



UNIVERSITAS INDONESIA

**RANCANG BANGUN SISTEM NAVIGASI ROBOT
PENERIMA TAMU**

SKRIPSI

NOVIKA GINANTO

0806 455383

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
DEPOK
JUNI 2012**



UNIVERSITAS INDONESIA

**RANCANG BANGUN SISTEM NAVIGASI ROBOT
PENERIMA TAMU**

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

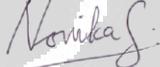
NOVIKA GINANTO

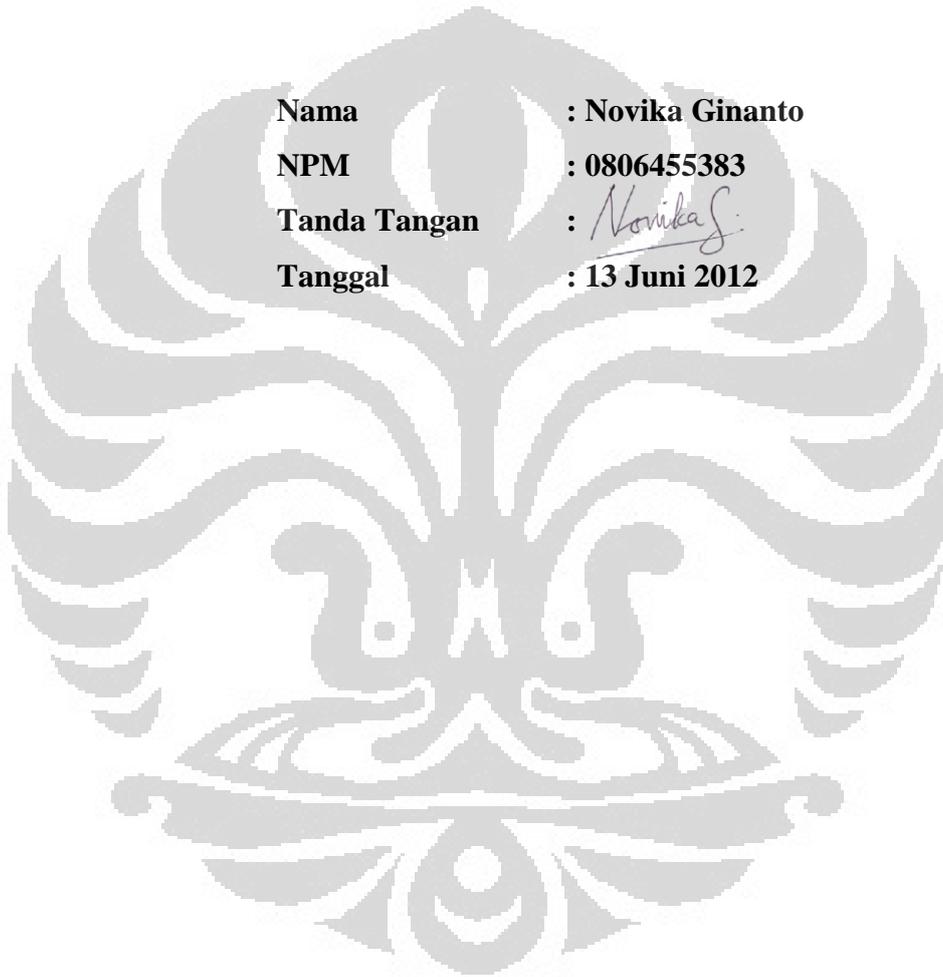
0806 455383

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
DEPOK
JUNI 2012**

HALAMAN PERNYATAAN ORISINALITAS

**Skripsi ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.**

Nama : Novika Ginanto
NPM : 0806455383
Tanda Tangan : 
Tanggal : 13 Juni 2012



HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :
Nama : Novika Ginanto
NPM : 0806455383
Program Studi : Teknik Elektro
Judul Skripsi : Rancang Bangun Sistem Navigasi Robot Penerima
Tamu

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia.

DEWAN PENGUJI

Pembimbing :
Dr. Abdul Muis S.T, M.Eng.

()

Penguji :
Ir. Wahidin Wahab M.Sc, Ph.D.

()

Prof. Dr. Eng. Drs. Benyamin Kusumoputro M.Eng.

()

Ditetapkan di : Depok

Tanggal : 25 Juni 2012

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI
UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Novika Ginanto

NPM : 0806455383

Program Studi : Teknik Elektro

Departemen : Teknik Elektro

Fakultas : Teknik

Jenis karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif** (*Non-exclusive Royalty-Free Right*) atas karya ilmiah saya yang berjudul:

RANCANG BANGUN SISTEM NAVIGASI ROBOT

PENERIMA TAMU

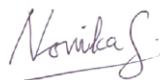
beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis / pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di: Depok

Pada tanggal : 13 Juni 2012

Yang menyatakan,



(Novika Ginanto)

ABSTRAK

Nama : Novika Ginanto

Program Studi : Teknik Elektro

Judul : Rancang Bangun Sistem Navigasi Robot Penerima Tamu

Abad ke-21 adalah abad perkembangan teknologi robotika untuk menuju era kecanggihan. Berbagai jenis robot sedang dikembangkan untuk diaplikasikan di masa mendatang. Salah satu robot yang dapat diaplikasikan adalah robot penerima tamu. Peranan robot penerima tamu dirasakan sangat penting untuk membantu dan mempermudah pekerjaan manusia. Salah satunya adalah dalam menyapa dan menjamu para tamu dalam suatu acara tertentu seperti pesta dan pada tempat-tempat tertentu seperti rumah makan, tempat hiburan dan pameran. Dengan demikian untuk masa mendatang peranan penerima atau penyambut tamu dapat digantikan dengan sebuah robot yang ramah dan mengerti keadaan tamunya. Demi menciptakan robot penerima tamu yang baik, salah satunya adalah melalui sistem navigasi robot yang handal dalam berbagai keadaan. Sistem navigasi robot penerima tamu yang dirancang menggunakan sistem *mobile robot* dengan menggunakan *Dead Reckoning* untuk memperbaharui posisi dan pengendali PD untuk mengkompensasi eror posisi navigasi melalui trajektori tertentu. Pengujian dilakukan dengan $K_p=2350$ dan $K_d=1300$ dan didapatkan eror yang kecil. Berdasarkan hasil pengujian yang dilakukan dapat dinyatakan bahwa robot penerima tamu yang didisain mampu bernavigasi dengan baik pada koordinat tertentu untuk menuju suatu objek tanpa menabrak objek tersebut.

Kata Kunci : *Kinematics, Mobile Robot, Trajectory Planning, Odometry, Dead Reckoning, PD Controller.*

ABSTRACT

Name : Novika Ginanto
Program : Teknik Elektro
Title : Navigation System Development for Receptionist Robot

21st century is a century for robotics development to be great. In that century, some robots have been developing to be applied in the next time. One of robots that will be applied is receptionist robot. Receptionist robot is very important to help and to be easy for human works to serve the guest in party and other places like restaurants, recreation parks and exhibitions. Therefore, in the next time, the function as receptionist can be changed by robot that is very nice and understands the guest conditions. For creating the great receptionist robot, it can be done through a great robot navigation system in some conditions. Navigation system of receptionist robot is designed with mobile robot system and Dead Reckoning to update the robot position and PD controller to compensate the position error of the receptionist robot through trajectory planning. Experiments have been done at basic movement like forward and rotate movement with low error with $K_p = 2750$ and $K_d = 1300$. Based on the result of experiments it can be known that receptionist robot can navigate nicely at defined coordinates toward the object without collision.

