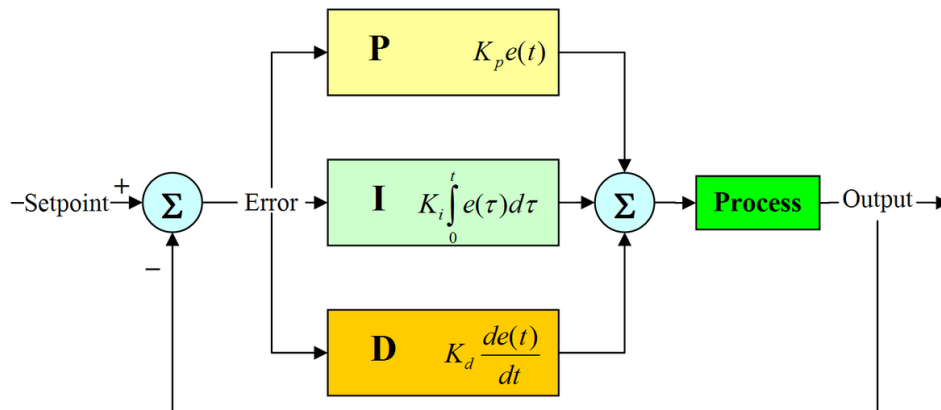
Gambar 2. 14 PWM (Pulse Width Modulation)

Aplikasi penggunaan PWM biasanya ditemui untuk pengaturan kecepatan motor dc, pengaturan cerah/redup LED, dan pengendalian sudut pada motor servo. Contoh penggunaan PWM pada pengaturan kecepatan motor dc semakin besar nilai *duty cycle* yang diberikan maka akan berpengaruh terhadap cepatnya putaran motor. Apabila nilai *duty cycle*-nya kecil maka motor akan bergerak lambat (firman, 2017).

Untuk membandingkannya terhadap tegangan DC, PWM memiliki 3 mode operasi yaitu:

1. *Inverted Mode*. Pada mode *inverted* ini jika nilai sinyal lebih besar dari pada titik pembanding (compare level) maka output akan di set high (5v) dan sebaliknya jika nilai sinyal lebih kecil maka output akan di set low (0v) seperti pada gelombang A pada gambar di atas.
2. *Non Inverted Mode*. Pada mode *non inverted* ini output akan bernilai high (5v) jika titik pembanding (compare level) lebih besar dari pada nilai sinyal dan sebaliknya jika bernilai low (0v) pada saat titik pembanding lebih kecil dari nilai sinyal seperti pada gelombang B pada gambar di atas.
3. *Toggle Mode*. Pada mode *toggle* output akan beralih dari nilai high (5v) ke nilai low (0v) jika titik pembanding sesuai dan sebaliknya beralih dari nilai low ke high

2.11 PID (Proportional Integral Derivatif)



Gambar 2. 15 Blok Diagram PID

Gambar di atas merupakan sistem kontrol PID yaitu kontroler untuk menentukan tingkat kepresisian suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut (*Feed Back*). Komponen kontrol PID ini terdiri dari tiga jenis yaitu *Proportional*, *Integral* dan *Derivative*, Ketiganya dapat bekerja bersamaan maupun sendiri-sendiri tergantung dari respon suatu alat. Dalam perancangan sistem kontrol PID yang perlu dilakukan adalah mengatur parameter P, I atau D agar tanggapan sinyal keluaran sistem terhadap masukan tertentu sebagaimana yang diinginkan.

2.11.1 Proporsional

Pengontrol proporsional memiliki keluaran yang sebanding (proporsional) dengan besarnya sinyal kesalahan. Secara sederhana dapat dinyatakan bahwa keluaran pengontrol P merupakan perkalian antara konstanta proporsional dengan masukannya. Perubahan pada sinyal sebesar konstanta pengalinya. Rumus dasar dari kontroler proporsional adalah :

$$P = K_p e(t) \quad (2.3)$$

dengan keterangan sebagai berikut :

P adalah Proporsional

Kp adalah nilai konstanta proposional

e(t) adalah nilai eror yang selalu akan berubah

Ciri-ciri pengontrol proporsional harus diperhatikan ketika pengontrol tersebut diterapkan pada suatu sistem. Secara eksperimen, pengguna pengontrol proporsional harus memperhatikan ketentuan berikut :

- Apabila nilai Kp kecil, pengontrol proporsional hanya mampu melakukan koreksi kesalahan yang kecil, sehingga akan menghasilkan respon sistem yang lambat.
- Apabila nilai Kp dinaikan, respon sistem menunjukkan semakin cepat mencapai set point dan keadaan stabil.

Namun jika nilai Kp diperbesar sehingga mencapai harga yang berlebihan, akan menyebabkan sistem bekerja tidak stabil. Pengontrol ini berfungsi menghasilkan respon sistem yang memiliki kesalahan keadaan stabil nol. Jika suatu plant tidak memiliki unsur integrator, pengontrol proporsional tidak akan mampu menjamin keluaran sistem dengan kesalahan keadaan stabilnya nol. Dengan pengontrol integral, respon sistem dapat diperbaiki yaitu mempunyai kesalahan keadaan stabilnya nol. Rumus dasar dari kontroler Integral adalah

$$I = K_i \int_0^t e(\tau) d\tau \quad (2.4)$$

dengan keterangan sebagai berikut :

- I adalah Integral
- K_i adalah nilai konstanta integral
- $e(t)$ adalah nilai eror
- dt adalah nilai perubahan waktu (second)

2.11.2 Integral

Kontrol integral tidak dapat digunakan sendiri, maka itu harus digabungkan dengan kontroler proporsional. Adapun karakteristik Pengontrol I yaitu sebagai berikut

- Keluaran pengontrol membutuhkan selang waktu tertentu, sehingga pengontrol I cenderung memperlambat respon.
- Saat sinyal kesalahan berharga nol, keluaran pengontrol akan bertahan pada nilai sebelumnya.
- Jika sinyal kesalahan tidak berharga nol, keluaran akan menunjukkan kenaikan atau penurunan yang dipengaruhi oleh besarnya sinyal kesalahan dan nilai K_i .
- Konstanta Integral K_i yang berharga besar akan mempercepat hilangnya offset, namun semakin besar nilai konstanta K_i akan mengakibatkan peningkatan osikasi dari sinyal keluaran pengontrol.

2.11.3 Derivative

Keluaran pengontrol D memiliki sifat seperti halnya suatu operasi diferensial. Perubahan yang mendadak pada masukan pengontrol akan mengakibatkan perubahan sangat besar dan cepat. Saat masukannya tidak

mengalami perubahan, keluaran pengontrol juga tidak akan mengalami perubahan, sedangkan apabila sinyal masukan berubah mendadak dan menaik, sinyal keluaran menghasilkan sinyal berbentuk impuls. Jika sinyal masukan berubah naik secara perlahan, keluaran justru merupakan fungsi step yang besar magnitudnya sangat dipengaruhi oleh kecepatan naik dari fungsi ramp dan faktor konstanta diferensialnya. Rumus dasar dari kontroler derivatif adalah:

$$D = K_d \frac{de(t)}{dt} \quad (2.5)$$

dengan keterangan sebagai berikut :

D adalah Derivatif

Kd adalah nilai konstanta Derivatif

de(t) adalah nilai perubahan error

dt adalah nilai perubahan waktu (second)

Adapun karakteristik pengontrol D yaitu sebagai berikut:

- Pengontrol ini tidak dapat menghasilkan keluaran bila tidak ada perubahan pada masukannya.
- Jika sinyal kesalahan berubah terhadap waktu, maka keluaran yang dihasilkan pengontrol tergantung pada nilai Td dan laju perubahan sinyal kesalahan.
- Pengontrol derivative mempunyai suatu karakter untuk mendahului, sehingga pengontrol ini dapat menghasilkan koreksi yang signifikan sebelum pembangkit kesalahan menjadi sangat besar.
- Meningkatkan stabilitas sistem

2.12 Arduino IDE

Arduino IDE (*Integrated Development Environment*) adalah software yang di gunakan untuk memprogram di arduino, dengan kata lain Arduino IDE sebagai media untuk memprogram *board* arduino. Arduino IDE ini berguna sebagai text editor untuk membuat, mengedit, dan juga mevalidasi kode program. bisa juga digunakan untuk meng-upload ke board Arduino. Kode program yang digunakan pada Arduino disebut dengan istilah Arduino “sketch” atau disebut juga source code arduino, dengan ekstensi file source code .ino



Gambar 2. 16 Arduino IDE

Editor Programming pada umumnya memiliki fitur untuk cut / paste dan untuk find / replace teks, demikian juga pada Arduino IDE. Pada bagian **keterangan aplikasi** memberikan pesan balik saat menyimpan dan mengeksport serta sebagai tempat menampilkan kesalahan. **Konsol log** menampilkan teks log dari aktifitas Arduino IDE, termasuk pesan kesalahan yang lengkap dan informasi lainnya. Pojok kanan bawah menampilkan port serial yang di gunakan. Tombol toolbar terdapat

ikon tombol pintas untuk memverifikasi dan meng-upload program, membuat, membuka, dan menyimpan sketch, dan membuka monitor serial



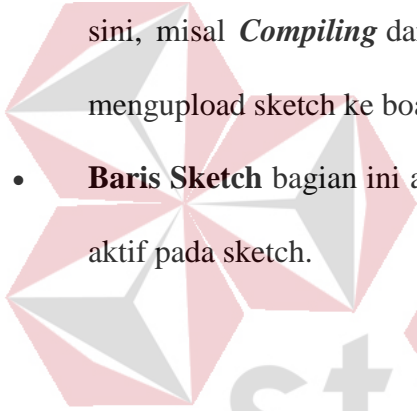
Gambar 2. 17 Interface Arduino IDE

Verify pada versi sebelumnya dikenal dengan istilah Compile. Sebelum aplikasi di-upload ke board Arduino, biasanya untuk memverifikasi terlebih dahulu sketch yang dibuat. Jika ada kesalahan pada sketch, nanti akan muncul error. Proses Verify / Compile mengubah sketch ke binary code untuk di-upload ke mikrokontroller.

Upload tombol ini berfungsi untuk mengupload sketch ke board Arduino. Walaupun kita tidak mengklik tombol verify, maka sketch akan di-compile, kemudian langsung diupload ke board. Berbeda dengan tombol verify yang hanya

berfungsi untuk memverifikasi source code saja.

- **Open Sketch** Membuka sketch yang sudah pernah dibuat. Sketch yang dibuat dengan IDE Arduino akan disimpan dengan ekstensi file .ino
- **New Sketch** Membuka window dan membuat sketch baru.
- **Save Sketch** menyimpan sketch, tapi tidak disertai dengan mengcompile.
- **Serial Monitor** Membuka interface untuk komunikasi serial, nanti akan kita diskusikan lebih lanjut pada bagian selanjutnya.
- **Keterangan Aplikasi** pesan-pesan yang dilakukan aplikasi akan muncul di sini, misal *Compiling* dan *Done Uploading* ketika kita mengcompile dan mengupload sketch ke board Arduino
- **Baris Sketch** bagian ini akan menunjukkan posisi baris cursor yang sedang aktif pada sketch.



INSTITUT BISNIS
DAN INFORMATIKA

stikom
SURABAYA

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini secara umum akan dijelaskan proses pembuatan proyek tugas akhir yang terdiri dari 3 bagian pokok yang meliputi perancangan mekanik, perancangan elektronik, perancangan perangkat lunak. Pada pembahasan bab ini difokuskan pada hal-hal pokok yang terdapat dalam proses pembuatan prototipe robot yaitu perancangan sistem, perancangan mekanik, pembuatan perangkat keras dan pembuatan algoritma yang akan digunakan dalam pembahasan proyek tugas akhir.

3.1 Perancangan sistem

Pada tugas akhir ini akan dirancang sebuah prototipe mobile robot yang menggunakan *frame base* berbentuk segitiga yang menggunakan roda berjenis *omni directional* sehingga mampu bergerak kesegala arah tanpa melakukan rotasi, prototipe mobile robot ini akan bergerak lurus ke arah depan, ke arah diagonal kanan dan lurus ke arah kanan berdasarkan fungsi odometri yang diberikan masukan dari luar sistem. Untuk mendapatkan pergerakan yang stabil serta respon sistem yang baik maka fungsi odometri tersebut ditambahkan dengan metode PID sehingga robot mampu mendapatkan respon yang baik dan stabil.

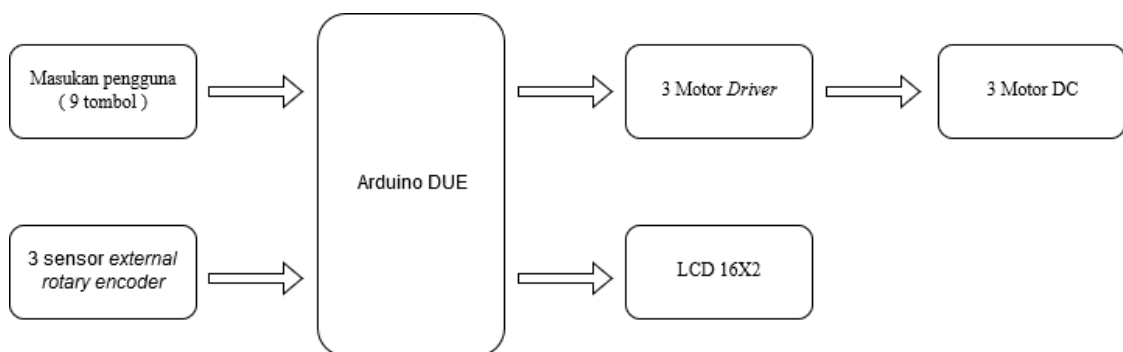
Metode PID yang digunakan dalam Tugas Akhir ini adalah dengan cara konsep trial and error yaitu dengan cara mencoba-coba nilai konstanta pada masing-masing parameter P-I-D untuk mendapat nilai dengan kondisi terbaik yang diinginkan. Penentuan nilai konstanta PID sangat penting dimana untuk menentukan parameter

pengujian yang akan untuk pengaturan kendali proportional, integrative, Derivatif dalam system prototipe mobile robot ini

Pada perancangan prototipe mobile robot ini terdapat 2 komponen penting yaitu 3 sensor external rotary encoder untuk memberikan sinyal umpan balik dan rangkaian optocoupler switch penahan feedback yang dihasilkan dari tegangan 12v. Menggunakan 3 sensor external rotary encoder sebagai pemberi sinyal umpan balik yang berfungsi untuk mendapatkan nilai kesalahan yang akan diproses oleh sistem. metode PID digunakan bertujuan untuk mengolah suatu sinyal kesalahan (error) yang dihasilkan oleh sistem untuk diolah kembali menjadi sinyal masukan pembeding/pengkoreksi karena nilai kesalahan tersebut akan diproses dengan rumus PID untuk dijadikan sinyal kendali yang akan diolah secara terus menerus hingga menjadi nilai output

3.1.1 Perancangan perangkat keras

Perangkat keras dalam prototipe mobile robot ini secara umum terbagi menjadi 2 bagian yaitu bagian masukan dan bagian keluaran. Perancangan Blok Diagram pada rancangan prototipe mobile robot ini dapat dilihat pada gambar 3.1

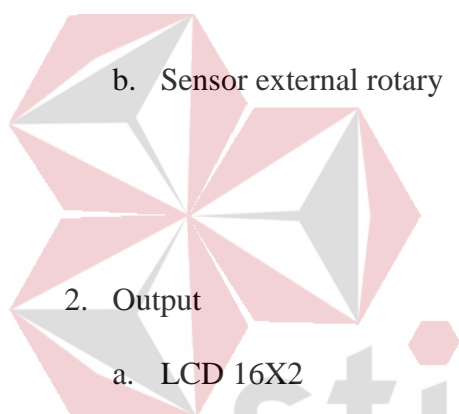


Gambar 3. 1 Blok Diagram perangkat keras

Tiap-tiap bagian dari diagram blok sistem pada gambar diatas dapat dijelaskan sebagai berikut :

1. Input

a. Masukan pengguna : Pada masukan pengguna terdapat 9 tombol/switch yang memiliki fungsi masing masing seperti tombol siaga , tombol konstanta P, I, D, tombol jarak, tombol kondisi lurus kedepan, lurus ke kanan, lurus diagonal kanan.



b. Sensor external rotary : Untuk mengambil nilai dari hasil tiap putaran encoder yang akan dikonversi menjadi jarak.

2. Output

a. LCD 16X2 : Digunakan untuk menampilkan nilai konstanta PID, jarak dan status prototipe robot.

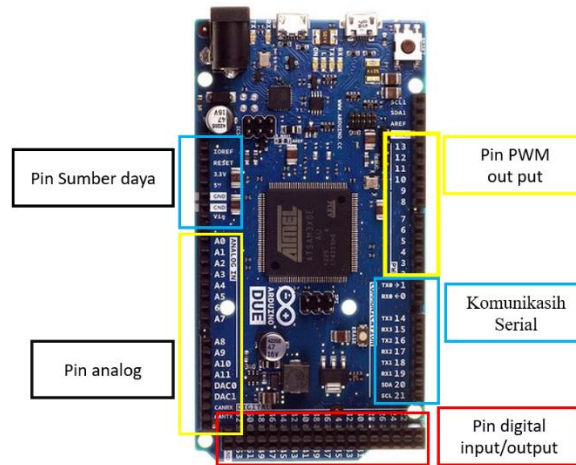
b. Driver dan Motor DC : digunakan sebagai actuator sehingga robot bisa bergerak.

Penjelasan dari blok diagram diatas yaitu tombol/switch digunakan untuk mengatur keadaan robot antara lain switch idle adalah masukan pengguna yang menandakan bahwa robot dalam keadaan siaga yaitu robot tidak bisa bergerak/berjalan namun hanya dalam mode input nilai konstanta PID. Tombol Kp, Ki, Kd digunakan untuk memasukan nilai input berupa konstanta bilangan bulat dari 1-10. Switch lurus maju maka robot berjalan maju, switch lurus kanan maka

robot berjalan kearah kanan, sitch lurus diagonal maka robot bergerak kearah diagonal kanan, Tombol jalan maka robot dalam kondisi running program untuk bergerak/berjalan. 3 sensor rotary encoder digunakan untuk umpan balik sinyal error yang akan diproses didalam PID. Agar pergerakan robot stabil dan tidak mengaami osilasi sehingga Motor DC ditambahkan metode PID untuk memperkecil atau menghilangkan osilasi tersebut. LCD 16x2 digunakan untuk menampilkan jarak, nilai PID dan kondisi yang sedang dikerjakan oleh robot.

3.1.2 Alokasi pin Arduino due

Pada proyek tugas akhir ini pemilihan sistem tertanam untuk dipasang pada prototipe robot sangat penting karena mempengaruhi proses eksekusi algoritma yang akan ditanamkan, system tertanam adalah perangkat elektronika yang berfungsi sebagai kontroler atau sering disebut mikrokontroler karena memiliki bentuk yang kecil, mikrokontroler memiliki banyak jenis dengan tipe yang berbeda-beda seperti mikrokontroler yang mengutamakan bentuk, kecepatan prosesor atau fungsinya. Salah satu mikrokontoler yang memiliki clock speed yang tinggi dan memiliki pin yang banyak adalah Arduino DUE yaitu dengan kecepatan 84MHz dan 54 pinout. Pemiihan mikrokontroler ini berdasarkan kebutuhan pin yang banyak dan untuk memenuhi 100 pulse per rotation yang dihasilkan oleh sensor external rotary encoder.



Gambar 3. 2 Pin Arduino DUE

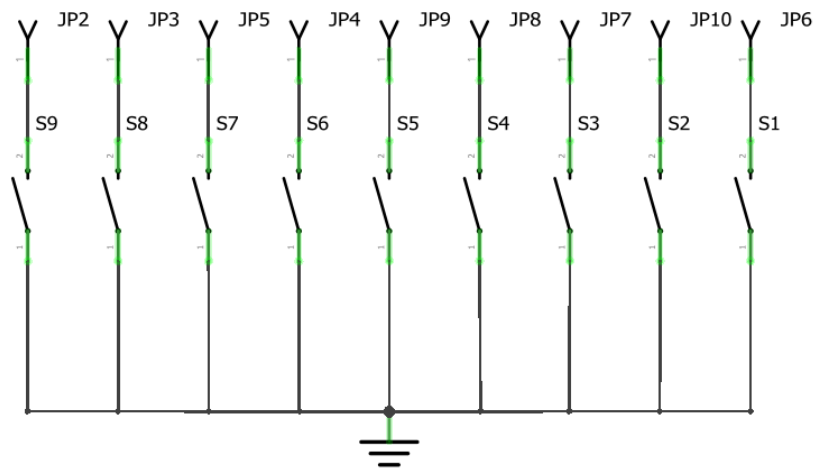
Berikut adalah pembagian pin Arduino due beserta alokasi penggunaan yang digunakan dalam pembuatan proyek tugas akhir ini dapat dilihat paa tabel 3.1.

Tabel 3. 1 alokasi pin Arduino due

PIN ARDUINO DUE	Alokasi penggunaan pin pada hardware
A0	Switch idle (kondisi siaga)
A1	Switch Maju (kondisi robot bergerak lurus kedepan)
A2	Switch Kanan (kondisi robot bergerak lurus kekanan)
A3	Switch Kanan (kondisi robot bergerak lurus diagonal ke kanan)
A4	Tombol masukan konstanta proporsional (Kp)
A5	Tombol masukan konstanta proporsional (Ki)
A6	Tombol masukan konstanta proporsional (Kd)
A7	Tombol jalan (robot bergerak)

A8	Tombol masukan besar jarak
22	Input rangkain optocoupler switch 1
25	Input rangkain optocoupler switch 1
31	Input rangkain optocoupler switch 2
33	Input rangkain optocoupler switch 2
49	Input rangkain optocoupler switch 3
51	Input rangkain optocoupler switch 3
24	Direksi motor 1
22	Direksi motor 1
3	Pwm motor 1
26	Direksi motor 2
28	Direksi motor 2
11	Pwm motor 2
32	Direksi motor 3
30	Direksi motor 3
5	Pwm motor 3

3.1.3 Rangkaian tombol



Gambar 3. 3 Rangkaian Tombol

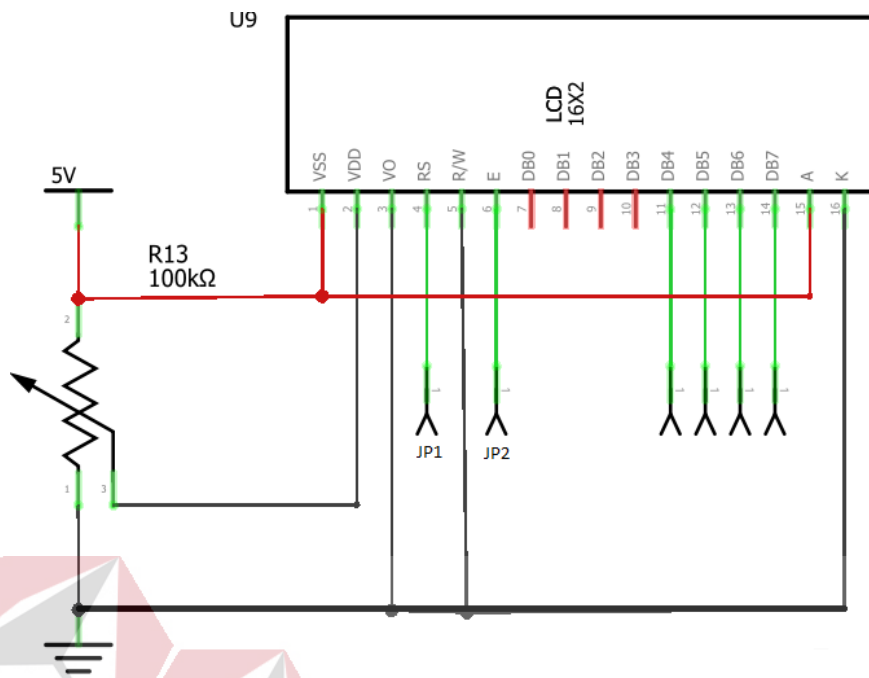
Keterangan:

JP = *Jumper point* (pin penghubung)

S = *Switch* (tombol)

Rangkaian input digunakan untuk memberikan sinyal digital ke mikrokontroler untuk diproses menjadi sebuah perintah berdasarkan algoritma yang dipasangkan untuk tiap-tiap pin input dari rangkain tersebut, pada prototipe mobile robot ini membutuhkan sebuah maskan untuk memerintah robot sehingga memiliki gerakan yang sesuai tanpa harus melakukan download program kembali. Terdapat 9 masukan berupa tombol dan switch yang akan mempengaruhi kerja system prototipe mobile robot ini yaitu tombol input konstanta P I D, input kondisi robot (bergerak maju, diagonal, keser ke kanan) dan tombol jalan (running).

3.1.4 LCD 16x2



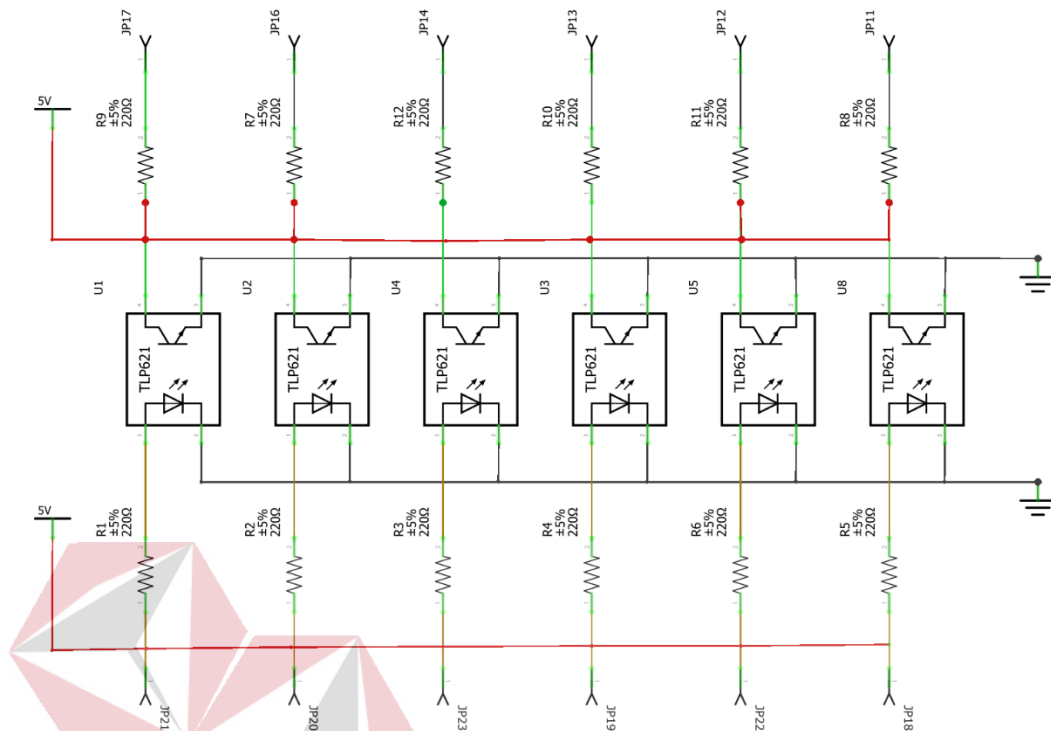
Gambar 3. 4 Rangkaian LCD 16x2

Keterangan:

JP = *Jumper point* (pin penghubung)

Rangkaian lcd digunakan untuk melihat nilai K_p , K_i , K_d dan status pada robot, display perlu digunakan dalam prototipe ini karena memaksimalkan kinerja prototipe robot tanpa perlu menghubungkan PC dengan robot. Untuk menentukan nilai K_p , K_i , K_d sangat penting karena tujuan utama pembuatan proyek tugas akhir ini adalah melakukan percobaan untuk mendapat respon dan kondisi terbaik robot berdasarkan penentuan nilai K_p , K_i , K_d yang dapat dilihat langsung dari *display* LCD 16x2.

3.1.5 Rangkaian optocoupler switch



Gambar 3. 5 Rangkaian optocoupler switch

Keterangan:

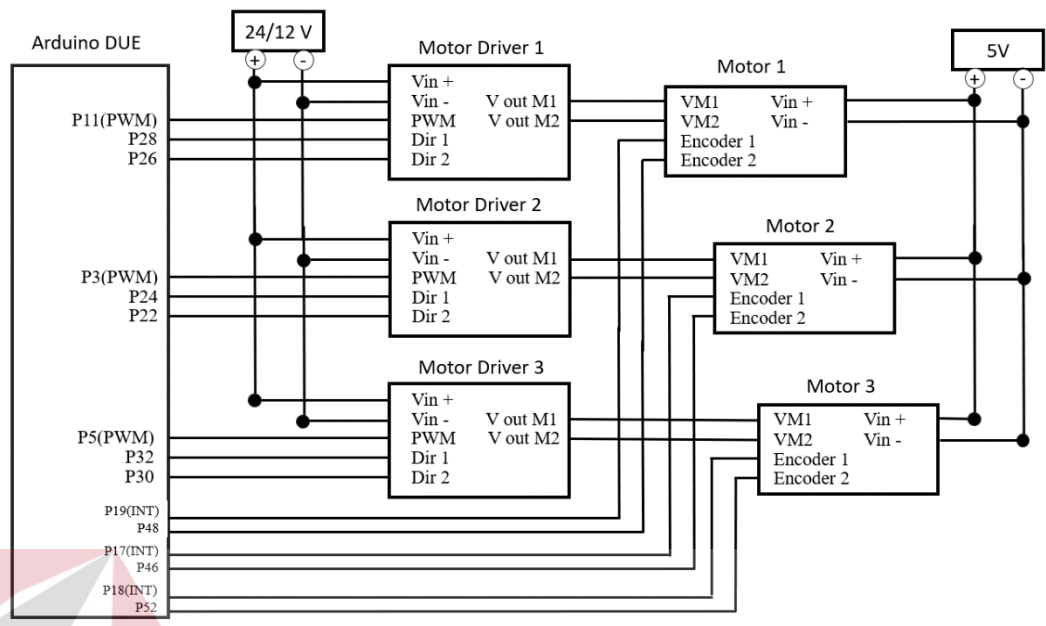
JP = Jumper point (pin penghubung)

R = Resistor

TLP621 = Optocoupler

Rangkaian switch optocoupler digunakan untuk mengurangi dan menghilangkan tegangan feedback besar yang dihasilkan dari tegangan 24v yg terhubung ke external rotary encoder. Karena pin Arduino tidak bisa menerima tegangan lebih dari 3.3 v maka menggunakan rangkaian ini sangat penting pada prototipe mobile robot ini.

3.1.6 Rangkaian driver dan motor dc



Gambar 3. 6 Rangkaian driver dan motor dc

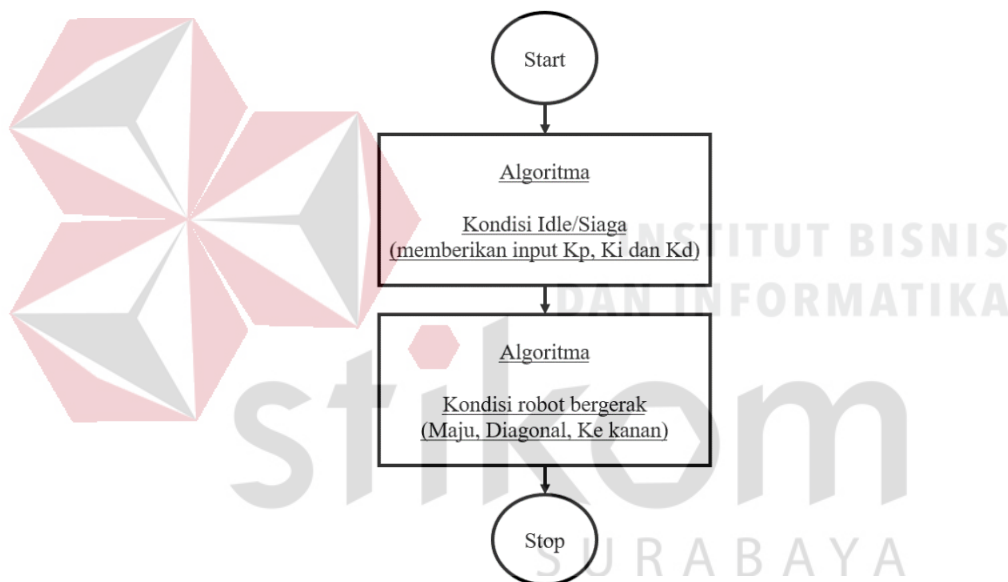
Pada prototipe mobile robot ini menggunakan 2 sumber daya yaitu 24/12v dari baterai Li-Po dan 5v dari UBEC. untuk kebutuhan actuator robot (input motor driver, motor DC, external rotary encoder) membutuhkan tegangan 12/24v karena membutuhkan tegangan dan arus yang besar, sedangkan untuk penggunaan sumber daya 5v digunakan untuk mikrokontroler, motor driver dan rotary encoder pada motor. Motor driver dengan tipe EMS-30A digunakan untuk memenuhi kebutuhan motor DC yang membutuhkan kapasitas ampere yang besar yaitu 30A.

Pada driver motor terdapat 3 jenis inpu yang digunakan untuk direksi 1, direksi 2 dan pengturan PWM. Pada input direksi digunakan untuk mengatur arah putaran motor yaitu berputar searah jarum jam atau berlawanan jarum jam, sedangkan untuk input PWM digunakan untuk memberikan masukan pulsa dengan perbandingan duty cycle sesuai dengan kebutuhan pada penggunaan motor DC. Rotary encoder munggunakan rotary encoder internal yang ada pada motor DC. Dalam membaca arah putaran motor maka digunakanlah kedua output dari rotary

encoder yaitu output A dan output B, dikarenakan kedua output tersebut mempunyai output *pulse* yang berdampingan. Salah satu output dari rotary encoder dihubungkan kepada pin kotntroler yang telah diset menjadi pin interrupt untuk melakukan pembacaan putaran.

3.2. Perancangan perangkat lunak

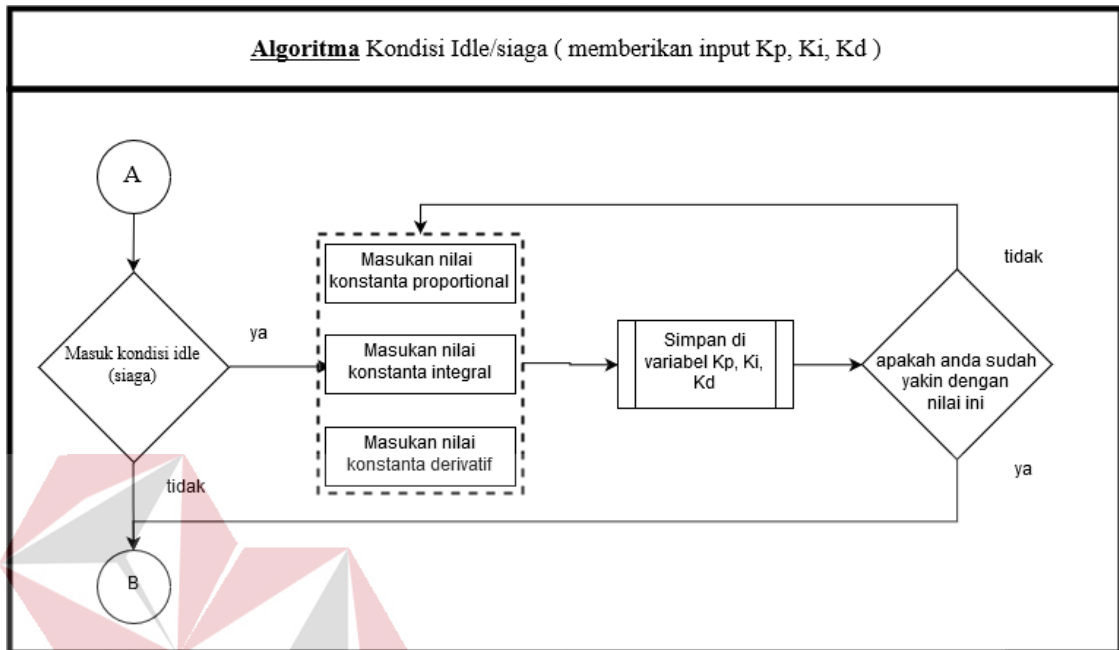
Perancangan flowchart untuk prototipe mobile robot menggunakan PID seperti yang terdapat pada gambar 3.6



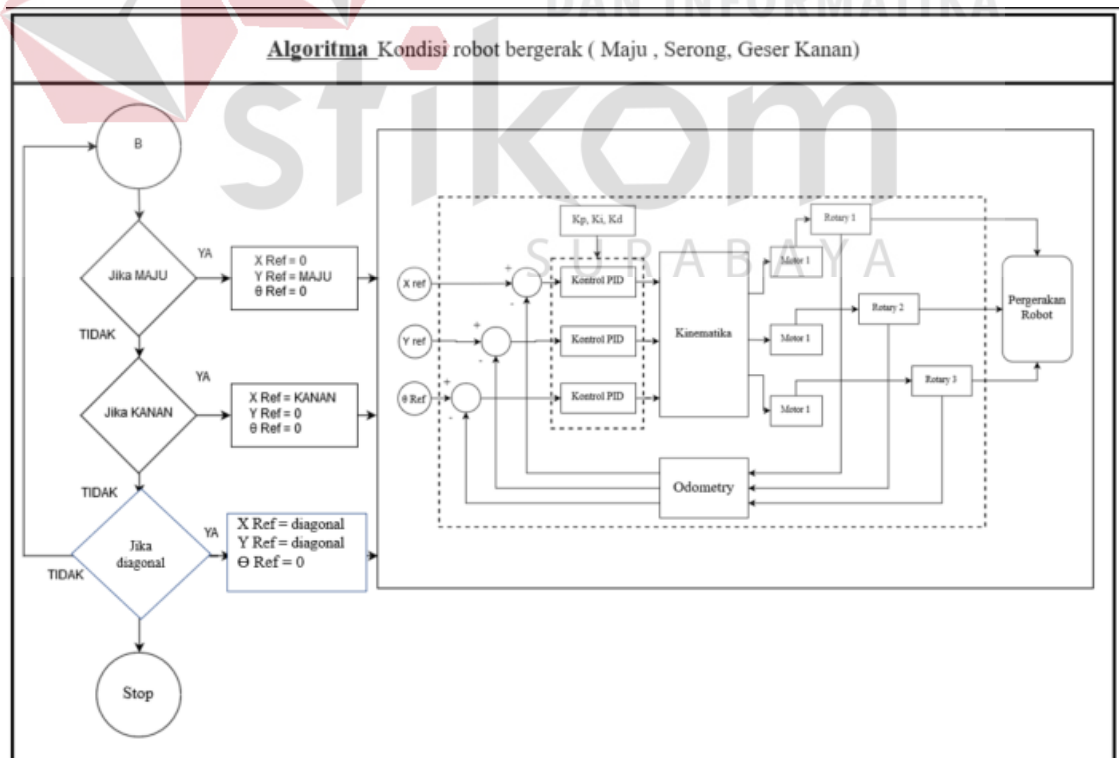
Gambar 3. 7 flowchart keseluruhan

Proses eksekusi algoritma yang ditanamkan pada prototipe mobile robot ini memiliki 2 bagian yaitu kondisi idle (siaga) dan kondisi running, masing-masing bagian memiliki fungsi masing-masing yaitu pada kondisi idle (siaga) robot menunggu masukan berupa masukan nilai variabel kontrol proporsional, integral dan derivative yang akan disimpan pada Kp, Ki dan Kd yang nanti akan diproses pada waktu kondisi running. Sedangkan untuk kondisi running adalah kondisi

dimana robot mengeksekusi perintah berupa status pergerakan robot, jarak tempuh robot dan nilai KP, Ki, Kd yang sudah di masukan.

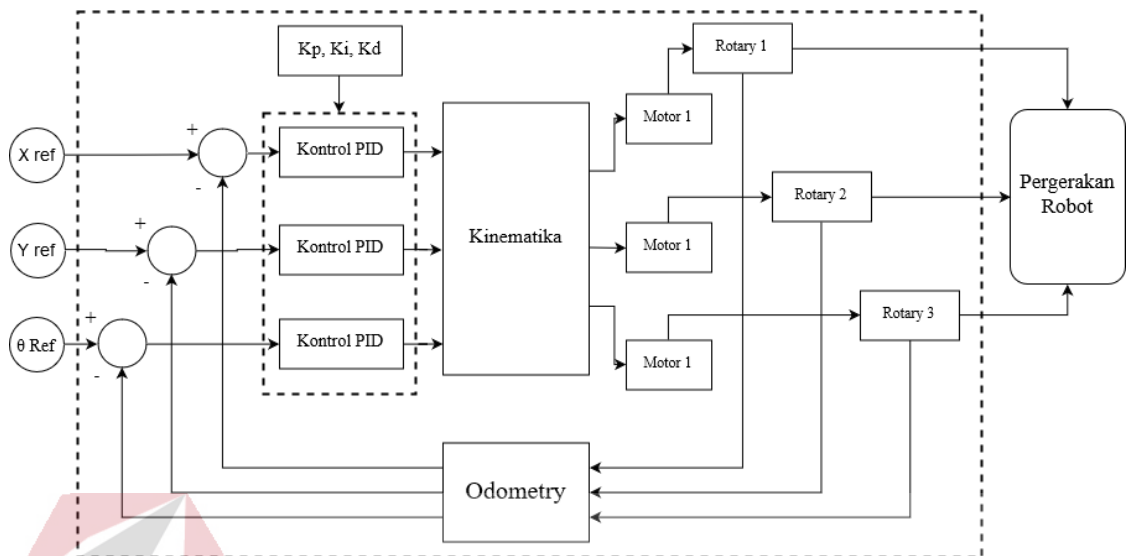


Gambar 3. 8 Algoritma kondisi idle (siaga)



Gambar 3. 9 Algoritma kondisi bergerak

3.2.1 Blok diagram sistem



Gambar 3. 10 blok diagram sistem

Pada blok diagram diatas ditunjukkan bahwa input sitem berasal dari nilai kordinat refrensi yaitu X_{ref} , Y_{ref} dan θ_{Ref} . Nilai refrensi ini digunakan sebagai tujuan pergerakan robot. Untuk menuju titik tujuan dan bisa membedakan robot bergerak maju, mundur, ke kanan, ke kiri maupun diagonal perlu dilakukan kontrol kinematika. kontrol navigasi kinematika digunakan untuk membagi proporsi kecepatan masing-masing motor, sehingga motor dapat dikontrol dengan permodelan matematis yang secara garis besar bisa membedakan gerakan.

Kontrol kinematika bertujuan untuk membagi proporsi untuk masing-masing motor namun tidak memiliki kemampuan untuk mempercepat kontrol sistem menuju setpoint yang telah ditentukan oleh kordinat refrensi sehingga membutuhkan kontrol PID untuk mempercepat dan menyetabilkan system tersebut, berikut ini perumusan kontrol PID pada sistem:

$$PID = KP \times error + Ki \times (error + error_{akhir}) + Kd \times (error - error_{akhir}) \quad (3.1)$$

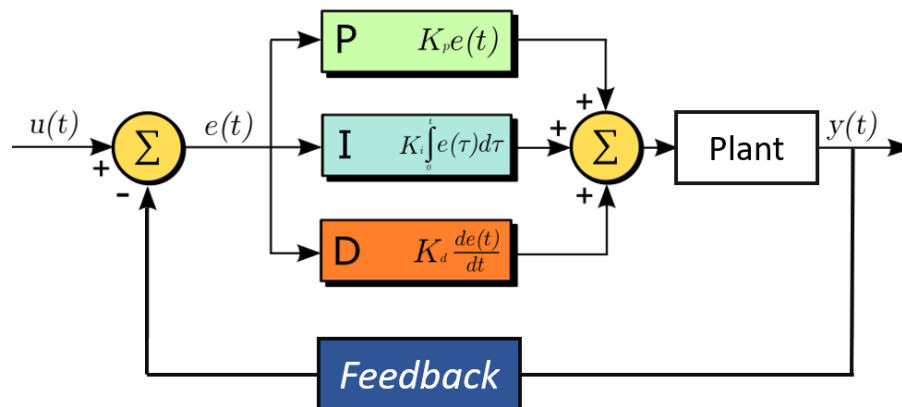
Keterangan:

PID	= Kontrol Proportional Integral Derivatif
Error	= Nilai Setpoint – Nilai Aktual
Nilai Setpoint	= Nilai yang sudah ditentukan
Nilai Aktual	= Nilai saat ini.
Kp	= konstanta untuk pengali error Proportional
Ki	= konstanta untuk pengali error Integral
Kd	= konstanta untuk pengali error Derivatif

Dengan menggunakan rumus kontrol PID sistem akan lebih cepat untuk merespon sistem dalam meminimalisir error yang terjadi sehingga membuat system lebih stabil. Perhitungan odometry mempunyai input berasal dari kombinasi ketiga rotary internal yang sudah terdapat pada motor DC, sistem umpan balik odometry ini selalu melakukan koreksi error terhadap posisi aktual dengan posisi setpoint yang ditetapkan meskipun dalam kondisi tidak menerima perintah untuk melakukan pergerakan. Oleh karena itu dalam keadaan diam robot selalu melakukan koreksi posisi aktual terhadap setpoint, sehingga saat mendapat gangguan pergerakan dari luar yang secara paksa robot dapat kembali ke posisi terakhir perintah diberikan.

3.2.2 Kontrol Proportional Integral Derivatif (PID)

Kontrol PID digunakan untuk membuat robot mampu merespon dengan cepat dan memiliki respon kontrol yang stabil. Kelebihan tersebut karena kontrol PID memiliki fungsi untuk mengolah sinyal error yang berasal dari sistem untuk diolah kembali menjadi nilai yang stabil. Berikut diagram blok kontrol PID secara umum dapat dilihat pada Gambar 3.10



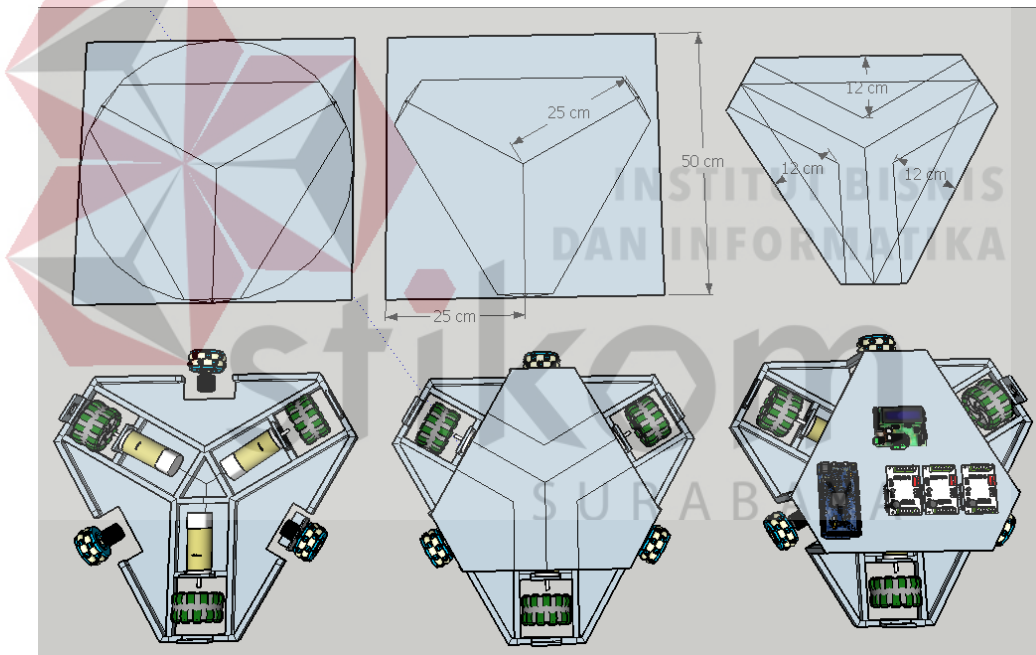
Gambar 3. 11 Diagram blok kontrol PID

Penjelasan dari blok diagram pada Gambar 3.9 yaitu nilai $r(t)$ merupakan nilai setpoint yaitu nilai yang ingin dicapai, sedangkan $y(t)$ merupakan hasil output pada sistem PID yang ingin di dikendalikan sampai nilainya sama dengan setpoint. Lalu ada nilai $e(t)$ atau nilai error yaitu selisih nilai output dengan nilai setpoint. Nilai error ini dimasukkan ke parameter-parameter pengendali PID yang berupa proporsional, integral, dan derivatif. Nilai $u(t)$ adalah hasil penjumlahan dari keseluruhan bagian proporsional, integral, dan derivative pada pengendali PID, sinyal umpan balik digunakan sebagai present value (error terakhir). Dalam perancangan sistem kendali PID adalah mengatur konstanta proporsional (K_p), integral (K_i), dan derivatif (K_d) agar respon pada sinyal keluaran sistem terhadap masukan menjadi tepat. Kemudian nilai tersebut dimasukkan ke plant atau aktuator untuk melakukan proses pengendalian. Keluaran yang dihasilkan dihitung kembali menggunakan sensor serta dibandingkan dengan nilai setpoint atau biasa yg disebut feedback.