Keywords: *Kinematics, Mobile Robot, Trajectory Planning, Odometry, Dead Reckoning, PD Controller.*

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN.....	Error! Bookmark not defined.
KATA PENGANTAR.....	iii
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	x
DAFTAR TABEL	xiv
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan dan Kontribusi Riset.....	2
1.3 Perumusan dan Pembatasan Masalah.....	3
1.4 Metode Penulisan	3
1.5 Sistematika Penulisan.....	4
BAB 2 SISTEM ROBOTIKA	5
2.1 Perangkat Keras Elektrik.....	6
2.1.1 Sensor.....	6
2.1.2 Aktuator.....	16
2.1.3 Mikrokontroler	19
2.1.4 Media Debugging.....	20
2.2 Perangkat Keras Mekanik	22
2.2.1 Aluminium	22
2.2.2 Akrilik	23
2.2.3 Bearing	24

2.2.4	Belt	25
2.2.5	Poros.....	25
2.2.6	Huben Putar.....	26
BAB 3 PERANCANGAN SISTEM ROBOT.....		27
3.1	Perancangan Mekanik Robot.....	27
3.1.1	Sistem Transmisi.....	28
3.1.2	Sistem Perodaan.....	29
3.1.3	Sistem Badan.....	31
3.2	Perancangan Sistem Kendali Robot	31
3.2.1	Sistem Elektrik.....	32
3.2.2	Sistem Kontrol Robotik	35
BAB 4 MOBILE ROBOT		40
4.1	Model Kinematika Robot	41
4.2	Algoritma Mobile Robot	43
4.3	<i>Trajectory Planning</i>	46
4.3.1	Trajektori Lurus	47
4.3.2	Trajektori Putar	48
4.3.3	Trajektori Elip.....	48
4.3.4	Trajektori Navigasi Robot Penerima Tamu	50
4.3.5	Trajektori Navigasi Robot Penerima Tamu Dengan SRF-08	53
BAB 5 PENGUJIAN		54
5.1	Kondisi Robot dan Lingkungan Pengujian	54
5.2	Perangkat Pendukung Pengujian	54
5.3	Pengujian Pengendali PD	55
5.4	Pengujian Trajektori Dasar Dengan Pengendali PD	58
5.4.1	Pengujian Trajektori Lurus	59
5.4.2	Pengujian Trajektori Putar	64
5.4.3	Pengujian Trajektori Elip.....	70
5.5	Pengujian Navigasi Robot Penerima Tamu.....	76
5.5.1	Hasil Pengujian Trajektori Navigasi Robot Penerima Tamu.....	77

5.5.2	Analisis Pengujian Trajektori Navigasi Robot Penerima Tamu	81
5.6	Pengujian Navigasi Robot Penerima Tamu Dengan SRF-08.....	82
5.6.1	Hasil Pengujian Navigasi Robot Penerima Tamu Dengan SRF-08	82
5.6.2	Analisis Pengujian Navigasi Robot Penerima Tamu Dengan SRF-08	83
BAB 6 KESIMPULAN		84
DAFTAR ACUAN.....		85
DAFTAR REFERENSI.....		86



DAFTAR GAMBAR

Gambar 1.1 Robot Penerima Tamu.....	1
Gambar 2.1 Robot Beroda dan Sistem Robotikanya	5
Gambar 2.2 Sistem Robotika	6
Gambar 2.3 Sistem Kerja Sensor Encoder	7
Gambar 2.4 Pulsa Keluaran Sensor Encoder	8
Gambar 2.5 Sistem Kerja Quadrature Encoder.....	8
Gambar 2.6 Sistem Kerja Quadrature Encoder.....	9
Gambar 2.7 HCTL-2032 dan Pinoutnya	9
Gambar 2.8 Penggunaan HCTL-2032.....	10
Gambar 2.9 Prinsip Kerja SRF-08	11
Gambar 2.10 Diagram Komunikasi I2C SRF-08.....	12
Gambar 2.11 Pin Koneksi SRF-08.....	13
Gambar 2.12 Pola Pembacaan SRF-08	14
Gambar 2.13 Pemasangan dan Pengalamanan SRF-08	15
Gambar 2.14 Spesifikasi Gearhead DC Motor	16
Gambar 2.15 Gearhead DC Motor	17
Gambar 2.16 Sinyal PWM.....	17
Gambar 2.17 Rangkaian Pengganti Motor.....	18
Gambar 2.18 Mikrokontroler ATmega2560.....	20
Gambar 2.19 Konektor PC DB9 Male dan Female.....	21
Gambar 2.20 IC MAX232 dan Pinout	21
Gambar 2.21 Skematik Komunikasi Serial.....	22
Gambar 2.22 Aluminium Kotak.....	22
Gambar 2.23 Aluminium Siku	23
Gambar 2.24 Aluminium Plat	23
Gambar 2.25 Akrilik Dalam Beraneka Warna.....	24
Gambar 2.26 Bagian Bearing.....	24
Gambar 2.27 (a) Bearing Standar (b) Bearing kuping	25

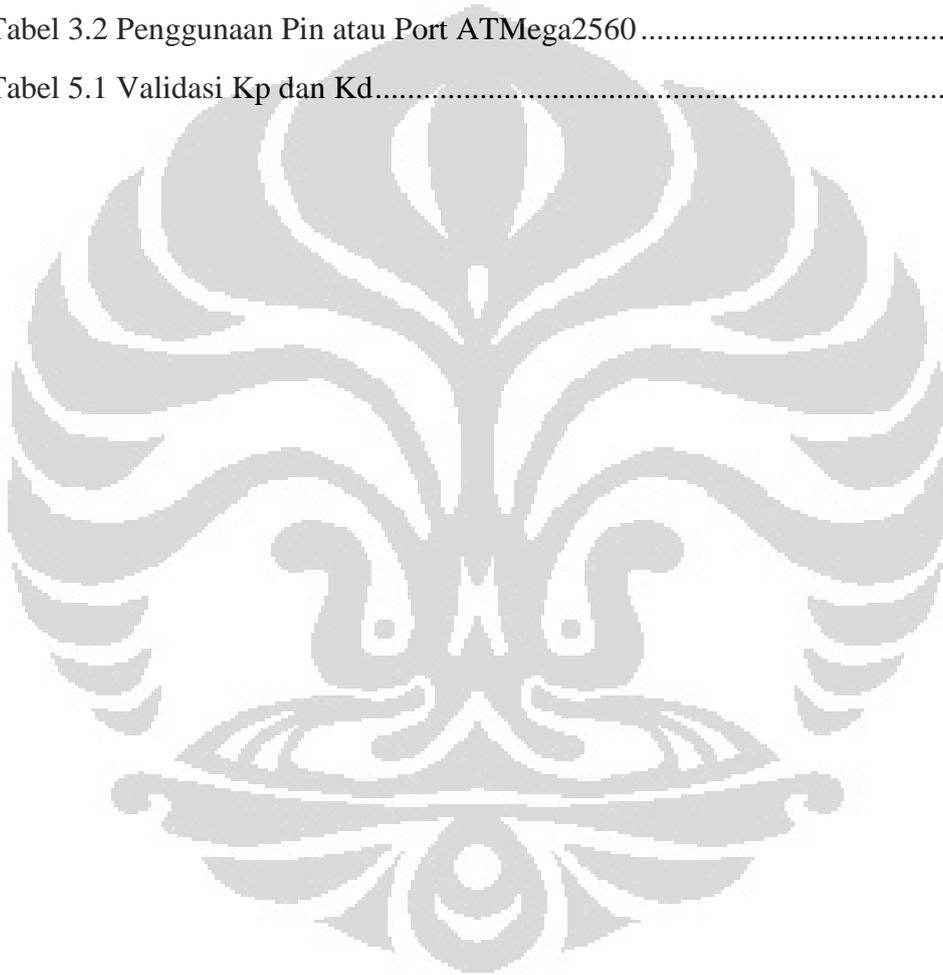
Gambar 2.28 Belt	25
Gambar 2.29 (a) Poros Standar (b) Poros Berulir	26
Gambar 2.30 Huben Putar	26
Gambar 3.1 Disain Mekanik Robot Tampak Isometris	27
Gambar 3.2 Disain Rancangan Sistem Transmisi	28
Gambar 3.3 Roda dan Pengencang Belt	30
Gambar 3.4 Sistem Perodaan Tampak Isometris	30
Gambar 3.5 Disain Rancangan Sistem Perodaan Tampak Samping	30
Gambar 3.6 Disain Sistem Perodaan Tampak Bawah	30
Gambar 3.7 Disain Badan Robot	31
Gambar 3.8 Disain Sistem Putar Badan Dengan Huben Putar	31
Gambar 3.9 Diagram Alir Sistem Elektrik	32
Gambar 3.10 Papan Sistem Minimum	33
Gambar 3.11 IRF540 dan IRF9540	33
Gambar 3.12 Diagram Alir Sistem Daya Robot	34
Gambar 3.13 LM-2576	34
Gambar 3.14 Skematik Rangkaian Volatage Regulator 5 Volt	35
Gambar 3.15 Algoritma Sistem Kontrol	35
Gambar 3.16 Kontrol P	36
Gambar 3.17 Kontrol P Pada Motor DC	36
Gambar 3.18 Kontrol D	38
Gambar 3.19 Kontrol D Pada Motor DC	38
Gambar 3.20 Kontrol PD	38
Gambar 3.21 Kontrol PD Pada Motor DC	39
Gambar 4.1 Koordinat Lokal dan Global Mobile Robot	40
Gambar 4.2 Model DMR dan TMR	41
Gambar 4.3 Algoritma Mobile Robot	43
Gambar 4.4 Diagram Alir Robot Service Mobile Robot	44
Gambar 4.5 Arah Koordinat Positif Trajektori Robot	47
Gambar 4.6 Trajektori Lurus Sejauh 2 Meter	47
Gambar 4.7 Trajektori Putar Sebesar 2 Radian	48