3.3 Perancangan mekanik

3.3.1 Perancangan sistem actuator robot

Pada tahap perancang mekanik prototipe robot ini, *mobile robot* dirancang dengan penggerak utama menggunakan tiga buah motor DC dan roda omni directional. Penggunaan roda omni ini bertujuan agar robot dapat bergerak atau bermanuver lebih lincah dibanding dengan menggunakan roda biasa pada umumnya. Pada sub bab ini dijelaskan proses pembuatan robot mulai dari pembuatan *frame* robot sampai pemasangan motor dan sensor *rotary encoder* yang digunakan pada robot.



Gambar 3. 12 Desain frame pada sketup 2016

Sebelum melakukan proses pembuatan mekanik robot, perlu dilakukan proses desain dari bentuk mekanik robot. Hal ini dilakukan agar dalam proses pembuatan mekaniknya tidak asal-asalan dalam memotong atau memproses bahan materialnya, sehingga dapat meminimalkan kesalahan dalam proses pengerjaannya serta sebagai acuan kita dalam merakit part-part yang telah selesai kita buat. Berikut

ditunjukkan rancangan mekanik robot yang dibuat dengan menggunakan *software sketup 2016* yang ditampilkan dari pandangan atas dari mulai desain awal pembuatan hingga pembuatan selesai. Yang ditunjukkan oleh gambar 3.12.

3.3.2 Pembuatan *Frame* Robot

Bahan yang digunakan sebagai *frame* untuk robot ini menggunakan bahan *aluminium hollow*, bahan ini dipilih karena kelebihan yang kuat, tidak mudah bengkok, tidak mudah berkarat dan penyatuan antar part tidak memerlukan pengelasan. Pada Gambar 3.12 di bawah ini ditunjukkan bahan material yang akan digunakan sebagai *frame* robot.

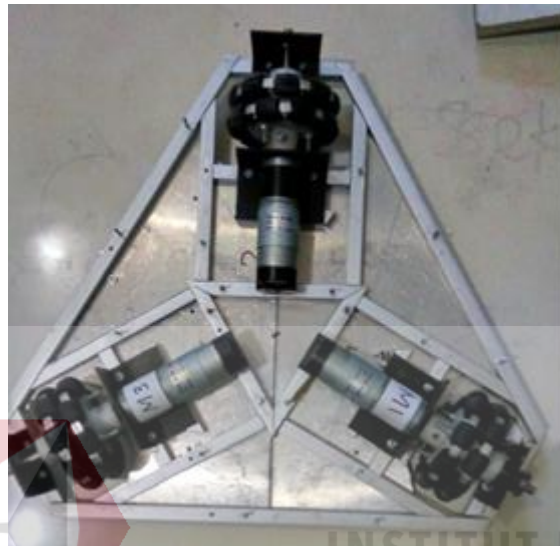


Gambar 3. 13 Bahan pembuatan frame robot

Untuk merangkai bahan material tersebut menjadi seperti desain robot yang telah dibuat sebelumnya dilakukan beberapa tahap diantaranya pemotongan bahan sesuai dimensi yang telah ditentukan, setelah itu dilakukan proses perakitan antar komponen yang sudah dibuat dengan cara dihubungkan menggunakan plat aluminium yang sudah di bor dan dipasang baut.

Hal yang perlu diperhatikan ketika selesai proses penggabungan adalah pengecekan *frame* robot yang telah terbentuk, *frame* harus rata dengan permukaan

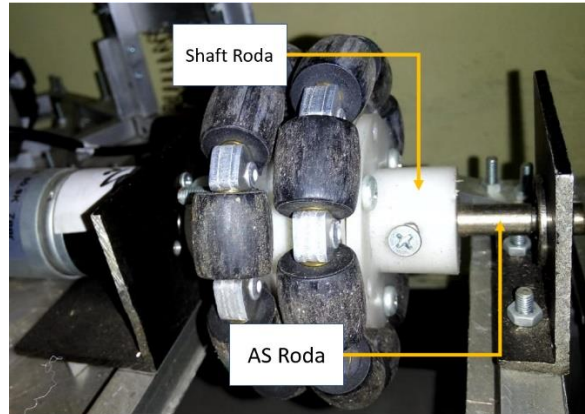
ketika diletakkan di atas lantai yang datar dan tidak boleh ada sisi yang masih mengambang karena ini bisa mempengaruhi dinamika dari robot ketika robot tersebut dipasang dengan roda. Jika ada bagian yang mengambang maka rawan terjadi slip pada roda ketika roda tersebut digerakkan.



Gambar 3. 14 Frame Robot yang sudah dirangkai

3.3.3 Pembuatan *Shaft* motor

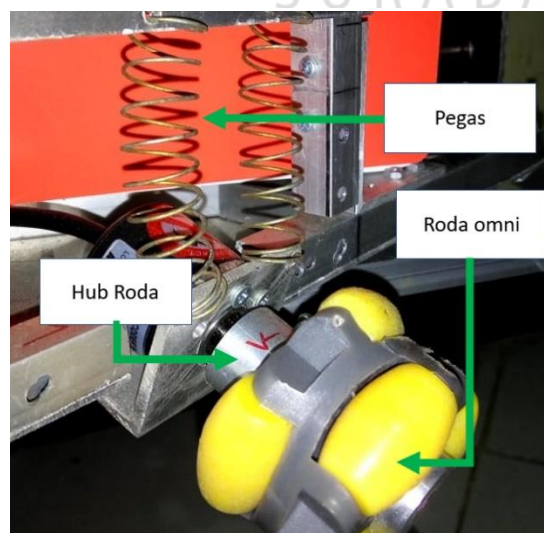
Untuk menghubungkan antara *shaft* motor dengan roda omni tidak bisa dilakukan secara langsung tetapi membutuhkan sebuah alat penghubung yakni as dari bahan besi yang ujung satunya dipasang ke *shaft* motor dan ujung satunya dihubungkan ke poros roda. Sebelum memasang as ke roda perlu dibuat *shaft* yang bertujuan untuk menghubungkan roda omni dengan as roda. *shaft* terbuat dari bahan teflon jenis *PE*, bahan ini cocok digunakan karena proses manufakturnya mudah dan cepat ketika di bubut menjadi seperti *shaft* roda. Setelah proses pembubutan selesai langkah selanjutnya dilakukan proses pengeboran untuk membuat lubang yang digunakan sebagai poros untuk menghubungkan *shaft* dengan roda.



Gambar 3. 15 Pemasangan Shaft roda dan AS roda

3.3.4 Pembuatan Hub dan As untuk *Rotary Encoder*

Setelah proses pemasangan motor utama dengan roda omni selesai dilakukan, proses mekanik berikutnya adalah pembuatan Hub yang menghubungkan antara roda omni dengan sensor rotary. Untuk menghindari slip pada penggunaan sensor rotary maka ditambahkan 2 pegas pada sisi kanan dan kiri untuk setiap sensor rotary, meskipun robot terangkat karena bidang atar yang bergelombang maka sensor akan terus berputar karena roda omni selalu mengikuti bentuk bidang datar tersebut.



Gambar 3. 16 Pemasangan Roda Omni dengan Rotary Encoder

BAB IV

PENGUJIAN DAN ANALISA SISTEM

Pada bab ini akan membahas tentang pengujian prototipe mobile robot berdasarkan perancangan sistem yang sudah dibuat. Tujuan pengujian dilakukan untuk mengetahui hasil dari sistem kontrol PID apakah sudah sesuai atau belum sesuai dalam implementasi pada robot. Pengujian dilakukan dalam beberapa tahap yaitu tahap terpisah untuk menguji tiap-tiap bagian yang digunakan dalam prototipe robot dan tahap keseluruhan yaitu tahap pengujian keseluruhan pada proyek tugas akhir sistem kontrol kecepatan robot *three omni directional* menggunakan metode PID. Pengujian yang dilakukan terdibagi menjadi beberapa bagian, yaitu:

1. Pengujian mikrokontroler Arduino DUE
2. Pengujian tombol
3. LCD 16x2
4. Pengujian sensor *external rotary encoder*
5. Pengujian kecepatan putar motor
6. Pengujian sistem kontrol PID

4.1 Pengujian Arduino DUE

4.1.1 Tujuan

Pengujian ini dilakukan untuk mengetahui perangkat mikrokontroler Arduino apakah masih berjalan dengan baik, serta dapat mengeksekusi program dengan benar.

4.1.2 Alat yang digunakan

1. Board mikrokontroler Arduino DUE.
2. Kabel *downloader* Arduino.
3. Leptop.
4. Arduino IDE.
5. Program Arduino.
6. *Power supply 9v*

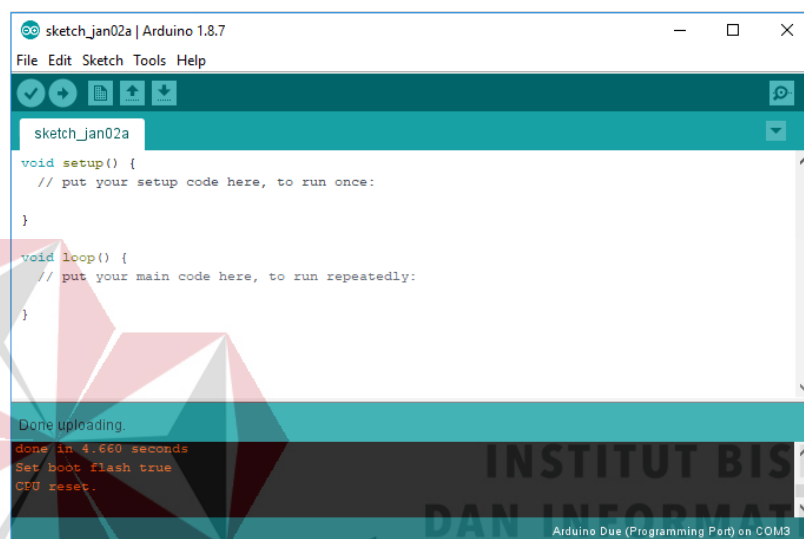
4.1.3 Prosedur pengujian

1. Menghidupkan komputer.
2. Sambungkan Arduino dengan komputer melalui kabel downloader.
3. Buka aplikasi Arduino IDE.
4. Pilih tipe Arduino yang akan dipakai.
5. Pilih port yang digunakan.
6. Buat program sederhana kemudian upload ke Arduino.
7. Kemudian *verify* untuk memastikan program yang dibuat benar.
8. Kemudian upload program ke Arduino.

9. Jika terdapat notifikasi “done uploading” maka Arduino masih berfungsi normal.

4.1.4 Hasil Pengujian

Dari pengujian diatas maka hasil nya dapat dilihat pada gambar



```

sketch_jan02a | Arduino 1.8.7
File Edit Sketch Tools Help
sketch_jan02a
void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}

Done uploading.
done in 4.660 seconds
Set boot flash true
CPU reset.
Arduino Due (Programming Port) on COM3

```

Gambar 4. 1 done uploading

4.2 Pengujian tombol input

4.2.1 Tujuan

Pengujian tombol digunakan untuk memberikan input digital yang akan dimasukan pada pin Arduino. Dalam pengujian ini memastikan bahwa tombol masih berfungsi dengan baik untuk digunakan sebagai input nilai konstanta P.I.D.

4.2.2 Alat yang digunakan

1. Komputer
2. Arduino DUE
3. Kabel downloader

4. Rangkain tombol
5. Kabel peenghubung

4.2.3 Prosedur pengujian

1. Hubungkan rangkaian tombol pada pin Arduino
2. Buka software Arduino IDE
3. Membuat program sederhana untuk pegujian tombol.
4. Hubungkan kabel downloader pada Arduino
5. Upload program yang sudah dibuat.
6. Pastikan bahwa program yang dibuat tidak ada yang salah dan pada proses upload adalah sukses

4.2.4 Hasil pengujian

Pengujian pada tombol menggunakan program sederhana yang digunakan untuk memeriksa apakah tombol bdalam kondisi baik kemudian sinyal input yang dikirim akan ditampilkan pada serial monitor Arduino.

```
const int t_tes = A0;
int tombol_tes;
void setup() {
  // put your setup code here, to run once:
  Serial.begin (115200);
  pinMode(t_tes, INPUT);
}

void loop() {
  // put your main code here, to run repeatedly:
  tombol_tes = digitalRead(t_tes);

  if (tombol_tes == LOW)
  {
    Serial.println(" tes berhasil ");
    delay(500);
  }
}
```

Gambar 4. 2 program sederhana tombol pada arduino

```
tes tombol berhasil  
tes tombol berhasil  
tes tombol berhasil  
tes tombol berhasil  
tes tombol berhasil  
tes tombol berhasil  
tes tombol berhasil  
tes tombol berhasil  
tes tombol berhasil
```

Gambar 4. 3 Tampilan pada serial monitor

4.3 Pengujian LCD

4.3.1 Tujuan

Pengujian LCD dilakukan untuk menampilkan nilai Kp, Ki, Kd. Pengujian ini dilakukan untuk mengetahui apakah LCD sudah bekerja dengan baik.

4.3.2 Alat yang digunakan

1. Komputer
2. Arduino Due dan kabbel downloader
3. Rangkaian LCD
4. Arduino IDE

4.3.3 Prosedur pengujian

1. Hubungkan rangkaian LCD pada pin Arduino
2. Buka software Arduino IDE
3. Membuat program sederhana untuk pengujian LCD
4. Hubungkan kabel downloader pada Arduino
5. Upload program yang sudah dibuat
6. Amati hasil yang ditampilkan pada LCD

4.3.4 Hasil pengujian

Pengujian LCD digunakan untuk menampilkan data yang diproses pada Arduino salah satunya untuk menampilkan nilai konstanta PID. Hasil pengujian dapat dilihat pada gambar 4.4

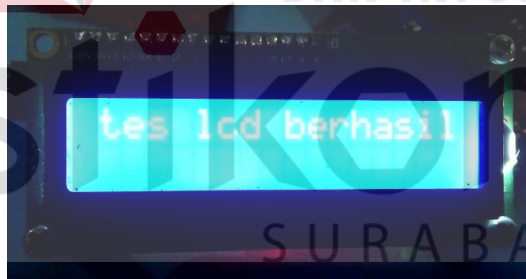
```
tes_lcd
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x3F, 16, 2);

void setup() {
  // put your setup code here, to run once:
  lcd.begin();
  lcd.setCursor(0,0);
  lcd.print("tes lcd berhasil");
}

void loop() {
  // put your main code here, to run repeatedly:
}
```

Gambar 4. 4 Program LCD



Gambar 4. 5 Hasil pengujian LCD

4.4 Pengujian external rotary encoder

4.4.1 Tujuan

Pengujian rotary encoder dilakukan untuk menampilkan nilai putaran sensor rotary. Pengujian ini dilakukan untuk mengetahui apakah rotary encoder sudah bekerja dengan baik.

4.4.2 Alat yang digunakan

1. Komputer
2. Arduino Due dan kabbel downloader
3. Rangkaian external rotary
4. Arduino IDE

4.4.3 Prosedur pengujian

1. Hubungkan rangkaian external rotary pada pin Arduino
2. Buka software Arduino IDE
3. Membuat program sederhana untuk pengujian external rotary
4. Hubungkan kabel downloader pada Arduino
5. Upload program yang sudah dibuat
6. Amati hasil yang ditampilkan pada serial monitor

4.4.4 Hasil pengujian

Pengujian ini digunakan untuk menampilkan data yang dihasilkan oleh putaran sensor rotary external yang ditampilkan pada serial monitor arduino. Hasil pengujian dapat dilihat pada gambar 4.5

```

int r_kiri=0, r_kirib=49;
int ta;
float sa;

void int_kiri()
{
  if(digitalRead(r_kirib)== 1)
  {
    r_kiri++;
    ta++;
  }
  else
  {
    r_kiri--;
    ta--;
  }
  sa=sa+1;
  Serial.print("Nilai rotari");
  Serial.print(" ");
  Serial.print(r_kiri);
  Serial.println(" ");
}

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  attachInterrupt(digitalPinToInterrupt(51), int_kiri, RISING);
  pinMode(r_kirib, INPUT);
}

```

Gambar 4. 6 Program sederhana external rotary

COM5 (Arduino Due (Programming Port))

```

Nilai rotari 0
Nilai rotari 1
Nilai rotari 2
Nilai rotari 3
Nilai rotari 4
Nilai rotari 5
Nilai rotari 6
Nilai rotari 7
Nilai rotari 8
Nilai rotari 9
Nilai rotari 10

```

Gambar 4. 7 Hasil pengujian external rotary

4.5 Pengujian kecepatan motor

4.5.1 Tujuan

Pengujian ini digunakan untuk mengetahui perbedaan kecepatan masing-masing motor yang digunakan dengan cara memberikan input kecepatan berupa PWM pada motor dan mengamati respon output kecepatan pada masing-masing motor (rpm). Dengan diketahuinya perbedaan respon kecepatan, nilai tersebut dapat ditentukan sebagai nilai untuk mensinkronkan kecepatan seluruh motor.

4.5.2 Alat yang digunakan

1. Komputer
2. Prototipe robot
3. Program untuk menampilkan RPM
4. Arduino IDE

4.5.3 Prosedur pengujian

1. Hubungkan komputer dengan prototipe robot.
2. Buka software Arduino IDE
3. Hubungkan kabel downloader pada Arduino DUE
4. Upload program yang sudah dibuat
5. Amati hasil yang ditampilkan pada serial monitor

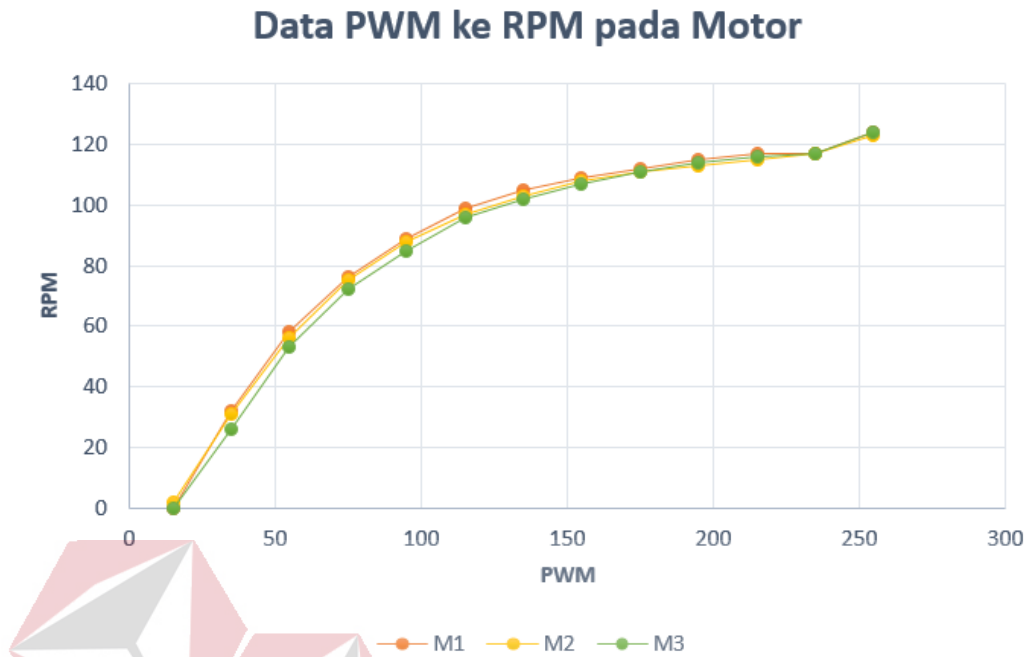
4.5.4 Hasil pengujian

Pengujian ini digunakan untuk menampilkan data yang dihasilkan oleh kecepatan putaran sensor rotary internal pada setiap motor yang ditampilkan pada serial monitor arduino. Hasil pengujian dapat dilihat pada tabel 4.1

Tabel 4. 1 Hasil pengujian motor

No	PWM	M1 (RPM)	M2 (RPM)	M3 (RPM)
1	255	124	123	124
2	235	117	117	117
3	215	117	115	116
4	195	115	113	114
5	175	112	111	111
6	155	109	108	107
7	135	105	103	102
8	115	99	97	96
9	95	89	88	85
10	75	76	75	72
11	55	58	56	53
12	35	32	31	26
13	15	0	2	0

Berikut adalah data pengujian dalam bentuk grafik pada gambar 4.7:



Gambar 4. 8 Grafik PWM dan RPM Motor

Hasil dari proses pengujian kecepatan masing-masing motor menunjukkan bahwa motor yang terpasang pada robot sepak bola beroda dalam Tugas Akhir ini mempunyai perbedaan kecepatan yang tidak signifikan, Seperti yang terlihat bahwa pada saat diberi input PWM yang tinggi kecepatan pada Motor 2 relatif lebih pelan dari motor lainnya, sedangkan pada saat diberi input PWM yang rendah Motor 3 yang berkecepatan lebih pelan dari motor lainnya.

4.6 Pengujian sistem keseluruhan

4.6.1 Tujuan

Pengujian dilakukan untuk mendapatkan nilai K_p , K_i , K_d terbaik untuk masing-masing pergerakan robot yaitu robot akan bergerak maju dengan jarak 400 cm menggunakan batas kecepatan (pwm) 100, 150, 200 kemudian akan Robot bergerak ke arah kanan menggunakan batas kecepatan (pwm) 100, 150, 200 dan bergerak diagonal ke arah kanan dengan menggunakan batas kecepatan (pwm) 100, 150, 200 sehingga nilai dan respon yang di hasilkan akan bervariasi berdasarkan hal-hal tersebut. Pengujian ini dilakukan untuk mengetahui kestabilan respon pada sistem kontrol robot dan waktu yang diperoleh setelah berhasil sampai tujuan.

4.6.2 Alat yang digunakan

1. Komputer
2. Prototipe robot
3. Lantai keramik dengan ukuran 4 m x 4 m
4. Stopwatch
5. Arduino IDE

INSTITUT BISNIS
DAN INFORMATIKA

stikom
SURABAYA

4.6.3 Prosedur pengujian

1. Rumus kontroler PID

$$\text{Kontroler PID} = \text{Kontroler P} + \text{Kontroler I} + \text{Kontroler D} \quad (4.1)$$

$$\text{Kontroler P (proportional)} = K_p \times \text{error} \quad (4.2)$$

$$\text{Kontroler I (integral)} = \left(\frac{K_i}{10}\right) \times (\text{error} + \text{error}_{\text{akhir}}) \times T_s \quad (4.3)$$

$$\text{Kontroler D (Derivative)} = (K_d \times (\text{error} - \text{error}_{\text{akhir}})) / T_s \quad (4.4)$$

Keterangan:

PID	= Kontrol Proportional Integral Derivatif
Error	= Nilai Setpoint – Nilai Aktual
Nilai Setpoint	= Nilai tujuan
Nilai Aktual	= Nilai saat ini yang sedang berjalan.
Kp	= konstanta pengali Proportional
Ki	= konstanta pengali Integral
Kd	= konstanta pengali Derivatif
Ts	= Time Sampling

Pada suatu sistem kontrol yang menggunakan metode PID setiap pengguna bebas menentukan seperti apa sistem kontrol yang akan mereka jalankan. Hal tersebut terjadi karena rumus dasar PID yang digunakan masih membuat sistem yang dibangun kurang stabil. Sehingga pengguna bebas memberikan variasi seperti yang penulis lakukan. Perubahan tersebut berdasarkan pengujian yang dilakukan yaitu merubah rumus kontroler Integral yang awalnya seperti ini $K_i \times (\text{error} + \text{error}_{\text{akhir}}) \times T_s$ menjadi $\left(\frac{K_i}{10}\right) \times (\text{error} + \text{error}_{\text{akhir}}) \times T_s$. Alasan

perubahan tersebut karena pada saat pengujian sistem jika K_i bernilai lebih besar dari 1 maka sistem kontrol memiliki gerakan osilasi yang besar sehingga membuat robot tidak stabil namun jika nilai K_i berada pada rentang 0,1-1 maka gerakan osilasi sangat kecil sehingga membuat gerakan pada robot stabil. Nilai 0,01 – 1 berasal dari rumus berikut

$$K_i = \text{input user} / 10$$

(4.4)

Keterangan :

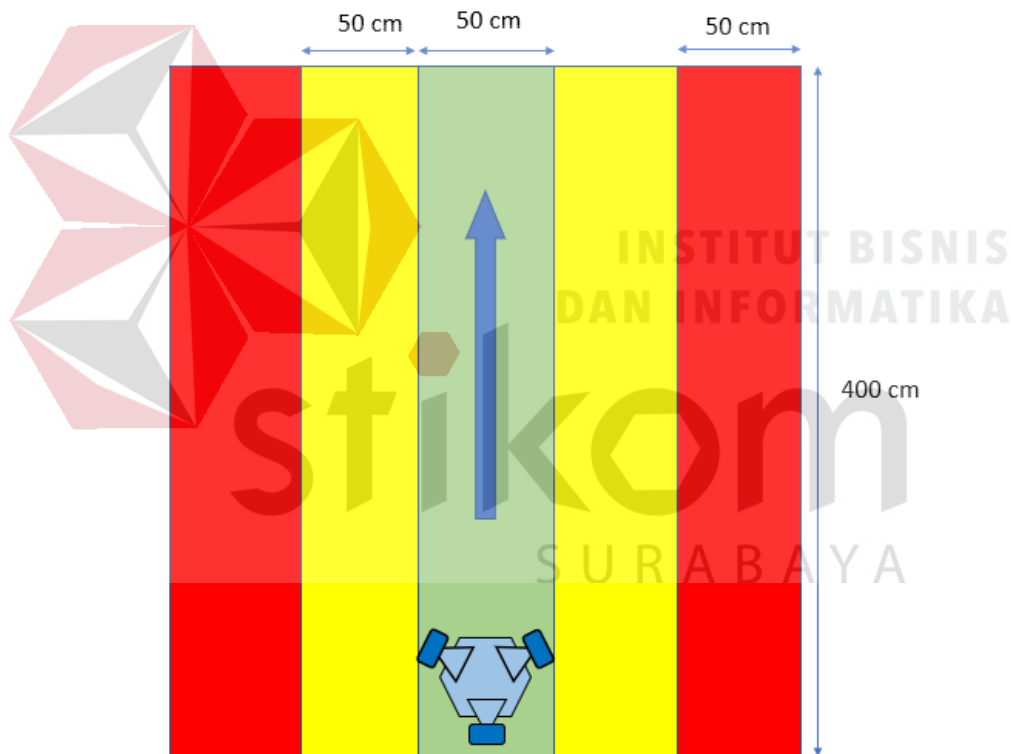
K_i = Konstanta pengali Integral

Input user = Nilai yang diberikan oleh *user* dengan angka 1 sampai 10



2. Pembagian gerakan untuk pengujian robot

A. Pergerakan robot maju digunakan untuk melakukan pengujian pergerakan maju pada robot untuk berjalan dengan menempuh jarak 400 cm. dalam gambar 4.9 terdapat 3 warna yang berbeda dan setiap warna mewakili daerah error yang berukuran 50 cm. Warna kuning tergolong pada eror jenis A, Warna merah tergolong pada eror jenis B, sesuai pada tabel pembagian daerah error, pembagian ini berfungsi untuk mengetahui apakah robot bergerak stabil dengan nilai error yang besar atau kecil.



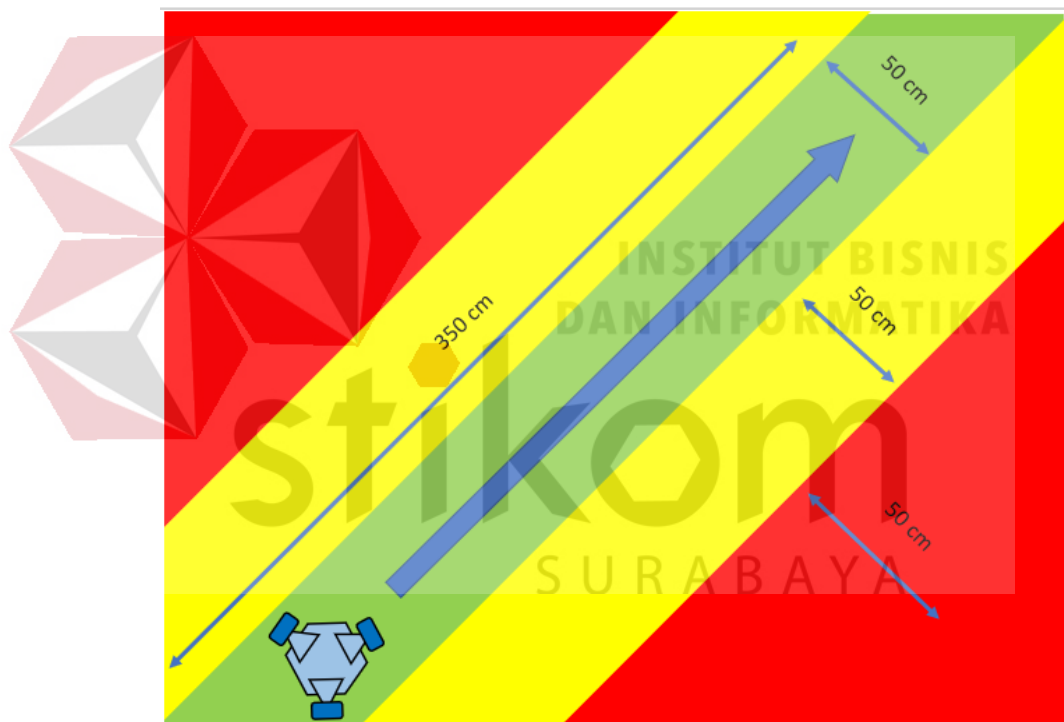
Gambar 4. 9 Gerak robot maju

B. Pergerakan robot ke arah kanan digunakan untuk melakukan pengujian pergerakan ke arah kanan pada robot untuk berjalan dengan menempuh jarak 400 cm. Dalam gambar 4.10 terdapat 3 warna yang berbeda dan setiap warna mewakili daerah error yang berukuran 50 cm. Warna kuning tergolong pada eror jenis A, Warna merah tergolong pada eror jenis B, sesuai pada tabel pembagian daerah eror. Pembagian ini berfungsi untuk mengetahui apakah robot bergerak stabil dengan nilai error yang besar atau kecil.



Gambar 4. 10 Gerak robot ke arah kanan

C. Pergerakan robot ke arah diagonal kanan digunakan untuk melakukan pengujian pergerakan ke arah diagonal kanan pada robot untuk berjalan dengan menempuh jarak 400 cm. Dalam gambar 4.11 terdapat 3 warna yang berbeda dan setiap warna mewakili daerah error yang berukuran 50 cm. Warna kuning tergolong pada eror jenis A, Warna merah tergolong pada eror jenis B, sesuai pada tabel pembagian daerah eror. Pembagian ini berfungsi untuk mengetahui apakah robot bergerak stabil dengan nilai error yang besar atau kecil.



Gambar 4. 11 Gerak robot diagonal kearah kanan

Pada pengujian ini terdapat 3 pembagian pada pergerakan robot. Pengujian tersebut meliputi pergerakan robot maju, robot bergerak kearah kanan, robot bergerak diagonal kearah kanan. Untuk setiap nilai K_p , K_i dan K_d memiliki fungsi untuk membuat robot menjadi stabil yaitu memiliki nilai eror yang kecil serta waktu tempuh yang cepat atau akan menurunkan

pergerakan robot menjadi tidak stabil yaitu memiliki nilai eror yang besar serta waktu tempuh yang lama untuk setiap gerakan, jika robot berada pada jalur hijau maka kontrol robot baik, jika berada pada jalur kuning maka kontrol robot kurang baik, jika robot berada di jalur merah maka kontrol robot eror seperti yang dijelaskan pada tabel 4.2 Tabel pembagian daerah eror.

D. Tabel pembagian daerah eror

Tabel 4. 2 Tabel pembagian daerah eror

Kategori	Keterangan
A	Robot bergerak stabil dengan error < +/- 30 cm, daerah dengan warna kuning.
B	Robot bergerak stabil dengan error < +/- 50 cm, daerah dengan warna merah.
C	Robot bergerak stabil sampai pertengahan jarak, kemudian bergerak tidak stabil sampai finish dan tidak berhenti
D	Error tinggi, robot tidak stabil, tidak sampai garis finish

3. Tabel pengamatan nilai Kp, Ki, Kd

Kp	Ki	Kd	Waktu	Error	Kp	Ki	Kd	Waktu	Error	Kp	Ki	Kd	Waktu	Error
Kecil	Kecil	Kecil			Kecil	Sedang	Kecil			Kecil	Besar	Kecil		
Kecil	Kecil	Sedang			Kecil	Sedang	Sedang			Kecil	Besar	Sedang		
Kecil	Kecil	Besar			Kecil	Sedang	Besar			Kecil	Besar	Besar		
Kp	Ki	Kd	Waktu	Error	Kp	Ki	Kd	Waktu	Error	Kp	Ki	Kd	Waktu	Error
Sedang	Kecil	Kecil			Sedang	Sedang	Kecil			Sedang	Besar	Kecil		
Sedang	Kecil	Sedang			Sedang	Sedang	Sedang			Sedang	Besar	Sedang		
Sedang	Kecil	Besar			Sedang	Sedang	Besar			Sedang	Besar	Besar		
Kp	Ki	Kd	Waktu	Error	Kp	Ki	Kd	Waktu	Error	Kp	Ki	Kd	Waktu	Error
Besar	Kecil	Kecil			Besar	Sedang	Kecil			Besar	Besar	Kecil		
Besar	Kecil	Sedang			Besar	Sedang	Sedang			Besar	Besar	Sedang		
Besar	Kecil	Besar			Besar	Sedang	Besar			Besar	Besar	Besar		

Gambar 4. 12 Tabel pengamatan nilai Kp, Ki dan Kd

Pengujian sistem kontrol PID pada prototipe robot ini adalah menggunakan metode *trial dan error*. Penulis memilih untuk menggunakan metode ini karena proses yang dilakukan untuk menentukan nilai Kp, Ki dan Kd sangat mudah untuk dipahami selain itu pada penggunaan metode ini memiliki kelebihan lain yaitu kita tidak perlu mengidentifikasi *plant* terlebih dahulu, tidak perlu membuat model matematis *plant*, tidak perlu menentukan parameter *plant* dengan grafis atau analisis. Karena pada metode pengujian ini yang perlu dilakukan hanya memasukan nilai Kp, Ki dan Kd secara terus menerus sampai mendapatkan kombinasi Kp, Ki dan Kd terbaik.

Pada pengujian menggunakan metode *trial dan error* memang sangat mudah untuk digunakan namun memiliki kekurangan dalam hal waktu. Karena untuk mendapatkan nilai kombinasi KP, Ki dn Kd terbaik memerlukan percobaan berkali-kali, sehingga waktu yang diperlukan untuk

percobaan sangat lama. Untuk mengatasi hal tersebut penulis mendapatkan sebuah cara yaitu membuat kelompok pemilihan nilai K_p , K_i dan K_d yang dijadikan sebuah tabel sehingga jika menggunakan metode trial dan error konvensional membutuhkan percobaan hingga ratusan kali namun jika menggunakan metode Tabel pengamatan nilai K_p , K_i dan K_d percobaan bisa dilakukan dibawah 50 kali. Untuk melakukan pengujian menggunakan metode tabel pengamatan K_p , K_i dan K_d terdapat tiga tahap yaitu tahap pertama Tabel K_p untuk mencari nilai K_p terbaik. Setelah mendapatkan nilai K_p terbaik berdasarkan pengujian yaitu memiliki eror kecil dan waktu tempuh yang cepat maka nilai K_p ini diasumsikan sebagai nilai K_{p_tetap} . Setelah didapatkan nilai K_p terbaik maka dilanjutkan pada tahap kedua yaitu mendapatkan nilai K_d terbaik, Setelah mendapatkan nilai K_d terbaik maka nilai K_d ini diasumsikan sebagai nilai K_{d_tetap} . Setelah didapatkan nilai K_p dan K_d terbaik maka dilanjutkan pada tahap ketiga yaitu mendapatkan nilai K_i terbaik, Setelah mendapatkan nilai K_i terbaik maka nilai K_i ini diasumsikan sebagai nilai K_{i_tetap} . Setelah mendapatkan nilai K_{p_tetap} , K_{d_tetap} dan K_{i_tetap} maka nilai tersebut yang akan digunakan untuk K_p , K_i dan K_d pada sistem kontrol robot.

a. Tahap mencari Nilai Kp

Kp	Ki	Kd	Waktu	Error	Kp	Ki	Kd	Waktu	Error	Kp	Ki	Kd	Waktu	Error
Kecil	Kecil	Kecil			Kecil	Sedang	Kecil			Kecil	Besar	Kecil		
Kecil	Kecil	Sedang			Kecil	Sedang	Sedang			Kecil	Besar	Sedang		
Kecil	Kecil	Besar			Kecil	Sedang	Besar			Kecil	Besar	Besar		
Kp	Ki	Kd	Waktu	Error	Kp	Ki	Kd	Waktu	Error	Kp	Ki	Kd	Waktu	Error
Sedang	Kecil	Kecil			Sedang	Sedang	Kecil			Sedang	Besar	Kecil		
Sedang	Kecil	Sedang			Sedang	Sedang	Sedang			Sedang	Besar	Sedang		
Sedang	Kecil	Besar			Sedang	Sedang	Besar			Sedang	Besar	Besar		
Kp	Ki	Kd	Waktu	Error	Kp	Ki	Kd	Waktu	Error	Kp	Ki	Kd	Waktu	Error
Besar	Kecil	Kecil			Besar	Sedang	Kecil			Besar	Besar	Kecil		
Besar	Kecil	Sedang			Besar	Sedang	Sedang			Besar	Besar	Sedang		
Besar	Kecil	Besar			Besar	Sedang	Besar			Besar	Besar	Besar		

Gambar 4. 13 Tabel pengamatan nilai Kp

Tahap mencari nilai Kp adalah proses pengujian untuk mendapatkan nilai Kp terbaik karena fungsi kontroler Proportional adalah respon kontrol sistem maka membutuhkan nilai dengan kriteria yaitu memiliki error paling kecil dan waktu eksekusi yang cepat. Untuk mendapatkan nilai tersebut dilakukan pengujian berdasarkan tabel pengamatan diatas. Yaitu memasukan nilai pada tabel berdasarkan aturan nilai yang sudah ditetapkan yaitu nilai kecil dengan anggota 1, 2, 3 dan 4. Untuk nilai sedang beranggotakan nilai 4, 5, 6 dan 7 sedangkan nilai besar dengan anggota 7, 8, 9, 10. Setelah mendapatkan komposisi nilai terbaik berdasarkan besarnya error dan waktu tercepat, sebagai contoh nilai yang dilingkari warna merah adalah komposisi terbaik, maka nilai Kp di simpan menjadi nilai Kp_tetap. Kemudian dilanjutkan pada tahap kedua untuk mencari nilai Kd terbaik.

b. Tahap mencari Nilai Kd

Kp	Ki	Kd	Waktu	Eror	Kp	Ki	Kd	Waktu	Eror	Kp	Ki	Kd	Waktu	Eror
Kp_tetap	Kecil	Kecil			Kp_tetap	Kecil	Kecil			Kp_tetap	Kecil	Kecil		
Kp_tetap	Sedang	Kecil			Kp_tetap	Sedang	Kecil			Kp_tetap	Sedang	Kecil		
Kp_tetap	Besar	Kecil			Kp_tetap	Besar	Kecil			Kp_tetap	Besar	Kecil		
Kp	Ki	Kd	Waktu	Eror	Kp	Ki	Kd	Waktu	Eror	Kp	Ki	Kd	Waktu	Eror
Kp_tetap	Kecil	Sedang			Kp_tetap	Kecil	Sedang			Kp_tetap	Kecil	Sedang		
Kp_tetap	Sedang	Sedang			Kp_tetap	Sedang	Sedang			Kp_tetap	Sedang	Sedang		
Kp_tetap	Besar	Sedang			Kp_tetap	Besar	Sedang			Kp_tetap	Besar	Sedang		
Kp	Ki	Kd	Waktu	Eror	Kp	Ki	Kd	Waktu	Eror	Kp	Ki	Kd	Waktu	Eror
Kp_tetap	Kecil	Besar			Kp_tetap	Kecil	Besar			Kp_tetap	Kecil	Besar		
Kp_tetap	Sedang	Besar			Kp_tetap	Sedang	Besar			Kp_tetap	Sedang	Besar		
Kp_tetap	Besar	Besar			Kp_tetap	Besar	Besar			Kp_tetap	Besar	Besar		

Gambar 4. 14 Tabel pengamatan nilai Kd

Tahap mencari nilai Kd adalah proses pengujian untuk mendapatkan nilai Kd terbaik Karena fungsi dari pengontrol derivative adalah meredam osilasi yang dihasilkan oleh kontrol proporsional. Sistem kontrol derivative mempunyai suatu karakter untuk mendahului, sehingga pengontrol ini dapat menghasilkan koreksi yang signifikan sebelum pembangkit kesalahan menjadi sangat besar sehingga memiliki fungsi menjaga kestabilan suatu sistem. Untuk mendapatkan nilai tersebut dilakukan pengujian berdasarkan tabel pengamatan diatas. Yaitu memasukan nilai pada tabel berdasarkan aturan nilai yang sudah ditetapkan yaitu nilai Kp berisi nilai yang disimpan pada pengujian pertama dengan nama Kp_tetap, kemudian memasukan nilai pada tabel berdasarkan aturan nilai yang sudah ditetapkan yaitu nilai kecil dengan anggota 1, 2, 3 dan 4. Untuk nilai sedang beranggotakan nilai 4, 5, 6 dan 7 sedangkan nilai besar dengan

anggota 7, 8, 9, 10. Setelah mendapatkan komposisi nilai terbaik kemudian simpan pada Kd_tetap dan melanjutkan pada tahap pencarian nilai Ki.

c. Tahap mencari Nilai Ki

Kp	Ki	Kd	Waktu	Error
Kp_tetap	1 – 10	Kd_tetap		

Gambar 4. 15 Tabel pengamatan nilai Ki

Diasumsikan nilai Kp_tetap adalah 4 dan nilai Kd_tetap adalah 4 sehingga tinggal mencari nilai Ki.

Kp	Ki	Kd	Waktu	Error
4	1	4		
4	2	4		
4	3	4		
4	4	4		
4	5	4		
4	6	4		
4	7	4		
4	8	4		
4	9	4		

Gambar 4. 16 Tabel pengamatan nilai Ki

Keterangan :

Tahap mencari nilai Ki adalah proses pengujian untuk mendapatkan nilai Ki terbaik, Karena fungsi dari pengontrol integral yaitu memperbaiki respon sistem dengan cara mengkoreksi kesalahan hingga menjadi nol. Syarat yang harus dipenuhi agar sistem stabil adalah Ki berada pada nilai dengan rentang

nilai yaitu 0,01-1. karena konstanta Integral (K_i) yang bernilai besar akan mempercepat hilangnya offset, namun semakin besar nilai konstanta K_i akan mengakibatkan peningkatan osilasi dari sinyal keluaran pengontrol dan membuat sistem semakin tidak stabil. Setelah menemukan komposisi nilai K_p , K_i dan K_d terbaik maka nilai tersebut dijadikan nilai PID.

d. Tahap mencari Nilai PID

K_p	K_i	K_d
4	5	4

Gambar 4. 16 Nilai PID terbaik

Keterangan :

Setelah melakukan pengujian dari beberapa tahap tersebut maka akan mendapatkan komposisi nilai K_p , K_i dan K_d terbaik yang bisa digunakan pada sistem kontrol tersebut dengan nilai eror paling kecil dan waktu eksekusi tercepat.

4. Inputkan nilai K_p , K_i , K_d pada tombol
5. Lihat hasil pergerakan robot.
6. Dan tampilkan nilai PID pada serial monitor arduino

A. Hasil pengujian untuk robot bergerak maju dengan kecepatan 100.

A.1 Pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Tabel 4. 3 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Nilai PID			Nilai Jarak (cm)		
X	Y	Z	X	Y	Z
0	0	4	0	0	0
0	100	4	0	399	0
13	100	0	0	396	0
9	100	0	0	391	0
9	100	4	0	387	0
9	100	4	0	382	0
18	100	4	0	379	0
13	100	4	0	374	0
4	100	4	0	369	0
4	100	4	0	365	0
0	100	-4	0	360	0
0	100	-8	0	356	0
-8	100	-4	0	353	0
-4	100	-9	0	348	0
-9	100	-4	0	344	0
-13	100	-8	0	340	0
-4	100	0	0	335	0
-4	100	-9	0	330	0
-9	100	-4	0	327	0
-9	100	-4	0	323	0
0	100	-9	0	319	0
-8	100	-4	0	314	0
-9	100	-4	0	310	0
0	100	0	0	305	0
0	100	-4	0	301	0
0	100	-4	0	297	0
9	100	0	0	292	0
4	100	0	0	288	0
4	100	0	0	283	0
4	100	0	0	279	0
4	100	0	0	275	0
0	100	-4	0	271	0
4	100	-4	0	267	0
13	100	0	0	262	0
13	100	0	0	258	0
9	100	0	0	253	0

Nilai PID			Nilai Jarak (cm)		
X	Y	Z	X	Y	Z
4	100	4	0	249	0
9	100	4	0	245	0
9	100	0	0	240	0
4	100	0	0	236	0
8	100	0	0	231	0
13	100	0	0	226	0
13	100	0	0	222	0
4	100	4	0	218	0
9	100	4	0	214	0
9	100	4	0	209	0
26	100	4	0	204	0
22	100	0	0	200	0
18	100	0	0	195	0
13	100	4	0	190	0
13	100	4	0	186	0
17	100	4	0	181	0
13	100	4	0	176	0
13	100	0	0	171	0
18	100	0	0	166	0
18	100	0	0	161	0
9	100	4	0	158	0
4	100	4	0	153	0
13	100	4	0	148	0
13	100	0	0	143	0
17	100	4	0	139	0
13	100	0	0	133	0
17	100	0	0	130	0
9	100	0	0	125	0
13	100	8	0	120	0
9	100	4	0	115	0
9	100	4	0	111	0
18	100	0	0	106	0
22	100	0	0	102	0
13	100	4	0	97	0
13	100	4	0	92	0
13	100	0	0	87	0
13	100	9	0	82	0
22	100	0	0	77	0
22	100	0	0	71	0
18	100	4	0	68	0
13	100	9	0	63	0

Nilai PID			Nilai Jarak (cm)		
X	Y	Z	X	Y	Z
18	100	0	0	58	0
13	100	0	0	53	0
13	100	8	0	49	0
17	100	0	0	44	0
22	100	0	0	40	0
18	100	0	0	35	0
4	100	4	0	30	0
9	100	4	0	26	0
13	100	4	0	21	0
17	100	9	0	17	0
26	100	0	0	13	0
18	100	0	0	8	0
4	100	4	0	3	0
9	-16	4	0	0	0
13	-22	8	0	0	0
9	-13	4	0	0	0
Waktu Tempuh (Detik)				16.02 detik	

Pada Tabel 4.3 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=5$ dan $K_d=4$. Hasil pngujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 30 cm. sehingga pada pengujian ini memiliki waktu tempuh yang lebih cepat yaitu 16.02 detik.