Gambar 4.8 Grafik Elip Untuk $0 \leq t \leq \pi$	49
Gambar 4.9 Trajektori ke Kiri.....	49
Gambar 4.10 Trajektori ke Kanan.....	50
Gambar 4.11 Navigasi Robot Penerima Tamu	51
Gambar 4.12 Trajektori dan Orientasi Objek.....	51
Gambar 4.13 Trajektori Navigasi Robot Penerima Tamu	51
Gambar 4.14 Diagram Alir Trajektori Navigasi Robot Penerima Tamu	52
Gambar 4.15 Derajat Kemiringan SRF-08.....	53
Gambar 5.1 Grafik Posisi Robot Pada Sumbu X ($K_p=500$ dan $K_d=100$)	56
Gambar 5.2 Grafik Posisi Robot Pada Sumbu X ($K_p= 5000$ dan $K_d=10000$)	56
Gambar 5.3 Grafik Posisi Robot Pada Sumbu X ($K_p=5000$ dan $K_d=10000$)	57
Gambar 5.4 Grafik Posisi Robot Pada Sumbu X ($K_p=2350$ dan $K_d=1300$)	57
Gambar 5.5 Validasi dan Error Trajektori Lurus 25 cm	59
Gambar 5.6 Grafik Kecepatan Trajektori Lurus 25 cm	59
Gambar 5.7 Grafik Posisi X-Y Trajektori Lurus 25 cm.....	60
Gambar 5.8 Validasi Trajektori Lurus 83 cm	61
Gambar 5.9 Grafik Kecepatan Trajektori Lurus 83 cm	61
Gambar 5.10 Grafik Posisi Trajektori Lurus 83 cm.....	61
Gambar 5.11 Validasi Trajektori Lurus 300 cm	62
Gambar 5.12 Error yang Terjadi 300 cm.....	62
Gambar 5.13 Grafik Kecepatan Trajektori Lurus 300 cm	63
Gambar 5.14 Grafik Posisi Trajektori Lurus 300 cm.....	63
Gambar 5.15 Gerakan dan Error Trajektori Putar $\pi/2$ rad	65
Gambar 5.16 Posisi Trajektori Putar $\pi/2$ rad	65
Gambar 5.17 Pergerakan Trajektori Putar $\pi/2$ rad.....	65
Gambar 5.18 Posisi Trajektori Putar π rad	66
Gambar 5.19 Gerakan dan Error Trajektori Putar π rad	66
Gambar 5.20 Pergerakan Trajektori Putar π rad.....	67
Gambar 5.21 Pergerakan dan Error Trajektori Putar $3/2 \pi$ rad.....	67
Gambar 5.22 Posisi Trajektori Putar $3/2 \pi$ rad	68
Gambar 5.23 Grafik Posisi untuk 2π rad	68

Gambar 5.24 Posisi Trajektori Putar 2 PI rad	69
Gambar 5.25 Pergerakan dan Eror Trajektori Putar 2 PI rad	69
Gambar 5.26 Grafik Posisi untuk 2 PI rad	69
Gambar 5.27 Eror Trajektori Elip Koordinat (0.5,0.5)	71
Gambar 5.28 Trajektori Elip Pada Koordinat (0.5,0.5).....	71
Gambar 5.29 Grafik Trajektori Elip di Koordinat (0.5,0.5).....	71
Gambar 5.30 Koordinat Spidol Terhadap Robot	72
Gambar 5.31 Eror Trajektori Elip Koordinat (1,2)	73
Gambar 5.32 Trajektori Elip Pada Koordinat (1,2).....	73
Gambar 5.33 Grafik Trajektori Elip di Koordinat (1,2).....	73
Gambar 5.34 Eror Trajektori Elip Koordinat (-0.5,0.5).....	74
Gambar 5.35 Trajektori Elip Pada Koordinat (-0.5,0.5)	74
Gambar 5.36 Grafik Trajektori Elip di Koordinat (-0.5,0.5)	75
Gambar 5.37 Eror Trajektori Elip Koordinat (-1,2).....	75
Gambar 5.38 Trajektori Elip Pada Koordinat (-1,2)	75
Gambar 5.39 Grafik Trajektori Elip di Koordinat (-1,2)	76
Gambar 5.40 Eror Trajektori Navigasi Robot Pada Koordinat (0.5,0.5)	77
Gambar 5.41 Trajektori Navigasi Robot Pada Koordinat (0.5,0.5)	77
Gambar 5.42 Grafik Trajektori Navigasi Robot Pada Koordinat (0.5,0.5).....	77
Gambar 5.43 Eror Trajektori Navigasi Robot Pada Koordinat (1,2)	78
Gambar 5.44 Trajektori Navigasi Robot Pada Koordinat (1,2)	78
Gambar 5.45 Grafik Trajektori Navigasi Robot Pada Koordinat (1,2).....	78
Gambar 5.46 Eror Trajektori Navigasi Robot Pada Koordinat (-0.5,0.5).....	79
Gambar 5.47 Trajektori Navigasi Robot Pada Koordinat (-0.5,0.5).....	79
Gambar 5.48 Grafik Trajektori Navigasi Robot Pada Koordinat (-0.5,0.5)	80
Gambar 5.49 Eror Trajektori Navigasi Robot Pada Koordinat (-1,2).....	80
Gambar 5.50 Trajektori Navigasi Robot Pada Koordinat (-1,2).....	81
Gambar 5.51 Grafik Trajektori Navigasi Robot Pada Koordinat (-0.5,0.5)	81
Gambar 5.52 Trajektori Navigasi Robot Penerima Tamu Dengan SRF-08.....	82
Gambar 5.53 Grafik Trajektori Navigasi Robot Pada Koordinat (-1,1)	82
Gambar 5.54 Grafik Jarak dan Posisi Terhadap Waktu	83

DAFTAR TABEL

Tabel 2.1 Pengalamatan SRF-08.....	13
Tabel 2.2 Spesifikasi SRF-08.....	14
Tabel 2.3 Frekuensi dan Periode PWM	18
Tabel 2.4 Pengujian Variasi Frekuensi PWM.....	19
Tabel 3.1 Kecepatan Robot.....	29
Tabel 3.2 Penggunaan Pin atau Port ATmega2560.....	32
Tabel 5.1 Validasi Kp dan Kd.....	55



BAB 1

PENDAHULUAN

1.1 Latar Belakang

Pada abad ke-21 ini, teknologi robotika sedang akan dikembangkan menuju kecanggihan, hingga saatnya nanti pekerjaan manusia di dunia ini dapat dilakukan oleh sebuah benda yang tak bernyawa, namun dibuat seakan-akan bernyawa yang dikenal sebagai robot. Salah satu tujuan penciptaan robot adalah untuk mempermudah dan menggantikan pekerjaan manusia, terlebih lagi di saat pekerjaan tersebut memiliki ketelitian yang tinggi dan berbahaya bagi manusia. Di masa mendatang sebuah robot tidak hanya diciptakan sebagai sebuah benda yang bisa melakukan pekerjaan manusia, namun robot juga dapat dianggap sebagai seorang teman yang mampu berkomunikasi dengan manusia. Sebagai contoh, saat seseorang akan menghadiri suatu acara tertentu, biasanya terdapat seseorang yang berperan sebagai penerima tamu yang ramah dan murah senyum kepada tamunya, menyapa dan mampu mengarahkan tamu menuju tempat acara. Pada masa mendatang peran tersebut dapat dilakukan oleh robot, yaitu sebuah robot penerima tamu.