A.2 Pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Tabel 4. 4 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Nilai PID			Nilai Jarak (cm)		
X	Y	Z	X	Y	Z
0	0	0	0	0	0
4	100	0	0	397	0
9	100	0	0	395	0
10	100	0	0	390	0
15	100	4	0	384	0
5	100	4	0	380	0

Nilai PID			Nilai Jarak (cm)		
X	Y	Z	X	Y	Z
9	100	0	0	374	0
19	100	0	0	369	0
14	100	4	0	364	0
9	100	4	0	360	0
14	100	4	0	355	0
9	100	4	0	349	0
19	100	4	0	344	0
19	100	0	0	339	0
10	100	4	0	333	0
13	100	9	0	329	0
14	100	0	0	324	0
15	100	4	0	318	0
19	100	4	0	313	0
19	100	4	0	308	0
0	100	0	0	303	0
4	100	4	0	299	0
9	100	4	0	295	0
14	100	4	0	290	0
24	100	4	0	286	0
14	100	4	0	281	0
19	100	4	0	276	0
19	100	0	0	272	0
9	100	0	0	268	0
9	100	0	0	263	0
9	100	0	0	259	0
5	100	4	0	254	0
19	100	0	0	249	0
19	100	0	0	244	0
10	100	0	0	240	0
10	100	5	0	236	0
19	100	0	0	231	0
19	100	4	0	227	0
14	100	4	0	222	0
19	100	0	0	217	0
19	100	0	0	214	0
9	100	4	0	209	0
9	100	0	0	205	0
4	100	0	0	200	0
9	100	0	0	196	0
14	100	-4	0	191	0
5	100	0	0	188	0

Nilai PID			Nilai Jarak (cm)		
X	Y	Z	X	Y	Z
0	100	-4	0	183	0
0	100	0	0	178	0
9	100	0	0	173	0
10	100	0	0	169	0
5	100	4	0	165	0
19	100	4	0	161	0
20	100	4	0	156	0
5	100	4	0	151	0
9	100	0	0	147	0
9	100	-4	0	142	0
0	100	0	0	137	0
4	100	0	0	132	0
9	100	0	0	129	0
4	100	0	0	124	0
4	100	-4	0	119	0
4	100	-4	0	114	0
0	100	-4	0	109	0
4	100	-4	0	104	0
4	100	-4	0	100	0
0	100	0	0	95	0
4	100	-4	0	91	0
4	100	0	0	85	0
4	100	0	0	81	0
14	100	-4	0	76	0
4	100	-4	0	72	0
4	100	-4	0	67	0
9	100	0	0	62	0
14	100	0	0	57	0
10	100	0	0	53	0
19	100	0	0	48	0
19	100	4	0	44	0
9	100	4	0	40	0
15	100	0	0	35	0
24	100	4	0	30	0
15	100	4	0	26	0
19	100	5	0	21	0
19	100	4	0	16	0
5	100	0	0	13	0
14	100	4	0	9	0
19	100	4	0	4	0
10	-1	5	0	0	0

Nilai PID			Nilai Jarak (cm)		
X	Y	Z	X	Y	Z
15	-19	9	0	0	0
Waktu Tempuh (detik)				17.04 detik	

Pada Tabel 4.4 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=9$, $K_d=4$. Hasil pengujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 50 cm. sehingga pada pengujian ini memiliki waktu tempuh yang cepat yaitu dengan waktu tempuh 17.04 detik.

A.3 Pengujian menggunakan nilai $K_p=7$, $K_i=5$, $K_d=2$

Tabel 4. 5 Tabel pengujian menggunakan nilai $K_p=7$, $K_i=5$, $K_d=2$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
9	0	0	0	0	0
-100	0	-9	-399	0	0
-100	0	9	-395	0	0
-100	0	9	-393	0	0
-100	0	-9	-387	0	0
-100	0	0	-381	0	0
-100	0	18	-376	0	0
-100	0	-18	-370	0	0
-100	0	0	-364	0	0
-100	-9	18	-358	0	0
-100	0	-28	-352	0	0
-100	0	18	-346	0	0
-100	0	0	-340	0	0
-100	0	-37	-334	0	0
-100	0	55	-328	0	0
-100	0	-36	-323	0	0
-100	0	-1	-317	0	0
-100	0	37	-312	0	0
-100	27	-83	-306	0	0
-100	0	64	-301	0	0
-100	0	-26	-295	0	0

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
-100	27	-29	-289	0	0
-100	-18	93	-284	0	0
-100	9	-100	-278	0	0
-100	0	44	-273	0	0
-100	0	20	-267	0	0
-100	46	-57	-261	0	0
-100	-36	83	-256	0	0
-100	-28	10	-252	0	0
-100	18	-38	-246	0	0
-100	0	65	-241	0	0
-100	27	-82	-234	0	0
-100	0	45	-229	0	0
-100	0	1	-223	0	0
-100	27	-29	-217	0	0
-100	9	65	-212	0	0
-100	18	-83	-205	0	0
-100	0	45	-200	0	0
-100	0	-7	-194	0	0
-100	9	-29	-189	0	0
-100	0	47	-183	0	0
-100	0	-73	-176	0	0
-100	0	54	-171	0	0
-100	-18	-8	-165	0	0
-100	9	-20	-159	0	0
-100	0	56	-154	0	0
-100	0	-73	-148	0	0
-100	0	54	-142	0	0
-100	0	1	-137	0	0
-100	0	-29	-131	0	0
-100	9	65	-125	0	0
-100	9	-82	-119	0	0
-100	9	54	-114	0	0
-100	-18	-7	-108	0	0
-100	0	-83	-104	0	0
-100	0	36	-99	0	0
-100	27	10	-93	0	0
-100	-8	-56	-87	0	0
-100	-19	74	-81	0	0
-100	37	-54	-75	0	0
-100	-36	17	-70	0	0
-100	0	29	-65	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
Waktu Tempuh (detik)				tidak terdefinisi	

Pada Tabel 4.5 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=7$, $K_i=5$, $K_d=2$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar dan sulit untuk sistem kontrol membuat robot kembali dalam keadaan stabil yang memiliki dampak robot tidak bisa berhenti.

A.4 Pengujian menggunakan nilai $K_p=6$, $K_i=5$, $K_d=2$

Tabel 4. 6 Tabel pengujian menggunakan nilai $K_p=6$, $K_i=5$, $K_d=2$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
0	0	0	0	0	0
-100	0	0	-399	0	0
-100	22	0	-397	0	0
-100	7	0	-392	0	0
-100	0	7	-387	0	0
-100	7	0	-381	0	0
-100	7	-7	-376	0	0
-100	7	14	-370	0	0
-100	7	0	-364	0	0
-100	23	-15	-358	0	0
-100	-7	0	-352	0	0
-100	0	8	-346	0	0
-100	7	-14	-340	0	0
-100	0	6	-334	0	0
-100	0	15	-328	0	0
-100	-7	-30	-322	0	0
-100	0	6	-317	0	0
-100	0	24	-311	0	0
-100	7	-44	-305	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
-100	-14	21	-300	0	0
-100	0	9	-294	0	0
-100	0	-45	-288	0	0
-100	0	14	-282	0	0
-100	0	16	-277	0	0
-100	7	-14	-274	0	0
-100	0	-16	-268	0	0
-100	-8	30	-262	0	0
-100	14	-14	-257	0	0
-100	8	-15	-251	0	0
-100	0	22	-245	0	0
-100	0	-6	-240	0	0
-100	7	-15	-234	0	0
-100	0	15	-228	0	0
-100	0	0	-222	0	0
-100	0	-15	-217	0	0
-100	0	0	-211	0	0
-100	0	15	-206	0	0
-100	0	-14	-200	0	0
-100	0	-1	-194	0	0
-100	0	15	-189	0	0
-100	0	0	-185	0	0
-100	0	-7	-179	0	0
-100	0	0	-174	0	0
-100	0	15	-168	0	0
-100	0	1	-164	0	0
-100	0	-38	-158	0	0
-100	0	29	-153	0	0
-100	15	-6	-147	0	0
-100	8	-16	-141	0	0
-100	0	30	-136	0	0
-100	0	16	-133	0	0
-100	0	-30	-127	0	0
-100	0	6	-122	0	0
-100	0	8	-116	0	0
-100	7	-22	-111	0	0
-100	0	6	-105	0	0
-100	7	15	-99	0	0
-100	0	-15	-93	0	0
-100	0	0	-88	0	0
-100	0	46	-84	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
-100	29	-59	-78	0	0
-100	-22	20	-73	0	0
-100	0	24	-68	0	0
-100	15	-53	-62	0	0
-100	-7	36	-57	0	0
-100	0	1	-51	0	0
-100	15	-46	-45	0	0
-100	0	29	-39	0	0
-100	0	1	-34	0	0
-100	0	-38	-28	0	0
-100	0	21	-22	0	0
-100	0	1	-17	0	0
-100	30	-31	-12	0	0
-100	-7	22	-7	0	0
-100	0	1	-1	0	0
68	7	-22	1	0	0
100	0	0	1	0	0
-100	0	-14	-2	0	0
-13	7	6	0	0	0
100	0	46	2	0	0
-70	-7	-28	-1	0	0
-100	0	-39	-1	0	0
100	0	44	1	0	0
100	7	40	1	0	0
-100	-7	-45	-2	0	0
-51	0	-9	0	0	0
100	7	53	2	0	0
-24	0	-5	0	0	0
-100	0	-31	-1	0	0
79	7	37	1	0	0
50	15	17	0	0	0
-100	0	-53	-2	0	0
-52	15	-1	0	0	0
100	0	53	2	0	0
-32	0	-13	0	0	0
-100	-7	-47	-2	0	0
86	0	29	1	0	0
100	7	39	1	0	0
-100	0	-45	-2	0	0
-29	0	-1	0	0	0
95	0	37	1	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
97	7	25	1	0	0
-100	0	-45	-2	0	0
-36	7	-1	0	0	0
100	7	53	2	0	0
-31	7	-13	0	0	0
-100	0	-39	-1	0	0
100	15	37	1	0	0
97	7	24	1	0	0
-100	0	-45	-2	0	0
-13	0	5	0	0	0
100	0	54	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
Waktu Tempuh (detik)			20.3 detik		

Pada Tabel 4.6 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=6$, $K_i=5$, $K_d=2$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar tapi sistem kontrol mampu membuat kondisi robot kembali stabil meskipun waktu yang diperlukan untuk membuat robot kembali stabil lebih lama yaitu dengan waktu tempuh 20.3 detik.

B. Hasil pengujian untuk robot bergerak maju dengan kecepatan 150.

B.1 Pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Tabel 4. 7 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Nilai PID			Nilai Jarak (cm)		
X	Y	Z	X	Y	Z
4	0	0	0	0	0
9	150	4	0	397	0
18	150	0	0	391	0
13	150	4	0	383	0
18	150	0	0	376	0
18	150	0	0	369	0
4	150	-4	0	361	0
18	150	0	0	355	0
18	150	0	0	347	0
22	150	0	0	339	0
13	150	0	0	331	0
4	150	0	0	324	0
9	150	0	0	316	0
22	150	0	0	311	0
22	150	0	0	303	0
9	150	0	0	296	0
4	150	-4	0	288	0
18	150	0	0	280	0
22	150	0	0	273	0
13	150	0	0	267	0
13	150	0	0	259	0
22	150	0	0	252	0
13	150	0	0	244	0
4	150	-9	0	236	0
4	150	-4	0	229	0
0	150	-4	0	223	0
9	150	-4	0	215	0
18	150	-9	0	208	0
4	150	-9	0	201	0
4	150	-4	0	193	0
4	150	-4	0	186	0
9	150	-9	0	178	0
8	150	-4	0	173	0
8	150	0	0	165	0
13	150	-4	0	158	0
13	150	-4	0	151	0

Nilai PID			Nilai Jarak (cm)		
X	Y	Z	X	Y	Z
4	150	0	0	143	0
18	150	0	0	135	0
18	150	-4	0	129	0
18	150	0	0	122	0
0	150	-4	0	114	0
4	150	-8	0	107	0
4	150	-4	0	99	0
17	150	-4	0	91	0
18	150	0	0	85	0
17	150	-4	0	78	0
13	150	-4	0	70	0
22	150	0	0	62	0
17	150	0	0	54	0
22	150	0	0	47	0
17	150	0	0	41	0
18	150	-4	0	33	0
9	150	-4	0	25	0
9	150	0	0	17	0
9	150	-4	0	10	0
4	101	-9	0	2	0
13	-129	-8	0	-3	0
0	-55	0	0	-1	0
-13	22	4	0	0	0
-9	9	0	0	0	0
-9	9	0	0	0	0
Waktu Tempuh (detik)				13.5 detik	

Pada Tabel 4.7 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=5$ dan $K_d=4$. Hasil pengujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 30 cm. sehingga pada pengujian ini memiliki waktu tempuh yang lebih cepat yaitu 13.5 detik.

B.2 Pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Tabel 4. 8 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Nilai PID			Nilai Jarak (cm)		
X	Y	Z	X	Y	Z
0	0	0	0	0	0
0	150	0	0	399	0
4	150	0	0	395	0
0	150	0	0	389	0
5	150	4	0	383	0
10	150	0	0	375	0
9	150	0	0	367	0
9	150	0	0	360	0
-4	150	0	0	352	0
0	150	0	0	344	0
4	150	-4	0	336	0
4	150	-4	0	330	0
-9	150	-4	0	322	0
0	150	-4	0	314	0
-4	150	-4	0	307	0
-4	150	-9	0	299	0
-10	150	-5	0	292	0
0	150	0	0	286	0
-18	150	-4	0	279	0
-5	150	-4	0	271	0
4	150	-9	0	263	0
-4	150	-4	0	256	0
-4	150	0	0	248	0
-9	150	-4	0	242	0
4	150	-9	0	235	0
0	150	-9	0	227	0
0	150	-9	0	219	0
0	150	-9	0	211	0
4	150	-9	0	204	0
-4	150	-4	0	198	0
4	150	-4	0	191	0
24	150	0	0	183	0
14	150	-4	0	175	0
4	150	-4	0	168	0
14	150	-4	0	160	0
13	150	-4	0	152	0
14	150	-9	0	146	0

Nilai PID			Nilai Jarak (cm)		
X	Y	Z	X	Y	Z
9	150	-4	0	139	0
9	150	-5	0	131	0
9	150	-5	0	124	0
14	150	-4	0	116	0
19	150	-4	0	109	0
15	150	0	0	104	0
19	150	0	0	96	0
24	150	0	0	88	0
9	150	-4	0	81	0
15	150	-4	0	73	0
15	150	0	0	65	0
19	150	4	0	60	0
15	150	0	0	52	0
15	150	0	0	45	0
9	150	0	0	37	0
15	150	0	0	30	0
19	150	-4	0	21	0
19	150	0	0	16	0
4	150	-4	0	8	0
4	60	-9	0	1	0
14	-98	-19	0	-2	0
-13	13	0	0	0	0
-15	10	0	0	0	0
Waktu tempuh (detik)				14.3 detik	

Pada Tabel 4.8 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=9$, $K_d=4$. Hasil pngujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 50 cm. sehingga pada pengujian ini memiliki waktu tempuh yang cepat yaitu dengan waktu tempuh 14.3 detik.

B.3 Pengujian menggunakan nilai $K_p=6$, $K_i=8$, $K_d=3$

Tabel 4. 9 Tabel pengujian menggunakan nilai $K_p=6$, $K_i=8$, $K_d=3$

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
6	0	0	0	0	0
6	0	0	0	0	0
-150	0	-5	-399	0	0
-150	19	-27	-394	0	0
-150	0	5	-387	0	0
-150	0	-5	-378	0	0
-150	25	-25	-369	0	0
-150	2	-14	-360	0	0
-150	-5	-1	-356	0	0
-150	0	-12	-347	0	0
-150	0	-26	-339	0	0
-150	5	-2	-330	0	0
-150	0	0	-321	0	0
-150	0	-24	-312	0	0
-150	0	-27	-304	0	0
-150	0	0	-295	0	0
-150	11	-5	-287	0	0
-150	5	-26	-278	0	0
-150	0	-14	-269	0	0
-150	6	0	-261	0	0
-150	5	-18	-252	0	0
-150	1	-26	-243	0	0
-150	12	-13	-234	0	0
-150	6	0	-226	0	0
-150	0	-19	-217	0	0
-150	0	-13	-208	0	0
-150	0	-13	-200	0	0
-150	0	-13	-191	0	0
-150	0	-13	-182	0	0
-150	0	-13	-174	0	0
-150	6	-13	-165	0	0
-150	7	-7	-156	0	0
-150	0	-6	-148	0	0
-150	0	-12	-144	0	0
-150	5	-26	-135	0	0
-150	6	-8	-126	0	0
-150	0	-5	-118	0	0

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
-150	11	-24	-109	0	0
-150	13	-20	-100	0	0
-150	0	0	-91	0	0
-150	0	-17	-82	0	0
-150	0	-27	-73	0	0
-150	0	-7	-65	0	0
-150	0	-5	-56	0	0
-150	5	-25	-47	0	0
-150	6	-13	-39	0	0
-150	6	0	-31	0	0
-150	0	-18	-22	0	0
-150	0	-27	-13	0	0
-150	0	-1	-5	0	0
150	11	-5	3	0	0
150	74	0	4	1	0
45	-1	20	0	0	0
-150	-53	-29	-3	0	0
15	64	-16	0	1	0
150	6	68	3	0	0
29	-39	21	0	0	0
-150	-1	-82	-3	0	0
-91	11	-37	-1	0	0
66	1	13	1	0	0
137	7	40	2	0	0
-133	0	-40	-2	0	0
-146	0	-41	-2	0	0
126	0	46	2	0	0
139	0	48	2	0	0
-140	-5	-46	-2	0	0
-150	0	-48	-2	0	0
107	0	40	2	0	0
125	0	48	1	0	0
-133	0	-40	-2	0	0
-139	0	-41	-2	0	0
121	12	53	2	0	0
144	7	55	2	0	0
-127	0	-34	-2	0	0
-138	0	-41	-2	0	0
114	11	46	2	0	0
131	0	48	2	0	0
-134	0	-40	-2	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
-150	0	-52	-2	0	0
Waktu Tempuh (detik)				tidak terdefinisi	

Pada Tabel 4.9 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=6$, $K_i=8$, $K_d=3$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar dan sulit untuk sistem kontrol membuat robot kembali dalam keadaan stabil yang memiliki dampak robot tidak bisa berhenti.

C. Hasil pengujian untuk robot bergerak maju dengan kecepatan 200.

C.1 Pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Tabel 4. 10 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
4	0	0	0	400	0
0	200	8	0	398	0
-8	200	0	0	390	0
-8	200	0	0	381	0
-4	200	0	0	372	0
0	200	0	0	365	0
13	200	0	0	355	0
18	200	0	0	346	0
13	200	0	0	336	0
8	200	0	0	327	0
9	200	0	0	318	0
13	200	0	0	309	0
18	200	4	0	302	0
13	200	0	0	292	0
13	200	0	0	283	0
13	200	0	0	274	0
0	200	-4	0	264	0
0	200	-4	0	255	0
0	200	-4	0	248	0
8	200	0	0	239	0
17	200	0	0	229	0
13	200	4	0	220	0
8	200	4	0	211	0
13	200	9	0	201	0
4	200	4	0	194	0
4	200	4	0	185	0
4	200	0	0	176	0
0	200	8	0	167	0
0	200	4	0	158	0
4	200	4	0	148	0
13	200	4	0	141	0
17	200	4	0	132	0
9	200	0	0	122	0
9	200	4	0	113	0
13	200	9	0	104	0
17	200	0	0	95	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
13	200	0	0	85	0
9	200	4	0	78	0
13	200	4	0	68	0
4	200	4	0	59	0
8	200	0	0	49	0
4	200	0	0	40	0
0	200	-4	0	31	0
0	200	-4	0	24	0
-4	200	-4	0	15	0
0	200	-4	0	6	0
8	-82	0	0	-2	0
22	-91	-13	0	-2	0
0	26	4	0	0	0
Waktu Tempuh (detik)				8.02 detik.	

Pada Tabel 4.10 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=5$ dan $K_d=4$. Hasil pengujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 30 cm. sehingga pada pengujian ini memiliki waktu tempuh yang lebih cepat yaitu 8.02 detik..

C.2 Pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Tabel 4. 11 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
4	0	0	0	0	0
4	200	4	0	398	0
4	200	5	0	391	0
4	200	0	0	382	0
9	200	0	0	376	0
14	200	-4	0	366	0
10	200	0	0	357	0
9	200	0	0	348	0
4	200	0	0	339	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
0	200	0	0	330	0
4	200	4	0	321	0
9	200	0	0	314	0
5	200	4	0	305	0
14	200	9	0	296	0
9	200	0	0	287	0
14	200	4	0	278	0
10	200	0	0	269	0
4	200	0	0	262	0
0	200	0	0	253	0
-4	200	4	0	243	0
0	200	0	0	234	0
-4	200	0	0	224	0
-9	200	-4	0	215	0
-4	200	-4	0	208	0
-5	200	-4	0	199	0
-4	200	-4	0	190	0
0	200	-4	0	180	0
-4	200	-10	0	170	0
-14	200	-14	0	161	0
0	200	-4	0	154	0
0	200	-4	0	144	0
0	200	-4	0	135	0
-9	200	-4	0	126	0
0	200	-4	0	116	0
0	200	-4	0	107	0
4	200	-9	0	98	0
14	200	-9	0	91	0
5	200	-4	0	82	0
0	200	0	0	73	0
0	200	0	0	63	0
9	200	0	0	54	0
5	200	0	0	44	0
9	200	0	0	38	0
0	200	0	0	28	0
0	200	-4	0	19	0
-9	200	-4	0	9	0
-4	37	-4	0	0	0
18	-180	-28	0	-4	0
25	-53	-20	0	-1	0
-9	49	5	0	1	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
0	19	0	0	0	0
0	19	0	0	0	0
Waktu Tempuh (detik)				9.03 detik	

Pada Tabel 4.11 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=9$, $K_d=4$. Hasil pngujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 50 cm. sehingga pada pengujian ini memiliki waktu tempuh yang cepat yaitu dengan waktu tempuh 9.03 detik.

C.3 Pengujian menggunakan nilai $K_p=5$, $K_i=8$, $K_d=2$

Tabel 4. 12 Tabel pengujian menggunakan nilai $K_p=5$, $K_i=8$, $K_d=2$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
0	13	0	0	0	0
0	13	0	0	0	0
0	200	0	0	400	0
29	200	-6	0	395	0
46	200	-12	0	386	0
22	200	-19	0	376	0
-11	200	-13	0	366	0
4	200	-13	0	355	0
50	200	-6	0	345	0
34	200	-13	0	335	0
-4	200	-13	0	325	0
-2	200	-8	0	316	0
43	200	0	0	306	0
35	200	-17	0	296	0
-11	200	-32	0	285	0
-2	200	-2	0	275	0
56	200	13	0	265	0
34	200	-24	0	255	0
-10	200	-33	0	245	0
3	200	-2	0	235	0
63	200	7	1	224	0
36	200	-30	0	215	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-43	200	-28	0	204	0
14	200	17	0	195	0
71	200	3	1	185	0
-12	200	-51	0	175	0
-60	200	-41	-1	169	0
31	200	22	0	160	0
105	200	4	1	150	0
-10	200	-81	0	140	0
-80	200	-19	-1	130	0
69	200	69	1	121	0
81	200	-37	1	112	0
-39	200	-93	0	102	0
25	200	38	0	93	0
92	200	25	1	83	0
-25	200	-81	0	74	0
7	200	0	0	64	0
79	200	22	1	54	0
-18	200	-61	0	44	0
-31	200	-11	0	34	0
98	200	38	1	24	0
59	200	-39	0	15	0
-85	200	-68	-1	5	0
-7	-148	51	0	-2	0
-12	-200	99	0	-5	0
-139	-135	-31	-2	-1	0
-54	183	-90	0	3	0
52	-7	3	0	0	0
21	-164	73	0	-2	0
-125	12	-45	-2	0	0
-137	113	-99	-2	2	0
-47	10	-34	0	0	0
75	-152	100	1	-2	0
-61	-12	-8	-1	0	0
-150	123	-13	-2	2	0
-55	-79	8	0	-1	0
100	-146	134	1	-2	0
-53	25	-25	-1	0	0
-139	79	-95	-2	1	0
33	-107	63	0	-1	0
78	-78	70	1	-1	0
-108	105	-84	-1	1	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-37	-29	9	0	0	0
126	-87	91	2	-1	0
-18	74	-36	0	1	0
-162	23	-64	-2	0	0
5	-107	58	0	-1	0
89	-15	45	1	0	0
-63	102	-81	-1	1	0
-25	-49	42	0	0	0
3	-64	44	0	0	0
-86	73	-55	-1	1	0
-1	-18	7	0	0	0
30	-36	17	0	0	0
26	-26	13	0	0	0
Waktu Tempuh (detik)				16.5 detik	

Pada Tabel 4.12 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=5$, $K_i=8$, $K_d=2$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar tapi sistem kontrol mampu membuat kondisi robot kembali stabil meskipun waktu yang diperlukan untuk membuat robot kembali stabil lebih lama yaitu dengan waktu tempuh 16.5 detik.

C.4 Pengujian menggunakan nilai $K_p=6$, $K_i=5$, $K_d=3$

Tabel 4. 13 Tabel pengujian menggunakan nilai $K_p=6$, $K_i=5$, $K_d=3$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
0	0	0	0	0	0
7	200	0	0	399	0
26	200	0	0	396	0
56	200	-20	0	387	0
1	200	-21	0	378	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-34	200	0	0	368	0
32	200	7	0	358	0
49	200	-13	0	348	0
-26	200	-28	0	338	0
4	200	6	0	328	0
48	200	1	0	318	0
-11	200	-40	0	307	0
-1	200	-1	0	297	0
54	200	21	0	287	0
16	200	-26	0	277	0
-21	200	-22	0	266	0
39	200	26	0	256	0
35	200	-19	0	246	0
-26	200	-28	0	236	0
19	200	19	0	226	0
35	200	1	0	216	0
-25	200	-34	0	206	0
5	200	5	0	196	0
52	200	20	0	190	0
23	200	-19	0	180	0
-41	200	-28	0	170	0
45	200	39	0	160	0
71	200	-17	1	150	0
-38	200	-70	0	140	0
10	200	38	0	131	0
69	200	9	1	121	0
-10	200	-54	0	111	0
-36	200	-9	0	101	0
101	200	47	1	90	0
32	200	-38	0	80	0
-41	200	-23	0	70	0
39	200	47	0	60	0
22	200	-18	0	50	0
-27	200	-35	0	39	0
45	200	19	0	29	0
51	200	-11	0	19	0
-53	200	-49	0	9	0
17	-6	32	0	0	0
77	-200	-18	1	-5	0
35	-200	-131	0	-3	0
96	101	46	1	1	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
132	6	80	2	0	0
-14	-149	-127	0	-2	0
-79	-127	-126	-1	-1	0
70	95	62	1	1	0
187	14	80	2	0	0
-51	-42	-94	0	0	0
-163	59	-17	-2	0	0
67	-9	0	1	0	0
101	-2	34	1	0	0
-125	14	-19	-2	0	0
-14	-6	-7	0	0	0
151	13	41	2	0	0
-6	-13	-4	0	0	0
-79	0	-21	-1	0	0
79	7	26	1	0	0
-9	0	-5	0	0	0
-43	0	-14	0	0	0
46	7	19	0	0	0
2	0	0	0	0	0
-21	0	-7	0	0	0
0	-6	0	0	0	0
0	-7	0	0	0	0
Waktu Tempuh (detik)				17.2 detik	

Pada Tabel 4.13 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=6$, $K_i=5$, $K_d=3$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar tapi sistem kontrol mampu membuat kondisi robot kembali stabil meskipun waktu yang diperlukan untuk membuat robot kembali stabil lebih lama yaitu dengan waktu tempuh 17.2 detik.

D. Hasil pengujian untuk robot bergerak kanan dengan kecepatan 100.

D.1 Pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Tabel 4. 14 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
-100	0	0	-350	0	0
-100	0	0	-349	0	0
-100	0	0	-348	0	0
-100	0	-4	-345	0	0
-100	0	0	-341	0	0
-100	-4	0	-337	0	0
-100	0	0	-332	0	0
-100	-4	0	-327	0	0
-100	-4	-4	-322	0	0
-100	-9	-4	-318	0	0
-100	-4	0	-313	0	0
-100	0	-4	-308	0	0
-100	0	0	-303	0	0
-100	0	0	-298	0	0
-100	0	0	-293	0	0
-100	0	0	-288	0	0
-100	-4	0	-284	0	0
-100	-4	-4	-278	0	0
-100	0	0	-273	0	0
-100	0	0	-268	0	0
-100	0	0	-263	0	0
-100	-4	0	-258	0	0
-100	-9	0	-254	0	0
-100	0	0	-249	0	0
-100	0	-4	-243	0	0
-100	-13	0	-238	0	0
-100	0	-4	-233	0	0
-100	0	0	-228	0	0
-100	0	0	-224	0	0
-100	0	-9	-219	0	0
-100	-8	-4	-213	0	0
-100	-9	-4	-208	0	0
-100	-9	-4	-203	0	0
-100	0	0	-197	0	0
-100	-9	0	-194	0	0
-100	0	0	-188	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
-100	0	-4	-183	0	0
-100	0	0	-178	0	0
-100	0	4	-173	0	0
-100	0	0	-168	0	0
-100	0	-4	-163	0	0
-100	-4	0	-159	0	0
-100	0	0	-154	0	0
-100	4	0	-149	0	0
-100	4	0	-143	0	0
-100	0	0	-138	0	0
-100	0	4	-133	0	0
-100	4	0	-129	0	0
-100	9	4	-124	0	0
-100	4	-4	-119	0	0
-100	0	0	-114	0	0
-100	-4	-4	-109	0	0
-100	-4	-4	-104	0	0
-100	0	-4	-100	0	0
-100	0	0	-95	0	0
-100	9	-9	-90	0	0
-100	4	0	-85	0	0
-100	0	0	-80	0	0
-100	0	-4	-75	0	0
-100	9	-4	-71	0	0
-100	4	0	-66	0	0
-100	0	-4	-61	0	0
-100	0	-4	-56	0	0
-100	-8	4	-50	0	0
-100	0	0	-45	0	0
-100	0	-4	-40	0	0
-100	-9	-4	-36	0	0
-100	0	0	-31	0	0
-100	-8	0	-26	0	0
-100	-4	0	-21	0	0
-100	-8	0	-16	0	0
-100	0	-4	-11	0	0
-100	-4	-4	-7	0	0
-100	0	-4	-3	0	0
43	0	0	1	0	0
18	0	-4	0	0	0
22	-4	0	0	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
Waktu tempuh (detik)			17.02 detik		

Pada Tabel 4.14 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=5$ dan $K_d=4$. Hasil pengujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 30 cm. sehingga pada pengujian ini memiliki waktu tempuh yang lebih cepat yaitu 17.02 detik.

D.2 Pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Tabel 4. 15 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
0	0	0	0	0	0
-100	0	0	-349	0	0
-100	0	0	-346	0	0
-100	0	0	-343	0	0
-100	0	-4	-339	0	0
-100	0	0	-334	0	0
-100	14	-4	-329	0	0
-100	0	0	-325	0	0
-100	0	0	-320	0	0
-100	-4	0	-316	0	0
-100	-4	-4	-311	0	0
-100	0	0	-307	0	0
-100	-4	0	-302	0	0
-100	-1	0	-297	0	0
-100	0	0	-292	0	0
-100	-5	-4	-289	0	0
-100	0	0	-284	0	0
-100	-5	0	-279	0	0
-100	0	0	-274	0	0
-100	0	0	-269	0	0
-100	-1	-4	-264	0	0
-100	0	-9	-260	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
-100	-4	0	-255	0	0
-100	0	0	-250	0	0
-100	-4	0	-245	0	0
-100	-4	0	-240	0	0
-100	-9	0	-235	0	0
-100	0	-4	-230	0	0
-100	0	0	-227	0	0
-100	0	0	-222	0	0
-100	0	-4	-217	0	0
-100	-5	0	-212	0	0
-100	9	-4	-207	0	0
-100	0	0	-203	0	0
-100	0	4	-199	0	0
-100	0	-9	-194	0	0
-100	0	0	-189	0	0
-100	0	0	-184	0	0
-100	4	0	-179	0	0
-100	4	0	-174	0	0
-100	10	0	-170	0	0
-100	0	0	-165	0	0
-100	0	-4	-160	0	0
-100	0	-4	-154	0	0
-100	0	0	-149	0	0
-100	-4	0	-144	0	0
-100	-1	-4	-141	0	0
-100	0	0	-136	0	0
-100	0	-4	-131	0	0
-100	0	-4	-126	0	0
-100	-4	0	-121	0	0
-100	-4	0	-116	0	0
-100	0	-4	-111	0	0
-100	0	-9	-107	0	0
-100	-4	0	-102	0	0
-100	0	-4	-97	0	0
-100	-4	0	-92	0	0
-100	-4	0	-87	0	0
-100	-9	-4	-82	0	0
-100	0	-5	-78	0	0
-100	0	-4	-73	0	0
-100	4	0	-68	0	0
-100	4	4	-63	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
-100	0	-4	-58	0	0
-100	-1	0	-53	0	0
-100	0	-9	-49	0	0
-100	0	0	-44	0	0
-100	0	0	-39	0	0
-100	0	-4	-34	0	0
-100	0	-4	-29	0	0
-100	9	0	-24	0	0
-100	0	-4	-21	0	0
-100	4	-4	-16	0	0
-100	4	0	-11	0	0
-100	0	0	-6	0	0
-73	0	-4	-1	0	0
68	0	0	1	0	0
5	0	-9	0	0	0
Waktu Tempuh (detik)				17.08 detik	

Pada Tabel 4.15 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=9$, $K_d=4$. Hasil pengujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 50 cm. sehingga pada pengujian ini memiliki waktu tempuh yang cepat yaitu dengan waktu tempuh 17.08 detik.

D.3 Pengujian menggunakan nilai $K_p=6$, $K_i=3$, $K_d=6$

Tabel 4. 16 Tabel pengujian menggunakan nilai $K_p=6$, $K_i=3$, $K_d=6$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
7	0	0	0	0	0
-100	0	0	-399	0	0
-100	22	0	-397	0	0
-100	7	0	-392	0	0
-100	0	7	-387	0	0
-100	7	0	-381	0	0
-100	7	-7	-376	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-100	7	14	-370	0	0
-100	7	0	-364	0	0
-100	23	-15	-358	0	0
-100	-7	0	-352	0	0
-100	0	8	-346	0	0
-100	7	-14	-340	0	0
-100	0	6	-334	0	0
-100	0	15	-328	0	0
-100	-7	-30	-322	0	0
-100	0	6	-317	0	0
-100	0	24	-311	0	0
-100	7	-44	-305	0	0
-100	-14	21	-300	0	0
-100	0	9	-294	0	0
-100	0	-45	-288	0	0
-100	0	14	-282	0	0
-100	0	16	-277	0	0
-100	7	-14	-274	0	0
-100	0	-16	-268	0	0
-100	-8	30	-262	0	0
-100	14	-14	-257	0	0
-100	8	-15	-251	0	0
-100	0	22	-245	0	0
-100	0	-6	-240	0	0
-100	7	-15	-234	0	0
-100	0	15	-228	0	0
-100	0	0	-222	0	0
-100	0	-15	-217	0	0
-100	0	0	-211	0	0
-100	0	15	-206	0	0
-100	0	-14	-200	0	0
-100	0	-1	-194	0	0
-100	0	15	-189	0	0
-100	0	0	-185	0	0
-100	0	-7	-179	0	0
-100	0	0	-174	0	0
-100	0	15	-168	0	0
-100	0	1	-164	0	0
-100	0	-38	-158	0	0
-100	0	29	-153	0	0
-100	15	-6	-147	0	0

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
-100	8	-16	-141	0	0
-100	0	30	-136	0	0
-100	0	16	-133	0	0
-100	0	-30	-127	0	0
-100	0	6	-122	0	0
-100	0	8	-116	0	0
-100	7	-22	-111	0	0
-100	0	6	-105	0	0
-100	7	15	-99	0	0
-100	0	-15	-93	0	0
-100	0	0	-88	0	0
-100	0	46	-84	0	0
-100	29	-59	-78	0	0
-100	-22	20	-73	0	0
-100	0	24	-68	0	0
-100	15	-53	-62	0	0
-100	-7	36	-57	0	0
-100	0	1	-51	0	0
-100	15	-46	-45	0	0
-100	0	29	-39	0	0
-100	0	1	-34	0	0
-100	0	-38	-28	0	0
-100	0	21	-22	0	0
-100	0	1	-17	0	0
-100	30	-31	-12	0	0
-100	-7	22	-7	0	0
-100	0	1	-1	0	0
68	7	-22	1	0	0
100	0	0	1	0	0
-100	0	-14	-2	0	0
-13	7	6	0	0	0
100	0	46	2	0	0
-70	-7	-28	-1	0	0
-100	0	-39	-1	0	0
100	0	44	1	0	0
100	7	40	1	0	0
-100	-7	-45	-2	0	0
-51	0	-9	0	0	0
100	7	53	2	0	0
-24	0	-5	0	0	0
-100	0	-31	-1	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
79	7	37	1	0	0
50	15	17	0	0	0
-100	0	-53	-2	0	0
-52	15	-1	0	0	0
100	0	53	2	0	0
-32	0	-13	0	0	0
-100	-7	-47	-2	0	0
86	0	29	1	0	0
100	7	39	1	0	0
-100	0	-45	-2	0	0
-29	0	-1	0	0	0
95	0	37	1	0	0
97	7	25	1	0	0
-100	0	-45	-2	0	0
-36	7	-1	0	0	0
100	7	53	2	0	0
-31	7	-13	0	0	0
-100	0	-39	-1	0	0
100	15	37	1	0	0
97	7	24	1	0	0
-100	0	-45	-2	0	0
-13	0	5	0	0	0
100	0	54	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
100	0	62	2	0	0
Waktu Tempuh (detik)				22.03 detik	

Pada Tabel 4.16 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=6$, $K_i=3$, $K_d=6$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar tapi sistem kontrol mampu membuat kondisi robot kembali stabil meskipun waktu

yang diperlukan untuk membuat robot kembali stabil lebih lama yaitu dengan waktu tempuh 22.03 detik.

D.4 Pengujian menggunakan nilai $K_p=8$, $K_i=1$, $K_d=8$

Tabel 4. 17 Tabel pengujian menggunakan nilai $K_p=8$, $K_i=1$, $K_d=8$

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
9	0	0	0	0	0
-100	0	-9	-399	0	0
-100	0	9	-395	0	0
-100	0	9	-393	0	0
-100	0	-9	-387	0	0
-100	0	0	-381	0	0
-100	0	18	-376	0	0
-100	0	-18	-370	0	0
-100	0	0	-364	0	0
-100	-9	18	-358	0	0
-100	0	-28	-352	0	0
-100	0	18	-346	0	0
-100	0	0	-340	0	0
-100	0	-37	-334	0	0
-100	0	55	-328	0	0
-100	0	-36	-323	0	0
-100	0	-1	-317	0	0
-100	0	37	-312	0	0
-100	27	-83	-306	0	0
-100	0	64	-301	0	0
-100	0	-26	-295	0	0
-100	27	-29	-289	0	0
-100	-18	93	-284	0	0
-100	9	-100	-278	0	0
-100	0	44	-273	0	0
-100	0	20	-267	0	0
-100	46	-57	-261	0	0
-100	-36	83	-256	0	0
-100	-28	10	-252	0	0
-100	18	-38	-246	0	0
-100	0	65	-241	0	0
-100	27	-82	-234	0	0

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
-100	0	45	-229	0	0
-100	0	1	-223	0	0
-100	27	-29	-217	0	0
-100	9	65	-212	0	0
-100	18	-83	-205	0	0
-100	0	45	-200	0	0
-100	0	-7	-194	0	0
-100	9	-29	-189	0	0
-100	0	47	-183	0	0
-100	0	-73	-176	0	0
-100	0	54	-171	0	0
-100	-18	-8	-165	0	0
-100	9	-20	-159	0	0
-100	0	56	-154	0	0
-100	0	-73	-148	0	0
-100	0	54	-142	0	0
-100	0	1	-137	0	0
-100	0	-29	-131	0	0
-100	9	65	-125	0	0
-100	9	-82	-119	0	0
-100	9	54	-114	0	0
-100	-18	-7	-108	0	0
-100	0	-83	-104	0	0
-100	0	36	-99	0	0
-100	27	10	-93	0	0
-100	-8	-56	-87	0	0
-100	-19	74	-81	0	0
-100	37	-54	-75	0	0
-100	-36	17	-70	0	0
-100	0	29	-65	0	0
-100	-18	-56	-60	0	0
-100	-37	64	-55	0	0
-100	55	-45	-49	0	0
-100	-73	-1	-44	0	0
-100	17	47	-39	0	0
-100	19	-65	-33	0	0
-100	-100	55	-28	-1	0
-100	35	38	-23	0	0
-100	57	-100	-18	0	0
-100	-100	100	-14	-1	0
-100	-57	20	-9	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-100	93	-100	-4	1	0
-3	-100	82	0	-1	0
100	17	1	1	0	0
-100	48	-65	-1	0	0
-21	-100	55	0	-1	0
100	100	0	1	1	0
30	66	-18	0	0	0
-100	-83	-18	-1	0	0
100	55	27	1	0	0
21	-9	10	0	0	0
-100	-18	-65	-2	0	0
-12	9	17	0	0	0
85	0	19	0	0	0
-100	0	-46	-1	0	0
-100	0	-47	-1	0	0
-100	0	-47	-1	0	0
Waktu Tempuh (detik)				23.02 detik	

Pada Tabel 4.17 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=8$, $K_i=1$, $K_d=8$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar tapi sistem kontrol mampu membuat kondisi robot kembali stabil meskipun waktu yang diperlukan untuk membuat robot kembali stabil lebih lama yaitu dengan waktu tempuh 23.02 detik.

E. Hasil pengujian untuk robot bergerak ke kanan dengan kecepatan 150.

E.1 Pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Tabel 4. 18 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
0	0	0	0	0	0
0	0	0	0	0	0
-150	0	0	-349	0	0
-150	4	-4	-346	0	0
-150	13	-13	-339	0	0
-150	0	-13	-334	0	0
-150	0	-13	-326	0	0
-150	-4	-9	-319	0	0
-150	0	-13	-312	0	0
-150	0	-9	-304	0	0
-150	-13	-9	-296	0	0
-150	-4	-18	-291	0	0
-150	0	-13	-283	0	0
-150	0	-13	-276	0	0
-150	-8	-9	-268	0	0
-150	0	-9	-260	0	0
-150	-9	-13	-252	0	0
-150	-4	-13	-247	0	0
-150	-4	-9	-239	0	0
-150	-9	-13	-231	0	0
-150	-13	-18	-223	0	0
-150	0	-13	-216	0	0
-150	-8	-9	-208	0	0
-150	0	-9	-202	0	0
-150	0	-8	-195	0	0
-150	0	-9	-187	0	0
-150	-4	-18	-179	0	0
-150	-9	-13	-172	0	0
-150	0	-8	-164	0	0
-150	0	-13	-157	0	0
-150	0	-9	-151	0	0
-150	0	-13	-143	0	0
-150	-4	-13	-135	0	0
-150	-13	-13	-128	0	0
-150	-4	-13	-120	0	0
-150	0	-9	-112	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
-150	-9	-13	-107	0	0
-150	0	-18	-99	0	0
-150	0	-9	-92	0	0
-150	0	-9	-84	0	0
-150	-8	-17	-77	0	0
-150	-4	-13	-69	0	0
-150	-9	-13	-63	0	0
-150	-4	-9	-56	0	0
-150	-4	-13	-48	0	0
-150	0	-9	-41	0	0
-150	0	-13	-33	0	0
-150	-4	-13	-25	0	0
-150	0	-13	-20	0	0
-150	0	-4	-12	0	0
-150	0	-13	-4	0	0
87	4	-9	2	0	0
37	22	-9	0	0	0
Waktu Tempuh (detik)				12.08 detik	

Pada Tabel 4.18 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=5$ dan $K_d=4$. Hasil pengujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 30 cm. sehingga pada pengujian ini memiliki waktu tempuh yang lebih cepat yaitu 12.08 detik.

E.2 Pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Tabel 4. 19 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	0	-349	0	0
-150	0	-4	-348	0	0
-150	0	-9	-343	0	0
-150	0	-9	-337	0	0
-150	-9	-14	-329	0	0
-150	-5	-9	-321	0	0
-150	0	-14	-314	0	0
-150	-9	-14	-306	0	0
-150	-4	-9	-300	0	0
-150	0	-9	-293	0	0
-150	-4	-14	-285	0	0
-150	0	-19	-278	0	0
-150	-4	-9	-270	0	0
-150	-15	-14	-263	0	0
-150	-10	-10	-257	0	0
-150	0	-9	-249	0	0
-150	0	-14	-241	0	0
-150	0	-9	-234	0	0
-150	-10	-14	-226	0	0
-150	-9	-9	-219	0	0
-150	0	-14	-211	0	0
-150	0	-9	-205	0	0
-150	0	-10	-197	0	0
-150	4	-9	-189	0	0
-150	4	-19	-182	0	0
-150	0	-9	-174	0	0
-150	-4	-9	-166	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
-150	-9	-14	-160	0	0
-150	0	-14	-152	0	0
-150	0	-9	-144	0	0
-150	-5	-14	-136	0	0
-150	-10	-10	-128	0	0
-150	0	-9	-120	0	0
-150	0	-14	-115	0	0
-150	0	-14	-107	0	0
-150	-5	-14	-100	0	0
-150	-4	-9	-92	0	0
-150	0	-14	-84	0	0
-150	0	-14	-76	0	0
-150	0	-9	-70	0	0
-150	0	-9	-63	0	0
-150	0	-14	-55	0	0
-150	-5	-14	-47	0	0
-150	-4	-5	-39	0	0
-150	0	-14	-31	0	0
-150	0	-14	-23	0	0
-150	-9	-14	-18	0	0
-150	-10	-14	-10	0	0
-128	0	-9	-2	0	0
107	18	-10	2	0	0
7	15	-9	0	0	0
Waktu Tempuh (detik)			13.05 detik		

Pada Tabel 4.19 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=9$, $K_d=4$. Hasil pngujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 50 cm. sehingga pada pengujian ini memiliki waktu tempuh yang cepat yaitu dengan waktu tempuh 13.05 detik.

E.3 Pengujian menggunakan nilai $K_p=7$, $K_i=8$, $K_d=2$

Tabel 4. 20 Tabel pengujian menggunakan nilai $K_p=7$, $K_i=8$, $K_d=2$

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
8	0	0	0	0	0
8	0	0	0	0	0
-150	0	0	-399	0	0
-150	16	-32	-397	0	0
-150	1	6	-391	0	0
-150	0	0	-382	0	0
-150	0	-40	-373	0	0
-150	0	-10	-364	0	0
-150	-7	0	-356	0	0
-150	0	-32	-347	0	0
-150	0	-27	-338	0	0
-150	-7	15	-329	0	0
-150	-1	-22	-321	0	0
-150	17	-35	-312	0	0
-150	-23	7	-304	0	0
-150	-2	-14	-295	0	0
-150	32	-50	-286	0	0
-150	3	-4	-282	0	0
-150	-52	26	-274	0	0
-150	54	-56	-265	0	0
-150	-26	3	-256	0	0
-150	-55	10	-248	0	0
-150	104	-49	-239	1	0
-150	-57	4	-231	0	0
-150	-115	49	-223	-1	0
-150	85	15	-216	1	0
-150	91	-109	-207	1	0
-150	-150	107	-201	-3	0
-150	-150	102	-194	-2	0
-150	85	-105	-187	1	0
-150	150	-15	-182	2	0
-150	-150	109	-176	-2	0
-150	-150	-44	-168	-2	0
-150	150	-13	-161	2	0
-150	59	10	-154	0	0
-150	-150	-25	-146	-3	0
-150	-115	46	-139	-1	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-150	150	-20	-131	3	0
-150	51	-87	-125	0	0
-150	-150	133	-120	-4	0
-150	-150	136	-114	-4	0
-150	-126	-150	-108	-1	0
-150	75	-147	-104	1	0
-150	150	150	-99	2	0
-150	82	137	-93	0	0
-150	-31	-150	-85	0	0
-150	-10	1	-78	0	0
-150	15	122	-71	0	0
-150	43	-142	-63	0	0
-150	-18	20	-55	0	0
-150	-150	114	-48	-1	0
-150	7	-93	-40	0	0
-150	32	25	-33	0	0
-150	-101	38	-25	-1	0
-150	50	-82	-16	0	0
-150	9	46	-10	0	0
-150	-93	-2	-2	-1	0
150	132	-13	4	1	0
150	12	49	3	0	0
-82	-111	-54	-1	-1	0
-150	41	-57	-2	0	0
104	57	72	1	0	0
150	-126	-18	2	-1	0
-150	32	-79	-2	0	0
-150	72	39	-1	0	0
150	-35	13	2	0	0
Waktu Tempuh (detik)				tidak terdefinisi	

Pada Tabel 4.20 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=7$, $K_i=8$, $K_d=2$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar dan sulit untuk sistem kontrol membuat robot kembali dalam keadaan stabil yang

memiliki dampak robot tidak bisa berhenti.