Gambar 1.1 Robot Penerima Tamu

Pada dasarnya robot penerima tamu ini tidak hanya digunakan dalam acara-acara tertentu, namun bisa digunakan di tempat-tempat perbelanjaan, sekolah, kantor dan lain sebagainya sehingga mampu menggantikan pekerjaan

manusia dan manusia bisa lebih aktif lagi dalam mengerjakan pekerjaannya yang lain. Dengan demikian, robot penerima tamu dirasakan cukup penting untuk dikembangkan dan diterapkan dalam kehidupan manusia, terutama di Indonesia yang memiliki cukup banyak tempat perbelanjaan, perkantoran dan tempat hiburan.

Perancangan adalah salah satu langkah awal untuk membuat sebuah robot. Dalam hal ini, akan dirancang sebuah robot penerima tamu dengan sistem gerak menggunakan roda yang dibuat menyerupai tank yang digerakkan oleh sebuah gearhead DC motor. Setelah disain selesai, telah dilakukan pengujian pembebanan dengan hasil robot mampu bergerak dengan beban lebih dari 30 Kg.

Gearhead DC motor adalah sebuah motor DC yang telah terintegrasi dengan sistem gear dan sensor encoder, sehingga untuk mengendalikan motor tersebut dibutuhkan sebuah data umpan balik, yaitu data dari sensor encoder. Data dari sensor encoder tersebut nantinya akan diolah menjadi data kecepatan putar motor dalam satuan rad/s, berdasarkan kecepatan itu akan didapatkan jarak jelajah sebagai acuan posisi robot.

Sebuah sistem *mobile robot* dirasakan sangat perlu sebagai sistem navigasi robot. Dengan adanya sistem ini, maka robot akan mengetahui posisinya ketika bernavigasi sehingga akan memudahkan robot untuk berpindah-pindah dari suatu tempat ke tempat yang lain dengan baik.

Tujuan dari skripsi ini adalah membuat suatu sistem navigasi robot yang baik dengan menerapkan metode *mobile robot* pada robot penerima tamu melalui beberapa pengujian dengan trajektori tertentu, yaitu trajektori lurus dan putar sebagai gerakan dasar robot dan membuat trajektori khusus sebagai trajektori navigasi dari robot penerima tamu. Dengan memanfaatkan pengendali PD sebagai kompensator eror posisi robot, diharapkan besarnya eror yang akan terjadi mampu mendekati nilai nol.

1.2 Tujuan dan Kontribusi Riset

Tujuan penulisan ini adalah untuk merancang bangun sistem *tracked mobile robot* yang dapat diterapkan pada robot penerima tamu untuk :

- a. Menghampiri pengunjung atau tamu melalui penerapan sistem navigasi dengan *trajectory planning*.
- b. Menghindari tabrakan dengan benda di sekitar robot seperti dinding dan manusia melalui penggunaan sensor jarak ultrasonik

1.3 Perumusan dan Pembatasan Masalah

Inti permasalahan dalam riset ini adalah membuat sebuah disain *tracked mobile robot* yang handal yang mampu bernavigasi di berbagai jenis lingkungan. Robot akan diuji pada lantai berkeramik karena kebanyakan lantai yang ada menggunakan keramik. Pembacaan data sensor encoder yang tepat dan akurat akan mempermudah robot dalam bernavigasi, begitu pula sebaliknya jika pembacaan data sensor encoder yang kurang tepat dan kurang akurat akan mempersulit robot dalam bernavigasi. Jika untuk pembacaan sensor encoder saja terdapat kesalahan, apalagi jika ditambah slip pada pergerakan, maka kesalahan yang terjadi akan semakin besar. Oleh karena itu, dirancang sistem pembacaan sensor encoder yang lebih akurat, yaitu dengan menggunakan IC HCTL-2032 sebagai pembaca data sensor encoder. Berdasarkan pengujian yang telah dilakukan, pembacaan data sensor encoder dengan menggunakan HCTL-2032 dinilai lebih efektif jika dibandingkan dengan penggunaan fitur *interrupt* pada mikrokontroler. Dengan demikian, kecepatan dan jarak tempuh robot dapat diprediksi akan mendekati jarak sebenarnya.

1.4 Metode Penulisan

Metode penulisan yang digunakan dalam penyusunan laporan skripsi ini antara lain :

1. Studi literatur, yaitu dengan mencari materi dan buku-buku referensi ataupun sumber-sumber lainnya yang berhubungan dengan *service robot*
2. Konsultasi dengan pembimbing
3. Pengujian, yaitu dengan menguji kehandalan robot

1.5 Sistematika Penulisan

Laporan seminar ini terdiri dari enam bab, yaitu :

BAB 1 PENDAHULUAN

Bab ini berisi tentang latar belakang, tujuan dan kontribusi riset, perumusan dan pembatasan masalah dalam skripsi dan metode penulisan skripsi. Bab ini merupakan awal untuk memberikan gambaran umum tentang isi laporan skripsi.

BAB 2 SISTEM ROBOTIKA

Bab ini berisi tentang sistem robotika robot penerima tamu, baik perangkat elektrik maupun perangkat mekanik yang digunakan dalam pembuatan robot penerima tamu.

BAB 3 PERANCANGAN SISTEM ROBOT

Bab ini berisi tentang perancangan sistem robot, yaitu perancangan mekanik dan perancangan sistem kendali robot penerima tamu menggunakan pengendali PD.

BAB 4 *MOBILE ROBOT*

Bab ini berisi tentang penjelasan *mobile robot* dan algoritma yang digunakan pada sistem navigasi robot penerima tamu.

BAB 5 PENGUJIAN

Bab ini berisi tentang metode dan langkah-langkah eksperimen yang digunakan untuk mengambil data dan menganalisis hasil data yang telah didapatkan. Metode pengujian yang dilakukan adalah pengujian pengendali PD dan navigasi robot berdasarkan trajektori tertentu.

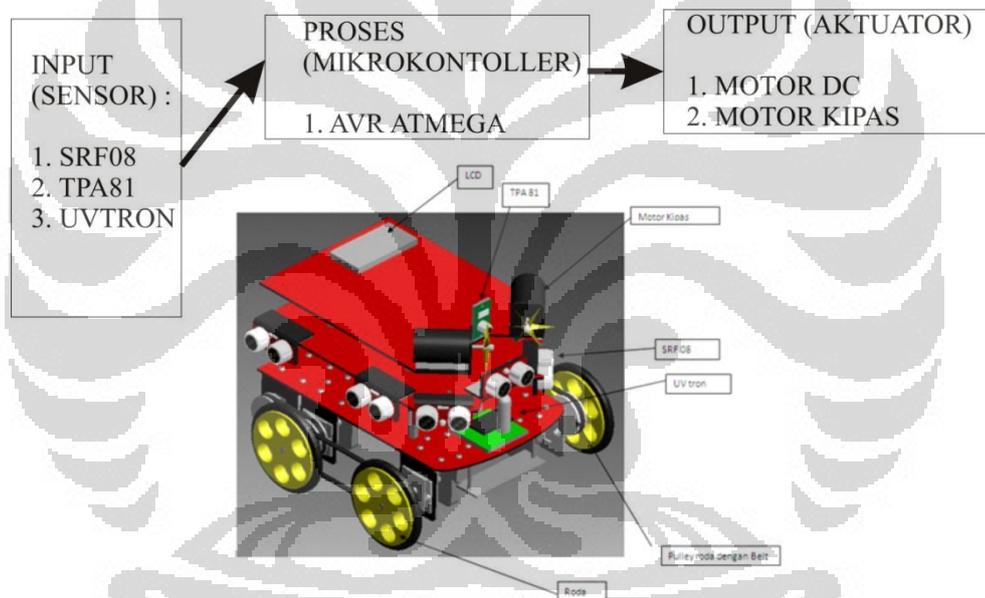
BAB 6 KESIMPULAN

Bab ini berisi tentang kesimpulan dari sistem navigasi robot penerima tamu yang telah dibuat menggunakan metode *mobile robot* dan pengendali PD melalui trajektori tertentu.