E.4 Pengujian menggunakan nilai $K_p=5$, $K_i=8$, $K_d=7$

Tabel 4. 21 Tabel pengujian menggunakan nilai $K_p=5$, $K_i=8$, $K_d=7$

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
0	0	9	0	0	0
0	0	9	0	0	0
-150	0	-37	-399	0	0
-150	0	-11	-394	0	0
-150	0	29	-386	0	0
-150	28	-75	-377	0	0
-150	-65	27	-369	0	0
-150	-11	20	-360	0	0
-150	76	-76	-351	0	0
-150	-150	65	-344	-1	0
-150	-60	67	-337	0	0
-150	133	-84	-329	1	0
-150	-150	36	-323	-2	0
-150	-150	134	-317	-2	0
-150	33	-55	-310	0	0
-150	107	-20	-304	1	0
-150	-150	85	-299	-3	0
-150	-150	1	-292	-3	0
-150	138	-19	-284	1	0
-150	24	-56	-278	0	0
-150	-150	83	-272	-3	0
-150	-150	135	-266	-2	0
-150	-137	-6	-263	-1	0
-150	140	-40	-256	1	0
-150	-110	123	-249	-1	0
-150	-107	-74	-241	-1	0
-150	150	7	-234	2	0
-150	-91	39	-226	-1	0
-150	-59	-142	-216	0	0
-150	104	102	-208	1	0
-150	1	12	-200	0	0
-150	0	-106	-191	0	0
-150	-9	150	-184	0	0
-150	56	-82	-176	0	0
-150	-17	-51	-168	0	0
-150	-124	150	-162	-1	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-150	-2	-82	-154	0	0
-150	58	25	-148	0	0
-150	-132	68	-141	-1	0
-150	63	-132	-131	0	0
-150	79	83	-124	0	0
-150	-133	-6	-116	-1	0
-150	83	-59	-107	0	0
-150	-7	123	-100	0	0
-150	-19	-121	-92	0	0
-150	19	44	-84	0	0
-150	-47	78	-76	0	0
-150	-1	-63	-72	0	0
-150	76	-79	-63	0	0
-150	-150	150	-57	-2	0
-150	-127	-34	-49	-1	0
-150	150	-40	-42	2	0
-150	-72	123	-36	0	0
-150	-88	-92	-28	0	0
-150	141	7	-19	1	0
-150	-140	58	-12	-1	0
-150	-3	-104	-3	0	0
150	40	130	3	0	0
150	-150	-6	3	-1	0
-62	16	-150	0	0	0
-97	142	150	-1	1	0
150	-5	-23	1	0	0
-15	-134	-150	0	-1	0
-67	26	150	0	0	0
94	125	-79	1	1	0
-17	57	-150	0	0	0
-39	0	150	0	0	0
105	0	21	1	0	0
105	0	19	1	0	0
Waktu Tempuh (detik)				19.08 detik	

Pada Tabel 4.21 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=6$, $K_i=5$, $K_d=2$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot

mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar tapi sistem kontrol mampu membuat kondisi robot kembali stabil meskipun waktu yang diperlukan untuk membuat robot kembali stabil lebih lama yaitu dengan waktu tempuh 19.08 detik.

F. Hasil pengujian untuk robot bergerak kanan dengan kecepatan 200.

F.1 Pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Tabel 4. 22 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
0	0	0	0	0	0
-200	-18	4	-349	0	0
-200	4	-12	-347	0	0
-200	18	-22	-341	0	0
-200	-4	-18	-332	0	0
-200	-13	-22	-323	0	0
-200	4	-27	-314	0	0
-200	9	-27	-305	0	0
-200	0	-22	-298	0	0
-200	0	-26	-289	0	0
-200	0	-22	-280	0	0
-200	-4	-22	-270	0	0
-200	0	-22	-261	0	0
-200	0	-27	-252	0	0
-200	0	-27	-245	0	0
-200	-4	-18	-236	0	0
-200	-4	-18	-227	0	0
-200	-4	-27	-218	0	0
-200	-4	-22	-208	0	0
-200	-4	-26	-199	0	0
-200	0	-27	-189	0	0
-200	-9	-22	-183	0	0
-200	-4	-26	-173	0	0
-200	0	-26	-164	0	0
-200	0	-27	-155	0	0
-200	0	-27	-146	0	0
-200	0	-18	-137	0	0

-200	0	-26	-130	0	0
-200	0	-27	-121	0	0
-200	0	-22	-112	0	0
-200	4	-22	-102	0	0
-200	0	-27	-93	0	0
-200	4	-22	-84	0	0
-200	-9	-18	-77	0	0
-200	-4	-22	-67	0	0
-200	0	-27	-58	0	0
-200	0	-27	-49	0	0
-200	4	-26	-39	0	0
-200	0	-27	-30	0	0
-200	0	-22	-23	0	0
-200	0	-22	-14	0	0
-200	0	-31	-5	0	0
92	0	-5	2	0	0
46	9	-4	1	0	0
-18	0	-4	0	0	0
Waktu Tempuh (detik)				08.04 detik	

Pada Tabel 4.21 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=5$ dan $K_d=4$. Hasil pengujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 30 cm. sehingga pada pengujian ini memiliki waktu tempuh yang lebih cepat yaitu 08.04 detik.

F.2 Pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Tabel 4. 23 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Nilai PID			Nilai Pengukuran		
X	Y	Z	X	Y	Z
-200	0	0	-349	0	0
-200	19	-14	-344	0	0
-200	0	-19	-336	0	0
-200	-4	-24	-326	0	0
-200	0	-19	-317	0	0
-200	-4	-19	-308	0	0

Nilai PID			Nilai Pengukuran		
X	Y	Z	X	Y	Z
-200	0	-29	-298	0	0
-200	0	-29	-291	0	0
-200	0	-19	-282	0	0
-200	0	-19	-273	0	0
-200	4	-29	-263	0	0
-200	0	-24	-254	0	0
-200	0	-24	-245	0	0
-200	0	-29	-238	0	0
-200	-4	-24	-228	0	0
-200	-15	-19	-219	0	0
-200	-15	-24	-210	0	0
-200	0	-29	-201	0	0
-200	5	-24	-192	0	0
-200	0	-19	-185	0	0
-200	0	-29	-175	0	0
-200	0	-29	-166	0	0
-200	0	-24	-157	0	0
-200	-14	-19	-147	0	0
-200	-9	-29	-138	0	0
-200	0	-24	-131	0	0
-200	-5	-24	-122	0	0
-200	0	-19	-112	0	0
-200	0	-29	-103	0	0
-200	-9	-29	-94	0	0
-200	-4	-19	-84	0	0
-200	0	-24	-75	0	0
-200	0	-24	-68	0	0
-200	-4	-24	-59	0	0
-200	0	-24	-50	0	0
-200	0	-24	-40	0	0
-200	0	-24	-31	0	0
-200	-4	-24	-21	0	0
-200	0	-24	-14	0	0
-200	0	-24	-5	0	0
118	0	-6	2	0	0
57	14	4	1	0	0
-44	9	0	0	0	0
-19	4	4	0	0	0
Waktu Tempuh (Detik)				08.06 detik	

Pada Tabel 4.22 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=9$, $K_d=4$. Hasil pngujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 50 cm. sehingga pada pengujian ini memiliki waktu tempuh yang cepat yaitu dengan waktu tempuh 08.06 detik.

F.3 Pengujian menggunakan nilai $K_p=5$, $K_i=8$, $K_d=7$

Tabel 4. 24 Tabel pengujian menggunakan nilai $K_p=5$, $K_i=8$, $K_d=7$

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
0	-6	0	0	0	0
0	-6	0	0	0	0
-200	-6	0	-399	0	0
-200	35	-49	-396	0	0
-200	10	-17	-388	0	0
-200	-38	7	-378	0	0
-200	0	-43	-367	0	0
-200	32	-53	-356	0	0
-200	-11	-15	-345	0	0
-200	-14	-19	-335	0	0
-200	5	-38	-324	0	0
-200	13	-33	-314	0	0
-200	-11	-26	-303	0	0
-200	-7	-26	-293	0	0
-200	11	-33	-282	0	0
-200	0	-26	-272	0	0
-200	0	-32	-261	0	0
-200	0	-33	-251	0	0
-200	0	-33	-240	0	0
-200	0	-27	-229	0	0
-200	0	-25	-219	0	0
-200	-6	-33	-208	0	0
-200	-6	-33	-203	0	0
-200	0	-27	-193	0	0
-200	0	-25	-182	0	0
-200	0	-38	-172	0	0
-200	0	-27	-161	0	0

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
-200	-12	-26	-151	0	0
-200	-6	-26	-140	0	0
-200	0	-32	-130	0	0
-200	5	-27	-120	0	0
-200	0	-26	-109	0	0
-200	0	-38	-98	0	0
-200	0	-27	-88	0	0
-200	0	-25	-78	0	0
-200	0	-26	-67	0	0
-200	0	-26	-57	0	0
-200	0	-26	-46	0	0
-200	-7	-33	-36	0	0
-200	-1	-26	-25	0	0
-200	0	-25	-15	0	0
-200	0	-33	-5	0	0
200	0	13	4	0	0
200	0	129	6	0	0
153	0	-3	2	0	0
-200	0	-46	-5	0	0
-200	12	-45	-4	0	0
114	2	8	2	0	0
200	-17	54	4	0	0
87	-1	49	1	0	0
-200	-17	-60	-3	0	0
-134	-2	-63	-2	0	0
157	0	35	2	0	0
10	0	4	0	0	0
-200	-5	-82	-3	0	0
-92	0	-31	-1	0	0
200	6	74	3	0	0
87	6	31	1	0	0
-200	-5	-74	-3	0	0
-98	5	-37	-1	0	0
185	12	61	3	0	0
72	0	30	0	0	0
-187	-13	-68	-3	0	0
-41	0	-18	0	0	0
200	0	68	3	0	0
35	-5	12	0	0	0
-200	-19	-82	-4	0	0
-80	5	-31	-1	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
200	6	86	4	0	0
145	0	51	2	0	0
-181	-11	-60	-3	0	0
-59	0	-17	0	0	0
200	5	68	3	0	0
164	0	60	2	0	0
-138	-5	-52	-2	0	0
-200	-6	-80	-3	0	0
-145	0	-39	-2	0	0
-145	0	-39	-2	0	0
-145	0	-39	-2	0	0
Waktu Tempuh (detik)				17.3 detik.	

Pada Tabel 4.24 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=5$, $K_i=8$, $K_d=7$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar tapi sistem kontrol mampu membuat kondisi robot kembali stabil meskipun waktu yang diperlukan untuk membuat robot kembali stabil lebih lama yaitu dengan waktu tempuh 17.3 detik.

F.4 Pengujian menggunakan nilai $K_p=7$, $K_i=8$, $K_d=2$

Tabel 4. 25 Tabel pengujian menggunakan nilai $K_p=7$, $K_i=8$, $K_d=2$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
8	0	0	0	0	0
8	0	0	0	0	0
-200	39	7	-398	0	0
-200	60	-32	-392	0	0
-200	-46	-1	-382	0	0
-200	-3	-32	-371	0	0
-200	68	-43	-361	0	0
-200	-93	-1	-351	-1	0
-200	-25	-32	-340	0	0

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
-200	86	-19	-330	1	0
-200	-121	17	-321	-1	0
-200	24	-58	-311	0	0
-200	107	-2	-301	1	0
-200	-200	1	-292	-2	0
-200	-23	-19	-282	0	0
-200	179	1	-273	2	0
-200	-200	-43	-265	-3	0
-200	-200	3	-260	-4	0
-200	-165	151	-252	-1	0
-200	125	-40	-244	1	0
-200	23	-121	-236	0	0
-200	-200	149	-231	-3	0
-200	-121	104	-223	-1	0
-200	101	-136	-215	1	0
-200	46	-51	-206	0	0
-200	-193	67	-198	-2	0
-200	-3	62	-189	0	0
-200	147	-104	-180	1	0
-200	-200	-24	-172	-2	0
-200	-200	163	-165	-2	0
-200	-3	21	-157	0	0
-200	153	-151	-148	1	0
-200	-179	24	-141	-2	0
-200	-177	169	-134	-2	0
-200	41	-55	-126	0	0
-200	162	-156	-117	2	0
-200	-200	75	-110	-3	0
-200	-200	89	-102	-2	0
-200	111	-46	-94	1	0
-200	75	-86	-86	0	0
-200	-200	117	-80	-4	0
-200	-200	168	-73	-4	0
-200	-200	22	-69	-3	0
-200	149	-61	-60	2	0
-200	78	7	-52	0	0
-200	-200	-23	-44	-4	0
-115	-129	-104	-1	-1	0
-200	160	-129	-3	2	0
-124	67	137	-1	0	0
200	4	-111	2	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
38	46	-179	0	0	0
-118	117	200	-1	1	0
36	157	-46	0	1	0
59	111	-200	0	1	0
60	111	-200	0	1	0
Waktu Tempuh (detik)				19.05 detik.	

Pada Tabel 4.25 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=7$, $K_i=8$, $K_d=2$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar tapi sistem kontrol mampu membuat kondisi robot kembali stabil meskipun waktu yang diperlukan untuk membuat robot kembali stabil lebih lama yaitu dengan waktu tempuh 19.05 detik.

G. Hasil pengujian untuk robot bergerak diagonal dengan kecepatan 100.

G.1 Pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Tabel 4. 26 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Nilai PID			Nilai Pengukuran		
X	Y	Z	X	Y	Z
0	0	0	0	0	0
-100	100	4	-248	248	0
-100	100	0	-246	245	0
-100	100	0	-242	240	0
-100	100	-4	-237	235	0
-100	100	0	-233	229	0
-100	100	-4	-228	224	0
-100	100	0	-223	218	0
-100	100	0	-220	214	0
-100	100	0	-215	208	0
-100	100	4	-210	203	0
-100	100	0	-204	197	0

Nilai PID			Nilai Pengukuran		
X	Y	Z	X	Y	Z
-100	100	0	-199	191	0
-100	100	0	-194	186	0
-100	100	0	-189	180	0
-100	100	0	-185	176	0
-100	100	0	-180	171	0
-100	100	4	-175	165	0
-100	100	0	-170	159	0
-100	100	0	-165	154	0
-100	100	0	-160	148	0
-100	100	0	-156	144	0
-100	100	0	-151	138	0
-100	100	0	-146	133	0
-100	100	0	-142	127	0
-100	100	0	-136	122	0
-100	100	0	-131	116	0
-100	100	4	-127	112	0
-100	100	4	-122	106	0
-100	100	0	-117	101	0
-100	100	0	-112	95	0
-100	100	0	-108	89	0
-100	100	0	-102	83	0
-100	100	0	-99	79	0
-100	100	0	-94	73	0
-100	100	0	-89	68	0
-100	100	0	-84	62	0
-100	100	0	-78	57	0
-100	100	0	-73	51	0
-100	100	9	-68	45	0
-100	100	0	-64	41	0
-100	100	0	-59	35	0
-100	100	0	-54	29	0
-100	100	0	-49	24	0
-100	100	0	-44	18	0
-100	100	0	-39	13	0
-100	100	0	-35	8	0
-100	100	0	-31	3	0
-100	-61	-4	-26	-1	0
-100	-23	0	-21	0	0
-100	17	4	-17	0	0
-100	0	-8	-12	0	0
-100	-4	-9	-8	0	0

Nilai PID			Nilai Pengukuran		
X	Y	Z	X	Y	Z
-100	-4	4	-4	0	0
11	0	0	0	0	0
Waktu Tempuh (Detik)				16.03 detik	

Pada Tabel 4.26 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=5$ dan $K_d=4$. Hasil pngujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 30 cm. sehingga pada pengujian ini memiliki waktu tempuh yang lebih cepat yaitu 16.03 detik.

G.2 Pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Tabel 4. 27 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
0	0	0	0	0	0
-100	100	0	-249	250	0
-100	100	0	-248	248	0
-100	100	0	-244	244	0
-100	100	-4	-239	238	0
-100	100	0	-234	232	0
-100	100	0	-230	228	0
-100	100	4	-225	223	0
-100	100	0	-220	217	0
-100	100	0	-215	211	0
-100	100	0	-210	205	0
-100	100	0	-205	199	0
-100	100	4	-201	194	0
-100	100	0	-196	188	0
-100	100	4	-191	183	0
-100	100	0	-186	177	0
-100	100	5	-181	171	0
-100	100	0	-176	165	0
-100	100	0	-172	161	0
-100	100	0	-167	155	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	Z	X	Y	Z
-100	100	0	-162	150	0
-100	100	0	-157	144	0
-100	100	0	-152	139	0
-100	100	4	-147	133	0
-100	100	0	-142	127	0
-100	100	0	-138	123	0
-100	100	0	-133	117	0
-100	100	0	-128	111	0
-100	100	0	-123	105	0
-100	100	4	-118	100	0
-100	100	0	-113	94	0
-100	100	0	-109	90	0
-100	100	0	-104	84	0
-100	100	0	-99	78	0
-100	100	0	-94	72	0
-100	100	4	-89	66	0
-100	100	4	-83	60	0
-100	100	0	-79	56	0
-100	100	0	-74	50	0
-100	100	4	-69	44	0
-100	100	0	-64	38	0
-100	100	0	-59	32	0
-100	100	4	-53	27	0
-100	100	4	-50	22	0
-100	100	0	-44	16	0
-100	100	0	-39	11	0
-100	100	0	-34	5	0
-100	-20	4	-29	0	0
-100	-65	0	-24	-1	0
-100	27	4	-19	0	0
-100	15	0	-16	0	0
-100	-5	0	-11	0	0
-100	0	0	-6	0	0
-100	0	0	-2	0	0
62	0	4	1	0	0
11	0	0	0	0	0
Waktu Tempuh (detik)				16.07 detik	

Pada Tabel 4.27 dilakukan pengujian pergerakan robot menggunakan nilai

input $K_p=4$, $K_i=9$, $K_d=4$. Hasil pengujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 50 cm. sehingga pada pengujian ini memiliki waktu tempuh yang cepat yaitu dengan waktu tempuh 16.07 detik.

G.3 Pengujian menggunakan nilai $K_p=6$, $K_i=3$, $K_d=6$

Tabel 4. 28 Tabel pengujian menggunakan nilai $K_p=6$, $K_i=3$, $K_d=6$

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
-100	100	0	-24	249	0
-100	100	-25	-24	247	0
-100	100	15	-24	242	0
-100	100	26	-23	236	0
-100	100	-24	-23	230	0
-100	100	-1	-22	224	0
-100	100	33	-22	217	0
-100	100	-22	-21	211	0
-100	100	-10	-20	204	0
-100	100	49	-20	197	0
-100	100	14	-19	192	0
-100	100	-51	-19	186	0
-100	100	55	-18	179	0
-100	100	-3	-18	173	0
-100	100	-47	-17	170	0
-100	100	13	-17	164	0
-100	100	67	-17	157	0
-100	100	-46	-16	152	0
-100	100	-3	-15	146	0
-100	100	43	-15	139	0
-100	100	-39	-14	134	0
-100	100	4	-14	127	0
-100	100	35	-13	121	0
-100	100	-24	-12	115	0
-100	100	-2	-12	109	0
-100	100	42	-11	102	0
-100	100	-22	-11	96	0
-100	100	-2	-10	90	0
-100	100	42	-10	83	0
-100	100	-22	-94	77	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-100	100	6	-87	72	0
-100	100	25	-81	65	0
-100	100	-15	-76	59	0
-100	100	-2	-70	53	0
-100	100	25	-64	46	0
-100	100	1	-58	40	0
-100	100	-17	-52	34	0
-100	100	23	-45	27	0
-100	100	2	-39	21	0
-100	100	-24	-33	15	0
-100	100	-10	-30	12	0
-100	100	32	-25	5	0
-100	-31	2	-19	0	0
-100	-100	-52	-13	-2	0
-100	20	80	-7	0	0
-100	100	-67	-2	1	0
100	-74	-1	2	-1	0
38	-28	100	0	0	0
-100	89	-100	-2	1	0
-100	-100	-43	-1	-2	0
100	-9	100	1	0	0
-14	78	-60	0	0	0
-82	-83	1	0	-1	0
100	87	68	2	1	0
95	-34	-54	1	0	0
-100	-31	-13	-2	0	0
37	60	8	0	0	0
100	-9	-22	1	0	0
-100	32	13	-1	0	0
-2	-3	2	0	0	0
96	-45	-18	1	0	0
-100	100	16	-2	1	0
-25	-23	3	0	0	0
38	-32	23	0	0	0
-100	77	-22	-1	0	0
-99	14	-10	-1	0	0
98	-35	23	1	0	0
-100	25	-22	-1	0	0
-85	8	-11	0	0	0
Waktu Tempuh (detik)				19.03 detik	

Pada Tabel 4.28 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=6$, $K_i=5$, $K_d=2$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar tapi sistem kontrol mampu membuat kondisi robot kembali stabil meskipun waktu yang diperlukan untuk membuat robot kembali stabil lebih lama yaitu dengan waktu tempuh 19.03 detik.

G.4 Pengujian menggunakan nilai $K_p=7$, $K_i=3$, $K_d=3$

Tabel 4. 29 Tabel pengujian menggunakan nilai $K_p=7$, $K_i=3$, $K_d=3$

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
-100	100	-8	-249	249	0
-100	100	28	-246	245	0
-100	100	-17	-241	240	0
-100	100	8	-236	234	0
-100	100	29	-230	228	0
-100	100	-17	-225	222	0
-100	100	-1	-219	215	0
-100	100	38	-213	208	0
-100	100	-26	-206	203	0
-100	100	6	-200	196	0
-100	100	31	-194	190	0
-100	100	-37	-188	184	0
-100	100	26	-182	177	0
-100	100	23	-177	171	0
-100	100	-34	-174	168	0
-100	100	-4	-168	162	0
-100	100	58	-162	156	0
-100	100	-53	-157	150	0
-100	100	14	-151	144	0
-100	100	42	-146	137	0
-100	100	-57	-140	131	0
-100	100	53	-135	125	0
-100	100	4	-129	119	0
-100	100	-40	-122	112	0
-100	100	73	-117	105	0

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
-100	100	-24	-111	99	0
-100	100	-13	-105	93	0
-100	100	66	-100	85	0
-100	100	-42	-94	80	0
-100	100	-3	-88	73	0
-100	100	60	-82	67	0
-100	100	-64	-77	62	0
-100	100	32	-71	55	0
-100	100	62	-66	49	0
-100	100	-75	-60	43	0
-100	100	51	-55	36	0
-100	100	6	-49	31	0
-100	100	-49	-42	25	0
-100	100	72	-37	18	0
-100	100	60	-34	15	0
-100	100	-64	-29	10	0
-100	100	42	-23	3	0
-100	-100	5	-18	-1	0
-100	-100	16	-13	-1	0
-100	100	-6	-8	2	0
-100	-60	-12	-4	0	0
-1	-100	38	0	-1	0
100	100	-25	1	1	0
-100	-75	-2	-1	0	0
-3	-18	-25	0	0	0
100	92	34	2	0	0
-30	-100	-26	0	-1	0
-100	46	13	-1	0	0
100	39	42	1	0	0
21	-100	-100	0	-1	0
-69	8	75	0	0	0
94	100	-28	1	1	0
-59	-100	-100	0	-1	0
13	62	100	0	0	0
-16	100	-100	0	1	0
11	-100	62	0	-1	0
-100	56	60	-1	0	0
59	75	-100	0	0	0
-8	-100	100	0	-1	0
-100	-56	93	-1	0	0
-100	85	-100	-1	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
38	-100	63	0	-1	0
49	-100	100	0	-1	0
49	-100	100	0	-1	0
49	-100	100	0	-1	0
Waktu Tempuh (Detik)				21.05	

Pada Tabel 4.29 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=7$, $K_i=3$, $K_d=3$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar tapi sistem kontrol mampu membuat kondisi robot kembali stabil meskipun waktu yang diperlukan untuk membuat robot kembali stabil lebih lama yaitu dengan waktu tempuh 21.05 detik.

H. Hasil pengujian untuk robot bergerak diagonal dengan kecepatan 150.

H.1 Pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Tabel 4. 30 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$ $K_d=4$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
0	0	0	0	0	0
-150	150	0	-249	249	0
-150	150	13	-247	247	0
-150	150	18	-242	241	0
-150	150	9	-235	234	0
-150	150	18	-229	227	0
-150	150	18	-224	221	0
-150	150	13	-217	214	0
-150	150	13	-210	206	0
-150	150	18	-204	199	0
-150	150	17	-197	191	0
-150	150	13	-190	184	0
-150	150	13	-185	178	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-150	150	18	-178	171	0
-150	150	13	-171	164	0
-150	150	18	-164	157	0
-150	150	13	-157	149	0
-150	150	13	-150	142	0
-150	150	13	-143	134	0
-150	150	13	-138	129	0
-150	150	18	-132	121	0
-150	150	13	-125	114	0
-150	150	13	-117	107	0
-150	150	17	-110	99	0
-150	150	18	-104	92	0
-150	150	17	-98	87	0
-150	150	9	-91	79	0
-150	150	17	-84	72	0
-150	150	17	-77	64	0
-150	150	18	-71	57	0
-150	150	13	-64	49	0
-150	150	18	-59	43	0
-150	150	13	-52	35	0
-150	150	13	-45	28	0
-150	150	13	-38	20	0
-150	150	17	-31	13	0
-150	150	22	-25	6	0
-150	34	22	-20	0	0
-150	-10	-21	-11	-2	0
-150	-10	0	-4	0	0
65	36	4	1	0	0
36	22	-13	0	0	0
Waktu Tempuh (detik)				12.05 detik	

Pada Tabel 4.29 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=5$ dan $K_d=4$. Hasil pngujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 30 cm. sehingga pada pengujian ini memiliki waktu tempuh yang lebih cepat yaitu 12.05 detik.