BAB 2

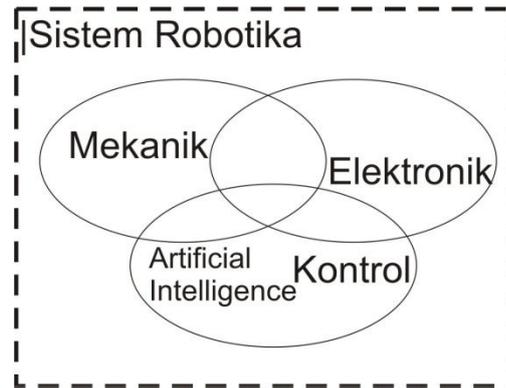
SISTEM ROBOTIKA

Reaksi awal seseorang melihat robot adalah perasaan hebat dan kagum yang sekaligus dicampur dengan rasa ingin tahu bagaimana robot itu bergerak. Pada dasarnya sebuah robot dapat bergerak karena adanya sistem-sistem yang tertanam di dalamnya yang saling bekerja sama membentuk suatu kesatuan sistem yang kompleks yang dinamakan sistem robotika. Sistem tersebut terdiri dari sistem mekanik, sistem elektrik dan sistem perangkat lunak. Perhatikan gambar berikut ini.



Gambar 2.1 Robot Beroda dan Sistem Robotikanya

Gambar 2.1 di atas merupakan gambar disain robot beroda dan sistemnya. Robot tersebut memiliki sistem mekanik yang berperan dalam pembentukan badan robot, sistem elektrik menggerakkan perangkat keras yang ada di robot dan sistem perangkat lunak melakukan pengaturan dan algoritma dalam proses penggerakkan perangkat keras robot (perhatikan gambar 2.2). Perangkat keras robot baik elektrik maupun mekanik saat ini cukup mudah untuk dicari, apalagi dengan adanya *online shopping*.



Gambar 2.2 Sistem Robotika

Dalam perancangan robot penerima tamu digunakan beberapa perangkat keras elektrik dan mekanik. Perangkat keras elektrik yang digunakan adalah mikrokontroler ATmega2560, sensor jarak SRF-08, gearhead DC motor yang terintegrasi dengan sensor jarak encoder, IC HCTL-2032 sebagai pembaca data sensor encoder dan perangkat lainnya yang dijelaskan pada pembahasan berikutnya, sedangkan perangkat keras mekanik yang digunakan adalah aluminium, akrilik, poros, belt dan huben putar sebagai rangka robot. Berikut rancangan perangkat keras elektrik dan mekanik pada disain robot penerima tamu.

2.1 Perangkat Keras Elektrik

Perangkat keras elektrik adalah suatu perangkat sistem robotika yang dalam pengoperasiannya menggunakan dan berhubungan dengan aliran listrik. Perangkat ini akan menentukan pembentukan sistem kontroler pada sistem robotika. Perangkat keras elektrik ini terdiri atas sensor, mikrokontroler, aktuator, perangkat *debugging* dan perangkat tambahan lainnya yang menunjang kehandalan sistem robotika.

2.1.1 Sensor

Data input yang diperlukan oleh robot salah-satunya dapat berasal dari bagian robot itu sendiri, misalnya saja perputaran motornya. Untuk bisa membaca data perputaran motor tersebut, maka diperlukan suatu sensor. Sensor merupakan suatu perangkat yang berfungsi untuk menangkap dan mengkonversi data-data tentang kondisi robot dan kondisi di sekitar robot. Sensor banyak jenisnya

tergantung pemakaian dan keefektifan robot yang akan dibuat. Dari yang paling sederhana seperti sensor biner atau *On/Off* yang menggunakan saklar atau *limit switch*, hingga sensor yang memerlukan algoritma tertentu dalam pengolahan datanya seperti sensor analog dan sensor digital.

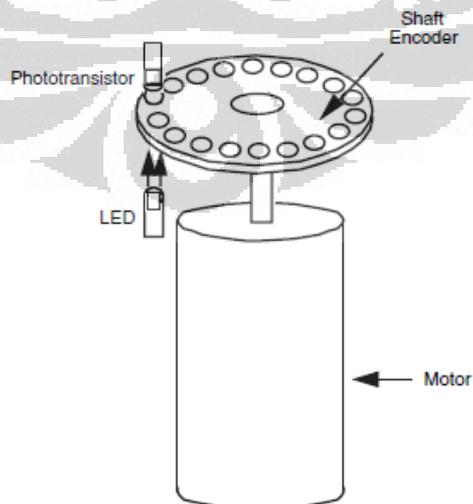
Secara umum, sensor yang digunakan pada robot jenis beroda adalah sensor incremental encoder dan sensor jarak ultrasonik. Kedua sensor tersebut sering digunakan karena harganya yang relatif murah dan saat ini sudah cukup banyak penelitian yang mengarah ke penggunaan kedua sensor tersebut.

2.1.1.1 Sensor Encoder Dengan IC HCTL-2032

Untuk dapat melakukan pengukuran posisi putaran yang lebih presisi, salah satu caranya adalah dengan menggunakan sensor encoder. Sensor encoder ini nantinya akan menghasilkan nilai jarak dan kecepatan setelah dilakukan pengkonversian.

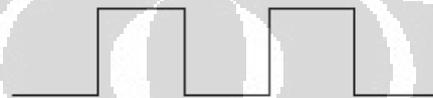
Sensor encoder bekerja dengan menghitung berapa banyak putaran yang telah terjadi pada roda robot, sehingga dengan adanya sensor ini perputaran motor dapat diumpanbalikkan ke sistem untuk dikoreksi kesalahannya dan terbentuklah pengendalian motor secara *close loop*.

Sensor encoder umumnya terdiri dari dua bagian, yaitu sebuah piringan dan sepasang *transmitter* dan *receiver* berupa *light emitting diode (LED)* dan *photo transistor*. Berikut contoh gambar sistem kerja sensor encoder.



Gambar 2.3 Sistem Kerja Sensor Encoder

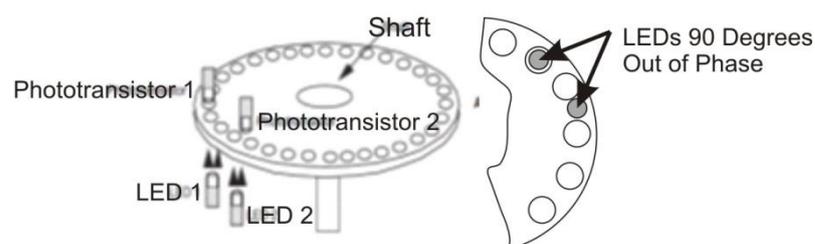
Prinsip kerja sensor encoder adalah ketika *receiver* menerima cahaya dari *transmitter*, maka *receiver* akan mengeluarkan sinyal dengan logika “1” atau “*high*”. Sebaliknya, ketika tidak mendapat cahaya, maka *receiver* tersebut akan mengeluarkan sinyal dengan logika “0” atau “*low*”. Dengan demikian, dalam satu putaran, akan diperoleh pulsa dari *encoder* yang berupa sinyal dengan logika “*high*” dan “*low*” yang muncul bergantian. Penghitungan terhadap pulsa yang dihasilkan akan menghasilkan jumlah putaran yang telah dilakukan. Berikut contoh pulsa keluaran encoder.



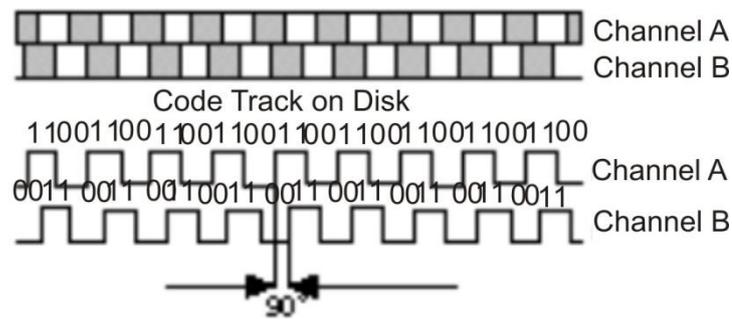
Gambar 2.4 Pulsa Keluaran Sensor Encoder

Jenis lain dari *encoder* ini disebut *quadrature encoder*. Pada *encoder* jenis ini terdapat dua pasang *transmitter-receiver* yang dipasang dengan konfigurasi seperti digambarkan pada gambar 2.5. Dengan konfigurasi ini, sensor *encoder* akan memiliki dua keluaran pulsa yang memiliki beda fasa antara keduanya, seperti dapat dilihat pada gambar 2.4 Dua data ini dapat digunakan untuk menentukan arah pergerakan *mobile robot*.

Sensor encoder yang digunakan adalah Wheel-Encoder TraxsterII, yaitu sensor encoder yang telah terintegrasi dengan motor DC pada gearhead DC motor TraxsterII. Spesifikasi utama sensor encoder ini adalah 624 per output shaft revolution. Sensor encoder ini merupakan sensor *quadrature encoder*. Berikut gambar sistem kerja dan keluaran sinyal yang dihasilkan dari sensor encoder tersebut.



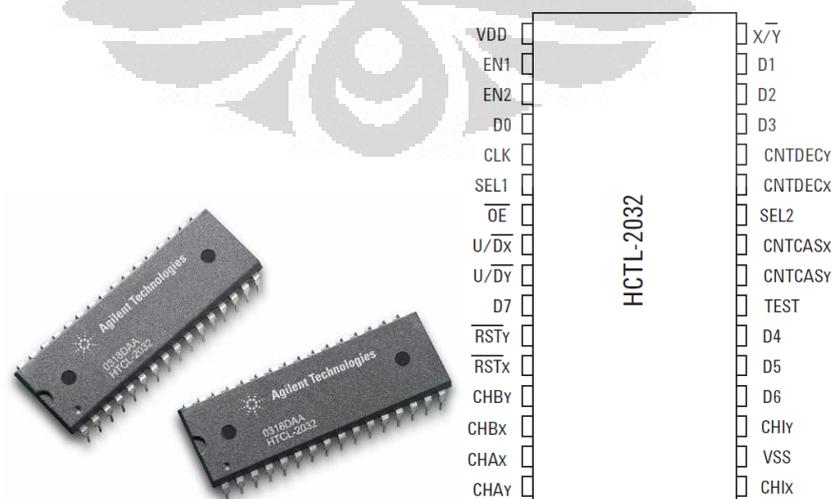
Gambar 2.5 Sistem Kerja Quadrature Encoder



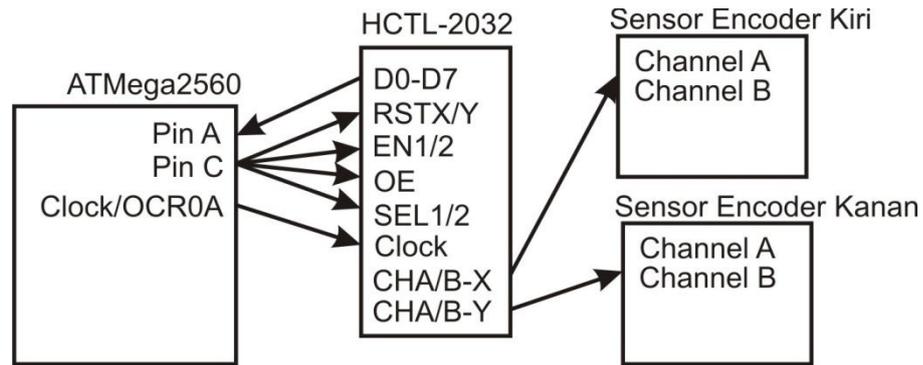
Gambar 2.6 Sistem Kerja Quadrature Encoder

Pengambilan data sensor encoder ini pada dasarnya dapat dilakukan dengan dua cara, yaitu menggunakan pin interrupt pada mikrokontroler dan menggunakan *Integrated Circuit* (IC) sebagai *interface* mikrokontroler dengan sensor encoder melalui sistem *counting*. *Integrated Circuit* yang digunakan adalah HCTL2032.

HCTL-2032 adalah IC CMOS yang memiliki fungsi sebagai *quadrature decoder*, *counter* dan *bus interface*. HCTL-2032 didisain untuk memperbaiki performansi sistem dalam sistem kendali lup tertutup dan sistem input data digital. HCTL-2032 ini diaplikasikan sebagai *interface* atau penghubung sensor encoder dengan ATmega2560. HCTL-2032 ini mempunyai kemampuan dual axis yang membuat HCTL-2032 bisa digunakan untuk dua buah sensor encoder dan juga mendukung *channel index* pada sensor encoder tiga fasa. Selain itu, HCTL-2032 juga bisa diprogram dengan mode perhitungan 1x/2x/4x. Berikut gambar IC HCTL-2032.



Gambar 2.7 HCTL-2032 dan Pinoutnya



Gambar 2.8 Penggunaan HCTL-2032

Pada pendisainannya, metode interrupt digunakan 4 buah interrupt, yaitu interrupt 4, 5, 6 dan 7 (pin E4, E5, E6 dan E7), dimana setiap motor mendapatkan input dua buah interrupt sebagai masukan ke channel A dan channel B pada sensor interrupt. Sedangkan pada penggunaan HCTL-2032, pin-pin yang digunakan adalah :

- a. Pin A : HCTL_DATA
- b. PortC.0 : HCTL_XY
- c. PortC.1 : HCTL_RSTY
- d. PortC.2 : HCTL_RSTX
- e. PortC.3 : HCTL_OE
- f. PortC.4 : HCTL_EN2
- g. PortC.5 : HCTL_EN1
- h. PortC.6 : HCTL_SEL2
- i. PortC.7 : HCTL_SEL1

Mode yang digunakan adalah double axis karena 1 HCTL-2032 akan digunakan untuk membaca 2 sensor encoder. Mode counting dari encodernya adalah mode 4X.

Algoritma penggunaan HCTL-2032 ini adalah :

- a. Inisialisasi Pin pada mikrokontroller
- b. Inisialisasi nilai pin EN1 dan EN2 untuk menentukan penggunaan mode counter pada encoder
- c. Inisialisasi nilai RSTX dan RSTY untuk mengaktifkan mode reset HCTL
- d. Inisialisasi awal nilai OE dengan logika 1

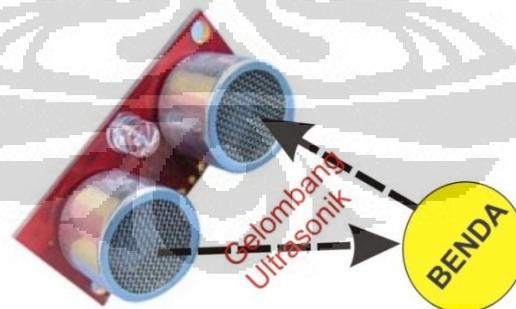
e. Pembacaan counter encoder

Pembacaan dilakukan dengan membaca empat nilai bit yang berasal dari pin Sel 1, Sel 2 dan OE pada setiap pembacaan bit. Urutan pembacaan bitnya adalah bit MSB, bit kedua, bit ketiga dan bit LSB. Pengesetan nilai bisa dilihat di datasheet HCTL-2032.