H.2 Pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Tabel 4. 31 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
0	0	0	0	0	0
-150	150	0	-249	249	0
-150	150	19	-246	246	0
-150	150	10	-241	239	0
-150	150	4	-236	234	0
-150	150	19	-229	226	0
-150	150	24	-223	219	0
-150	150	10	-216	211	0
-150	150	14	-210	204	0
-150	150	19	-203	196	0
-150	150	19	-198	191	0
-150	150	9	-191	184	0
-150	150	19	-184	177	0
-150	150	15	-178	169	0
-150	150	14	-171	162	0
-150	150	24	-164	155	0
-150	150	9	-158	149	0
-150	150	14	-152	142	0
-150	150	15	-145	134	0
-150	150	15	-138	127	0
-150	150	19	-130	120	0
-150	150	14	-123	113	0
-150	150	14	-118	108	0
-150	150	19	-111	101	0
-150	150	9	-104	94	0
-150	150	19	-97	86	0
-150	150	14	-90	79	0
-150	150	15	-83	72	0
-150	150	14	-77	64	0
-150	150	15	-71	58	0
-150	150	14	-65	51	0
-150	150	14	-58	43	0
-150	150	15	-50	36	0
-150	150	4	-44	29	0
-150	150	19	-37	21	0
-150	150	19	-32	15	0
-150	150	14	-25	8	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-150	74	19	-18	1	0
-150	-11	-9	-11	-2	0
-150	-13	9	-4	0	0
89	45	0	2	0	0
70	58	-23	1	1	0
-13	-13	0	0	0	0
-19	-19	0	0	0	0
Waktu Tempuh				13.08 detik	

Pada Tabel 4.31 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=9$, $K_d=4$. Hasil pengujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 50 cm. sehingga pada pengujian ini memiliki waktu tempuh yang cepat yaitu dengan waktu tempuh 13.08 detik.

H.3 Pengujian menggunakan nilai $K_p=5$, $K_i=3$, $K_d=7$

Tabel 4. 32 Tabel pengujian menggunakan nilai $K_p=5$, $K_i=3$, $K_d=7$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
7	0	0	0	0	0
7	0	0	0	0	0
-150	150	0	-249	249	0
-150	150	14	-248	247	0
-150	150	47	-242	240	0
-150	150	-6	-235	233	0
-150	150	7	-227	225	0
-150	150	30	-220	217	0
-150	150	23	-212	209	0
-150	150	8	-204	201	0
-150	150	15	-196	193	0
-150	150	30	-189	185	0
-150	150	15	-181	177	0
-150	150	15	-173	170	0
-150	150	23	-165	162	0
-150	150	15	-157	154	0
-150	150	15	-149	146	0

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
-150	150	15	-145	142	0
-150	150	23	-137	135	0
-150	150	15	-130	127	0
-150	150	15	-122	119	0
-150	150	8	-114	111	0
-150	150	23	-106	103	0
-150	150	15	-98	95	0
-150	150	15	-91	88	0
-150	150	15	-83	80	0
-150	150	46	-74	73	0
-150	150	-7	-66	66	0
-150	150	7	-58	58	0
-150	150	38	-50	49	0
-150	150	16	-43	41	0
-150	150	7	-35	34	0
-150	150	31	-26	27	0
-150	150	0	-19	19	0
-150	150	15	-11	11	0
-150	150	15	-4	3	0
150	-150	8	3	-3	0
150	-150	-53	3	-3	0
-150	85	-9	-2	1	0
-150	150	29	-5	3	0
-150	150	93	-2	2	0
150	-148	-66	2	-2	0
150	-150	-110	2	-2	0
6	-90	-4	0	-1	0
-150	150	100	-3	4	0
-67	150	40	0	4	0
129	-44	-45	1	0	0
-56	-150	-8	0	-2	0
-19	131	22	0	1	0
85	0	38	1	0	0
-66	-150	-37	0	-2	0
11	94	-2	0	1	0
123	5	63	1	0	0
-87	-150	-122	-1	-2	0
-89	9	-27	-1	0	0
150	50	85	2	0	0
6	-123	-74	0	-1	0
-95	64	11	-1	0	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
46	132	78	0	1	0
46	132	78	0	1	0
Waktu Tempuh (Detik)			17.09 detik		

Pada Tabel 4.31 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=5$, $K_i=3$, $K_d=7$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar tapi sistem kontrol mampu membuat kondisi robot kembali stabil meskipun waktu yang diperlukan untuk membuat robot kembali stabil lebih lama yaitu dengan waktu tempuh 17.09 detik.

H.4 Pengujian menggunakan nilai $K_p=7$, $K_i=8$, $K_d=2$

Tabel 4. 33 Tabel pengujian menggunakan nilai $K_p=7$, $K_i=8$, $K_d=2$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
8	0	0	0	0	0
-150	150	0	-249	249	0
-150	150	-10	-248	246	0
-150	150	15	-242	239	0
-150	150	42	-235	230	0
-150	150	2	-227	222	0
-150	150	15	-220	214	0
-150	150	34	-212	206	0
-150	150	9	-204	198	0
-150	150	16	-196	190	0
-150	150	33	-189	182	0
-150	150	49	-180	175	0
-150	150	51	-176	172	0
-150	150	-22	-167	165	0
-150	150	6	-159	157	0
-150	150	58	-152	148	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-150	150	-13	-144	141	0
-150	150	-2	-137	134	0
-150	150	58	-130	125	0
-150	150	-13	-121	117	0
-150	150	-2	-113	110	0
-150	150	50	-106	101	0
-150	150	20	-98	92	0
-150	150	0	-90	85	0
-150	150	32	-83	76	0
-150	150	26	-75	68	0
-150	150	8	-67	60	0
-150	150	16	-60	52	0
-150	150	42	-52	44	0
-150	150	43	-43	38	0
-150	150	-4	-35	33	0
-150	150	-10	-27	26	0
-150	150	49	-20	17	0
-150	150	35	-12	8	0
-150	77	0	-5	0	0
150	-150	-16	2	-4	0
150	-132	73	2	-1	0
-5	76	30	0	1	0
-150	22	-131	-5	0	0
-150	-150	32	-3	-3	0
90	-104	150	1	-1	0
-63	150	-150	0	2	0
-150	-137	4	-3	-1	0
-57	-150	150	0	-2	0
113	122	-116	1	1	0
-144	115	-11	-1	1	0
-109	-150	150	-1	-2	0
59	-64	3	0	0	0
-150	150	-123	-2	2	0
-150	-114	115	-1	-1	0
-60	-150	150	0	-2	0
-60	-150	150	0	-2	0
-60	-150	150	0	-2	0
Waktu Tempuh (Detik)				16.04 detik	

Pada Tabel 4.32 dilakukan pengujian pergerakan robot menggunakan nilai

input $K_p=6$, $K_i=5$, $K_d=2$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar tapi sistem kontrol mampu membuat kondisi robot kembali stabil meskipun waktu yang diperlukan untuk membuat robot kembali stabil lebih lama yaitu dengan waktu tempuh 16.04 detik.

I. Hasil pengujian untuk robot bergerak diagonal dengan kecepatan 200.

I.1 Pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Tabel 4. 34 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=5$, $K_d=4$

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
0	0	0	0	0	0
-200	200	0	-249	249	0
-200	200	35	-245	244	0
-200	200	31	-241	238	0
-200	200	31	-233	230	0
-200	200	22	-225	222	0
-200	200	27	-218	214	0
-200	200	27	-210	205	0
-200	200	31	-202	196	0
-200	200	31	-196	190	0
-200	200	36	-188	181	0
-200	200	36	-180	173	0
-200	200	27	-172	164	0
-200	200	31	-164	156	0
-200	200	31	-156	147	0
-200	200	31	-148	139	0
-200	200	27	-142	132	0
-200	200	31	-134	124	0
-200	200	31	-125	115	0
-200	200	31	-117	107	0
-200	200	31	-110	98	0
-200	200	31	-102	89	0
-200	200	27	-96	83	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-200	200	31	-88	74	0
-200	200	31	-80	66	0
-200	200	31	-72	57	0
-200	200	31	-63	49	0
-200	200	27	-55	40	0
-200	200	31	-50	34	0
-200	200	31	-42	25	0
-200	200	36	-34	17	0
-200	200	31	-26	8	0
-200	16	22	-18	0	0
-200	-11	-26	-8	-2	0
-93	-50	-5	-2	-1	0
152	62	9	3	1	0
51	19	0	1	0	0
-45	-22	-4	-1	0	0
-13	-4	0	0	0	0
Waktu Tempuh				08.02 detik	

Pada Tabel 4.33 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=5$ dan $K_d=4$. Hasil pengujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 30 cm. sehingga pada pengujian ini memiliki waktu tempuh yang lebih cepat yaitu 08.02 detik.

I.2 Pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Tabel 4. 35 Tabel pengujian menggunakan nilai $K_p=4$, $K_i=9$, $K_d=4$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
4	0	0	0	0	0
4	0	0	0	0	0
-200	200	14	-248	248	0
-200	200	39	-244	242	0
-200	200	29	-238	236	0
-200	200	24	-231	228	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
-200	200	24	-223	220	0
-200	200	38	-215	211	0
-200	200	34	-207	202	0
-200	200	29	-199	193	0
-200	200	24	-191	185	0
-200	200	24	-186	178	0
-200	200	29	-178	169	0
-200	200	34	-170	161	0
-200	200	34	-162	152	0
-200	200	34	-153	143	0
-200	200	29	-145	135	0
-200	200	34	-139	129	0
-200	200	29	-131	120	0
-200	200	34	-123	112	0
-200	200	29	-115	103	0
-200	200	34	-107	94	0
-200	200	29	-99	86	0
-200	200	29	-93	80	0
-200	200	33	-84	71	0
-200	200	29	-76	62	0
-200	200	29	-68	54	0
-200	200	34	-60	45	0
-200	200	39	-52	36	0
-200	200	29	-46	30	0
-200	200	34	-38	21	0
-200	200	29	-30	13	0
-200	200	29	-22	5	0
-200	-88	6	-14	-2	0
-198	-89	-20	-4	-1	0
132	42	9	3	0	0
124	39	9	2	0	0
-31	-4	0	0	0	0
Waktu Tempuh (detik)				09.05 detik	

Pada Tabel 4.34 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=4$, $K_i=9$, $K_d=4$. Hasil pngujian yang dilakukan menunjukkan respon robot yang stabil, karena robot tidak mengalami perubahan nilai yang signifikan sehingga error yang terjadi pada robot kecil yaitu ≤ 50 cm. sehingga pada pengujian ini

memiliki waktu tempuh yang cepat yaitu dengan waktu tempuh 09.05 detik.

I.3 Pengujian menggunakan nilai $K_p=5$, $K_i=3$, $K_d=6$

Tabel 4. 36 Tabel pengujian menggunakan nilai $K_p=5$, $K_i=3$, $K_d=6$

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
0	0	0	0	0	0
-200	200	4	-249	249	0
-200	200	53	-245	244	0
-200	200	51	-238	236	0
-200	200	27	-229	229	0
-200	200	23	-220	221	0
-200	200	23	-216	216	0
-200	200	31	-207	207	0
-200	200	41	-198	198	0
-200	200	41	-189	189	0
-200	200	45	-180	181	0
-200	200	63	-170	175	0
-200	200	46	-161	167	0
-200	200	23	-152	160	0
-200	200	27	-143	151	0
-200	200	31	-134	142	0
-200	200	45	-125	133	0
-200	200	50	-116	125	0
-200	200	41	-107	116	0
-200	200	32	-98	108	0
-200	200	32	-89	99	0
-200	200	36	-80	91	0
-200	200	41	-72	82	0
-200	200	41	-63	73	0
-200	200	36	-54	65	0
-200	200	32	-45	56	0
-200	200	31	-36	47	0
-200	200	41	-27	38	0
-200	200	54	-18	30	0
-200	200	55	-9	23	0
-36	200	28	0	16	0
129	200	27	3	11	0
200	168	-47	5	3	0
29	51	-101	0	1	0
-118	63	47	-2	1	0
0	51	20	0	1	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
54	-31	-18	1	0	0
-3	8	0	0	0	0
-9	13	0	0	0	0
-9	18	4	0	0	0
Waktu Tempuh (Detik)				tidak terdefinisi	

Pada Tabel 4.35 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=5$, $K_i=3$, $K_d=6$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar dan sulit untuk sistem kontrol membuat robot kembali dalam keadaan stabil yang memiliki dampak robot tidak bisa berhenti.

I.4 Pengujian menggunakan nilai $K_p=6$, $K_i=6$, $K_d=3$

Tabel 4. 37 Tabel pengujian menggunakan nilai $K_p=6$, $K_i=6$, $K_d=3$

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
8	-8	8	0	0	0
-200	200	16	-249	248	0
-200	200	91	-245	244	0
-200	200	39	-238	239	0
-200	200	3	-233	235	0
-200	200	7	-224	227	0
-200	200	41	-216	217	0
-200	200	76	-207	208	0
-200	200	11	-198	200	0
-200	200	16	-189	192	0
-200	200	41	-181	183	0
-200	200	60	-172	173	0
-200	200	35	-163	165	0
-200	200	17	-154	157	0
-200	200	41	-145	148	0

Nilai PID		Nilai Pengukuran (cm)			
X	Y	X	Y	X	Y
-200	200	43	-137	139	0
-200	200	43	-128	130	0
-200	200	26	-119	122	0
-200	200	25	-110	113	0
-200	200	50	-101	104	0
-200	200	52	-92	95	0
-200	200	17	-84	87	0
-200	200	25	-75	78	0
-200	200	42	-66	69	0
-200	200	51	-57	60	0
-200	200	34	-48	51	0
-200	200	50	-39	44	0
-200	200	43	-30	35	0
-200	200	17	-21	27	0
-200	200	25	-17	23	0
-200	200	42	-8	14	0
-64	200	43	0	5	0
200	-200	3	5	-3	0
200	-200	-15	2	-6	0
-200	-199	-27	-3	-2	0
129	200	35	1	4	0
200	200	-14	5	3	0
103	200	-14	1	2	0
-200	38	192	-3	0	0
-140	3	164	-1	0	0
184	53	-20	2	0	0
107	55	-82	1	0	0
-200	132	191	-3	1	0
-89	200	-11	0	3	0
171	33	-11	2	0	0
-104	11	89	-1	0	0
-150	119	-60	-1	1	0
56	-167	-58	0	-2	0
107	-71	200	1	0	0
-102	140	-14	-1	1	0
-138	-82	-15	-1	-1	0
-30	-170	200	0	-2	0
-17	-172	200	0	-2	0
-17	-172	200	0	-2	0
-17	-172	200	0	-2	0
-17	-172	200	0	-2	0

Nilai PID			Nilai Pengukuran (cm)		
X	Y	X	Y	X	Y
Waktu Tempuh (Detik)			Tidak terdefinisi		

Pada Tabel 4.36 dilakukan pengujian pergerakan robot menggunakan nilai input $K_p=6$, $K_i=6$, $K_d=3$. Pada pengujian ini robot menunjukkan respon yang tidak stabil karena mengalami perubahan nilai yang signifikan. Pergerakan robot mengalami osilasi sehingga membuat kondisi robot mengalami error yang besar dan sulit untuk sistem kontrol membuat robot kembali dalam keadaan stabil yang memiliki dampak robot tidak bisa berhenti.



J. Pemilihan nilai Kp, Ki dan Kd terbaik

J.1 Pemilihan nilai Kp,Ki, Kd terbaik untuk pergerakan maju

Tabel 4. 38 Pemilihan nilai Kp, Ki, Kd gerak maju

No	Arah pergerakan robot	Kecepatan robot (PWM)	Nilai Kp, Ki, Kd	Kategori Error	Waktu (detik)
1.	Maju	100	4 5 4	A	16.02
2.	Maju	100	4 9 4	B	17.04
3.	Maju	150	4 5 4	B	13.05
4.	Maju	150	4 9 4	A	14.03
5.	Maju	200	4 5 4	B	9.03
6.	Maju	200	4 9 4	A	8.02

Berdasarkan pengujian sistem untuk pergerakan robot maju terdapat nilai Kp, Ki, Kd yang memiliki nilai error yang kecil dan waktu pergerakan yang cepat, nilai tersebut akan mempengaruhi kestabilan pada robot untuk mencapai jarak finish. Pada tabel diatas terdapat nilai terbaik yaitu Kp= 4, Ki=5, Kd=4 yang memiliki nilai eror yang kecil dan waktu tempuh paling cepat yaitu 8.02 detik

J.2 Pemilihan nilai Kp,Ki, Kd terbaik untuk pergerakan kearah kanan

Tabel 4. 39 Pemilihan nilai Kp, Ki, Kd pergerakan ke arah kanan

No	Arah pergerakan robot	Kecepatan robot (PWM)	Nilai Kp, Ki, Kd	Kategori Error	Waktu (detik)
1.	kanan	100	4 5 4	B	17.02
2.	kanan	100	4 9 4	A	17.08
4.	kanan	150	4 9 4	B	13.05
5.	kanan	150	4 5 4	A	12.08
6.	kanan	200	4 9 4	B	8.06
7.	kanan	200	4 5 4	A	8.03

Berdasarkan pengujian sistem untuk pergerakan robot kearah kanan terdapat nilai Kp, Ki, Kd yang memiliki nilai error yang kecil dan waktu pergerakan yang cepat, nilai tersebut akan mempengaruhi kestabilan pada robot untuk mencapai jarak finish. Pada tabel diatas terdapat nilai terbaik yaitu Kp= 4, Ki=5, Kd=4 yang memiliki nilai eror yang kecil dan waktu tempuh paling cepat yaitu 8.03 detik

J.3. Pemilihan nilai Kp,Ki, Kd terbaik untuk pergerakan diagonal.

Tabel 4. 40 Pemilihan nilai Kp, Ki, Kd pergerakan ke arah kanan

No	Arah pergerakan robot	Kecepatan robot (PWM)	Nilai Kp, Ki, Kd	Kategori Error	Waktu (detik)
1.	Diagonal	100	4 5 4	A	16.03
2.	Diagonal	100	4 9 4	B	16.08
3.	Diagonal	150	4 5 4	B	13.08
4.	Diagonal	150	4 9 4	A	12.02
5.	Diagonal	200	4 9 4	B	9.05
6.	Diagonal	200	4 5 4	A	8.06

Berdasarkan pengujian sistem untuk pergerakan robot ke arah diagonal terdapat nilai Kp, Ki, Kd yang memiliki nilai eror yang kecil dan waktu pergerakan yang cepat, nilai tersebut akan mempengaruhi kestabilan pada robot untuk mencapai jarak finish. Pada tabel diatas terdapat nilai terbaik yaitu Kp= 4, Ki=5, Kd=4 yang memiliki nilai eror yang kecil dan waktu tempuh paling cepat yaitu 8.06 detik

BAB V

PENUTUP

Dalam bab ini berisi tentang kesimpulan dan saran yang dicantumkan berdasarkan dari hasil pengujian yang telah dilakukan selama mengerjakan tugas akhir ini, maka dapat diperoleh beberapa kesimpulan dan saran untuk pengembangan sistem berikutnya.

5.1 Kesimpulan

Berdasarkan hasil pengujian pada proyek Tugas akhir ini dapat disimpulkan beberapa poin sebagai berikut:

1. Hasil dari proses pengujian kecepatan pada motor DC yang digunakan untuk proyek tugas akhir ini masing-masing memiliki perbedaan kecepatan yang tidak signifikan.
2. Hasil pengujian penggunaan tabel pengamatan nilai konstanta akan membuat proses tuning nilai K_p , K_i dan K_d lebih cepat, karena tidak perlu melakukan operasi perhitungan secara matematika sehingga komposisi nilai dapat diatur oleh pengguna sesuai keinginan. Karena proses pemilihannya yang cepat dibandingkan dengan percobaan trial and error konvensional yang harus dilakukan secara satu per satu sehingga membutuhkan waktu yang sangat lama untuk mendapatkan nilai PID terbaik.
3. Dari hasil penelitian yang dilakukan, didapatkan hasil respon pergerakan robot dari berbagai nilai variabel K_p , K_i , K_d untuk jarak dan arah pergerakan robot yang berbeda-beda. Untuk pergerakan robot maju nilai variabel K_p , K_i , K_d dengan nilai error kurang dari 30 cm dan waktu tempuh 8.02 detik yaitu $K_p=4$, $K_i=5$, $K_d=4$, Sedangkan untuk pergerakan robot ke arah kanan nilai

variable K_p , K_i , K_d dengan nilai eror kurang dari 30 cm dan waktu tempuh 08.4 detik yaitu $K_p=4$, $K_i=5$, $K_d=4$, untuk pergerakan diagonal pada robot nilai variable K_p , K_i , K_d dengan nilai eror kurang dari 30 cm dan waktu tempuh pada rentang 8.02 detik yaitu $K_p=4$, $K_i=5$, $K_d=4$.

4. Syarat untuk membuat sebuah sistem kontrol stabil adalah membuat nilai K_i bernilai pada rentang angka 0,1-1 karena K_i yang bernilai besar akan mempercepat hilangnya offset, namun semakin besar nilai konstanta K_i akan mengakibatkan peningkatan osilasi dari sinyal keluaran pengontrol

5.2 Saran

Saran yang diberikan oleh penulis pada pengembangan proyek tugas akhir ini selanjutnya adalah sebagai berikut:

1. Pembuatan mekanik pada robot akan berpengaruh besar, karena penataan tata letak motor dan sensor external rotary sangat tergantung pada ketelitian pada rancang bangun mekanik, sehingga pembuatan mekanik yang baik seperti percetakan frame dan base robot akan membuat hasil lebih baik.
2. Penggunaan motor DC pada sebuah robot yang berfungsi sebagai actuator memiliki peranan yang sangat penting, untuk itu sebelum pemasangan pada sebuah robot lebih baik setiap motor diperiksa apakah memiliki kecepatan yang sama.
3. Dalam pengujian sistem pada proyek tugas akhir ini nilai K_p , K_i dan K_d yang digunakan oleh penulis adalah bilangan bulat yaitu 1 sampai 10. Untuk memperhalus gerakan akan lebih baik jika menggunakan nilai bilangan dibelakang koma.

DAFTAR PUSTAKA

- (FEC), F. E. (2016). *Datasheet Arduino DUE*. Chicago: Future Electronic Corporation (FEC).
- Kementerian Riset, T. d. (2018). "*Tentang KRI*". Retrieved Juli 11, 2018, from <http://kontesrobotindonesia.id/tentang-kri.html>
- P.B, P. A. (2018). *Kendali Temperatur Menggunakan PID Untuk Sistem Pasteurisasi Susu* (1 edisi.). SURABAYA: INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA.
- PRASTYAWAN, A. W. (2017). *Kendali Pergerakan Robot Sepak Bola Dengan PID (Proportional Integral Derivatif)*. SURABAYA: INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA.
- SAPUTRA, A. W., HARIANTO, & MARDIANA, D. G. (2014). Kendali Kecepatan dan Posisi Pada Mobile Robot yang menggunakan Triangle OMNI-DIRECTIONAL Wheels Dengan Metode PID. *JOURNAL OF CONTROL AND NETWORK SYSTEM*, 3, 81-89.
- Kariyanto, J. D., Alasiry, A. H., Ardila, F., & Hanafi, N (2012). Navigasi Mobile Robot Berbasis Trajektori dan Odometry Dengan Memulihkan Jalur Secara Otomatis. Politeknik Elektronika Negeri Surabaya. Institut Teknologi Sepuluh Nopember Surabaya Kampus ITS Sukolilo, Surabaya, 1-8
- Marta, B. S., Ardila, F., & Besari, A., (2011). Path Tracking Pada Mobile Robot Dengan Umpan Balik Odometri. Politeknik Elektronika Negeri Surabaya. Institut Teknologi Sepuluh Nopember Surabaya Kampus ITS Sukolilo, Surabaya, 1-7

Rochmanto, & Artha, R. (2014). Implementasi Robot Three Omi-Directional Menggunakan Kontroler PID Pada Robot Kontes Robot ABU Indonesia (KRAI). Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya, Surabaya, 1-7

Setiawan, S., Firdaus, Rahmadya, B., & Derisma. (2015). Penerapan Invers Kinematika Untuk Pergerakan Kaki Robot BIPED. Fakultas Teknik Universitas Muhammadiyah, Jakarta.