Berdasarkan percobaan yang telah dilakukan pada kedua metode tersebut dapat disimpulkan bahwa pada dasarnya penggunaan kedua metode tersebut sebagai pembacaan nilai sensor encoder adalah sama baiknya, namun ada pertimbangan lain, yaitu kestabilan nilai yang didapatkan, sehingga disimpulkan bahwa pengambilan data sensor encoder dengan menggunakan HCTL-2032 memiliki performansi lebih baik dalam pengambilan data sensor encoder, yaitu dalam hal kestabilan posisi dari nilai sensor encoder.

2.1.1.2 Sensor Ultrasonik SRF-08

Sensor jenis ini dikategorikan dalam sensor jarak. Dalam pendeteksian jaraknya sensor ini menggunakan gelombang ultrasonik yang dihasilkan oleh sensor ini. Sensor jarak ini memiliki dua buah bagian, yaitu *transmitter* pada sisi kirim dan *receiver* pada sisi terima. Prinsip kerjanya adalah gelombang ultrasonik akan dipancarkan oleh *transmitter* selama periode waktu tertentu yang kemudian gelombang ultrasonik tersebut akan diterima oleh *receiver*. Berikut ilustrasinya.



Gambar 2.9 Prinsip Kerja SRF-08

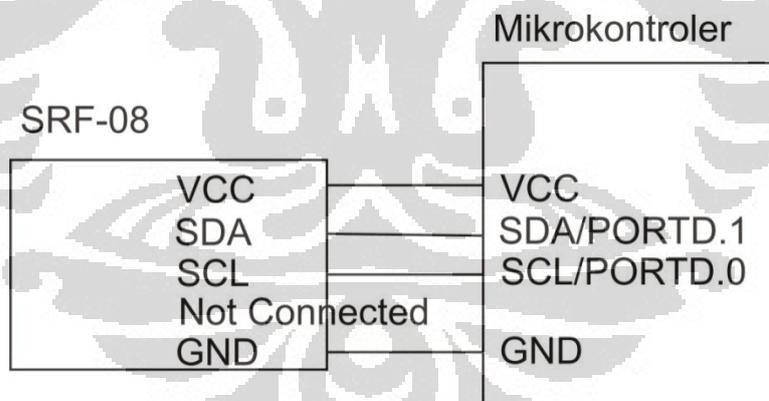
Data keluaran dari sensor ini berupa logika 0 atau 1 yang kemudian berdasarkan *datasheet* SRF-08 dapat dihitung besarnya jarak benda terhadap sensor ini. Perhitungan jaraknya adalah dengan mengalikan kecepatan gelombang ultrasonik dengan waktu tempuh dan kemudian dibagi 2 akibat adanya waktu

gelombang ultrasonik dipantulkan ke *receiver* yang besarnya sama dengan waktu gelombang ultrasonik menyentuh benda.

$$Jarak = \frac{Kecepatan\ Gelombang\ Ultrasonik \times Waktu\ Tempuh}{2} \quad (2.1)$$

, dengan kecepatan gelombang ultrasonik di udara adalah 344 m/s.

Sensor ini memiliki performansi yang cukup baik, yaitu mampu mengukur jarak sepanjang 3 cm sampai 6 m dan sensor ini sudah cukup banyak digunakan dalam berbagai aplikasi robotika. Komunikasi sensor ini dengan mikrokontroler dilakukan melalui komunikasi *two wire*, yaitu komunikasi I2C (*Inter Integrated Circuit*), yaitu suatu standar protokol sistem komunikasi data serial yang dikembangkan oleh Philips dengan mengirimkan data secara serial melalui sebuah jalur data dua arah. Karena menggunakan jalur data I2C, maka hanya memerlukan dua buah pin saja untuk berkomunikasi, yaitu pin untuk *data* (SDA) dan pin untuk *sinyal clock* (SCL). Sistem komunikasi ini cukup terkenal karena penggunaannya yang mudah dan mampu menghemat penggunaan pin-pin pada mikrokontroler.



Gambar 2.10 Diagram Komunikasi I2C SRF-08

SDA merupakan jalur data pada komunikasi I2C sedangkan SCL merupakan jalur *clock* dimana sinyal *clock* akan selalu muncul untuk setiap pengiriman bit. Berikut pin-pin SRF-08 yang terkoneksi dengan mikrokontroler.



Gambar 2.11 Pin Koneksi SRF-08

Alamat standar dari SRF-08 adalah 0xE0. Alamat tersebut dapat diganti ke 16 alamat, yaitu E0, E2, E4, E6, E8, EA, EC, EE, F0, F2, F4, F6, F8, FA, FC atau FE sehingga jumlah maksimum penggunaan I2C adalah 16 SRF-08. Alamat-alamat yang telah diinisialisasi pada SRF-08 dapat diketahui secara visual, yaitu melalui nyala LED yang ditimbulkan oleh SRF-08. Perinciannya sebagai berikut.

Tabel 2.1 Pengalamatan SRF-08

Address		Long Flash	Short Flasher
Decimal	Hex		
224	E0	1	0
226	E2	1	1
228	E4	1	2
230	E6	1	3
232	E8	1	4
234	EA	1	5
236	EC	1	6
238	EE	1	7
240	F0	1	8
242	F2	1	9
244	F4	1	10
246	F6	1	11
248	F8	1	12

250	FA	1	13
252	FC	1	14
254	FE	1	15

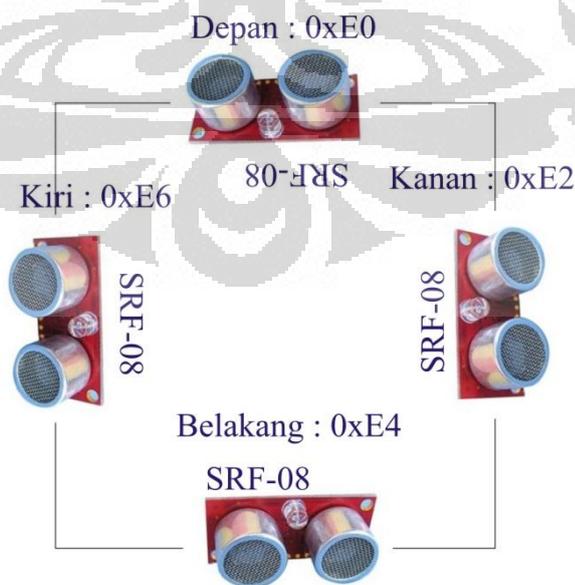
Spesifikasi lengkap untuk SRF-08 dijabarkan dalam tabel berikut ini :

Tabel 2.2 Spesifikasi SRF-08

Beam Pattern	<p style="text-align: center;">Beam Pattern</p> <p style="text-align: center;">Gambar 2.12 Pola Pembacaan SRF-08</p>
Voltage	5 v
Current	15mA typ. 3 mA standby
Frequency	40 KHz
Max. Range	6 m
Min. Range	3 cm
Max. Analogue Gain	Variable to 1025 in 32 step
Connection	I2C Bus
Light Sensor	Front Facing Light Sensor
Timing	Fully timed echo, freeing host computer of task
Echo	Multiple echo-keeps looking after first echo
Units	Range reported n uS, mm or inches
Weight	0.4 oz.
Size	43 mm x 20 mm x 17 mm (w x d x h)

Jarak maksimum yang dapat dideteksi oleh SRF-08 diset oleh timer internal yang secara standarnya diset sebesar 65 milisekon atau setara dengan jarak 11 meter. Jarak ini dinilai terlalu jauh dari kemampuan SRF-08 yang mampu mendeteksi hingga 6 meter. Hal ini dapat mengurangi besarnya waktu *echo* dari SRF-08 dan jarak yang ditulis di register jarak. Besarnya jarak maksimum dapat diset pada jarak 43 mm hingga 11 meter. Besarnya jarak adalah $((Range\ Register \times 43\ mm) + 43\ mm)$, sehingga untuk mengatur nilai *Range Register* dengan nilai 0 (0x00) yang akan memberikan jarak maksimum sebesar 43 mm. Pengaturan nilai *Range Register* ke nilai 1 (0x01) akan menghasilkan jarak maksimum sebesar 86 mm. Selanjutnya untuk menghasilkan jarak maksimum sebesar 1 meter, maka *Range Register* diset pada nilai 24 (0x18), nilai 140 (0x8C) untuk menghasilkan jarak maksimum 6 meter dan nilai 255 (0xFF) untuk mendapatkan jarak maksimum 11 meter. Untuk robot penerima tamu ini, sensor SRF-08 dirancang dengan jarak maksimum 100 cm. Jarak 100 cm dirasakan cukup untuk mengantisipasi tabrakan dan untuk memperkecil waktu pembacaan SRF-08 sehingga didapatkan *update* data sensor yang lebih banyak di setiap pergerakan robot.

Susunan pemasangan dan pengalamatan SRF-08 pada robot adalah sebagai berikut :



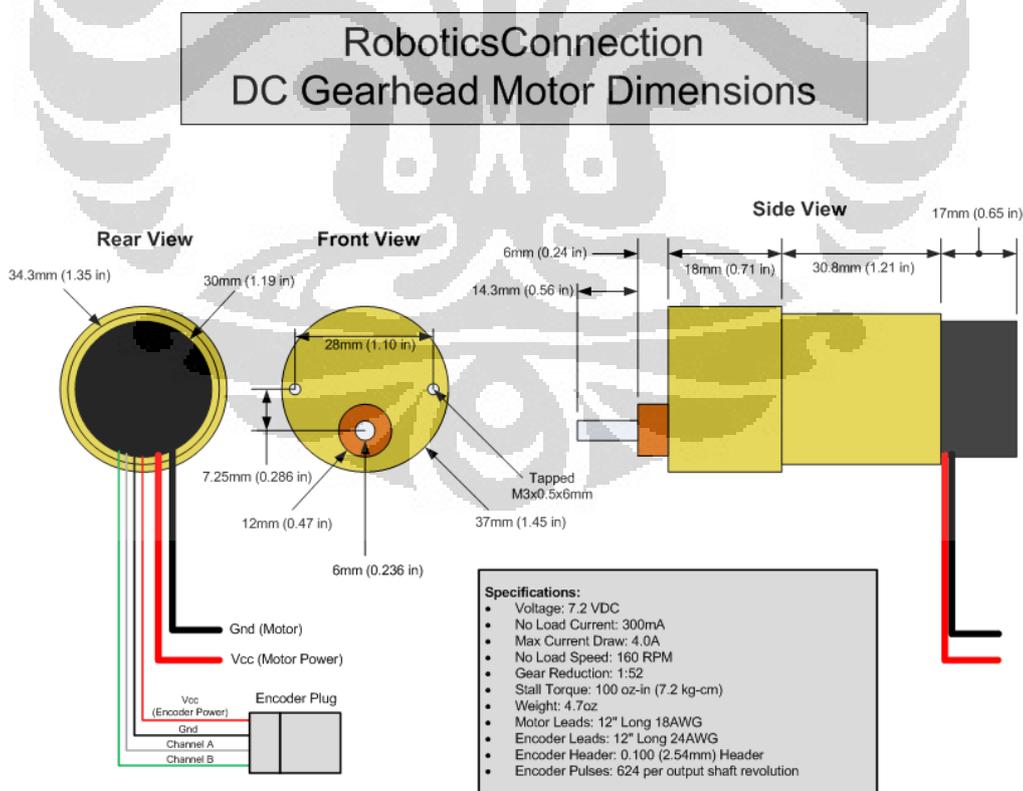
Gambar 2.13 Pemasangan dan Pengalamatan SRF-08

2.1.2 Aktuator

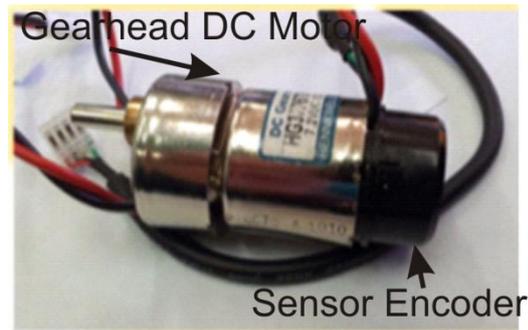
Aktuator merupakan perangkat penggerak robot dan biasanya ditempatkan sebagai keluaran dari sistem. Pada disain robot beroda ini terdapat dua buah aktuator yang digunakan, yaitu brush DC motor TraxsterII. Motor jenis ini dinilai lebih efisien karena dalam satu set motor tersebut sudah terdapat sensor encodernya dan sistem gear motor.

2.1.2.1 Gearhead DC Motor

Gearhead DC Motor merupakan sebuah motor DC yang menggunakan brush dan terintegrasi langsung dengan sistem gear dengan rasio 1 : 52 dan sebuah sensor encoder dengan dua channel. Brushed Motor DC ini dipilih karena sistem pengontrolannya yang relative mudah, namun kelemahannya adalah tingginya tingkat maintenance motor, terutama pada bagian brushnya yang cepat habis. Gambar dan spesifikasi dari gearhead DC motor tersebut adalah sebagai berikut :



Gambar 2.14 Spesifikasi Gearhead DC Motor



Gambar 2.15 Gearhead DC Motor

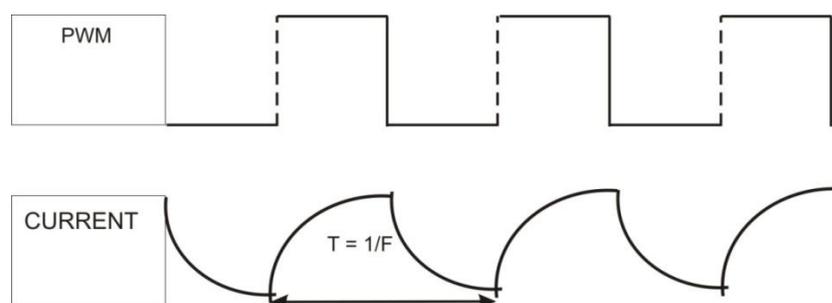
Gearhead DC Motor ini digerakkan dengan sebuah rangkaian H-Bridge yang menggunakan IRF540 dan IRF9540. Untuk mengendalikan kecepatan, digunakan metode PWM (*Pulse Width Modulation*). Pada metode ini, sinyal persegi (*square wave*) dengan periode dan lebar pulsa tertentu diaplikasikan pada motor. Nilai tegangan rata-rata yang diterima pada motor ditentukan oleh rasio lebar pulsa waktu bernilai “*High*” terhadap periode. Dengan kata lain, mengubah rasio lebar pulsa terhadap periode akan mengubah tegangan yang diberikan pada motor, yang pada akhirnya akan mempengaruhi kecepatan motor.

2.1.2.2 Pengendalian Brushed DC Motor

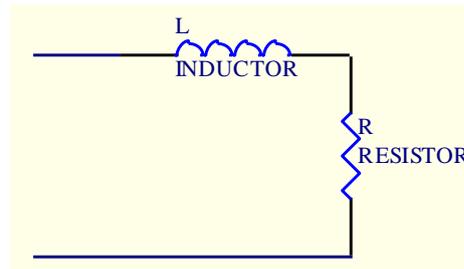
Gearhead DC Motor ini akan dikendalikan secara digital, yaitu menggunakan metode PWM. Metode PWM digunakan karena metode ini akan menjaga amplitudo tegangan pada level yang sama.

Besarnya frekuensi PWM yang digunakan untuk menggerakkan motor DC dapat dicari dengan persamaan :

$$\text{Frekuensi PWM} = \frac{\text{Frekuensi Clock}}{N(1 + \text{Top OCR})} \quad (2.2)$$



Gambar 2.16 Sinyal PWM



Gambar 2.17 Rangkaian Pengganti Motor

Berdasarkan gambar 2.17, dapat dicari besarnya frekuensi minimum untuk motor DC berdasarkan persamaan daya motor tersebut, yaitu :

$$P = \frac{(2I)^2 R}{2} + \frac{0^2 R}{2} \quad (2.3)$$

$$P = 2I^2 R \quad (2.4)$$

$$I e^{-\frac{TR}{2L}} = \left(1 - \frac{P}{100}\right) I e^0 \quad (2.5)$$

$$e^{-\frac{TR}{2L}} = \left(1 - \frac{P}{100}\right) \quad (2.6)$$

$$-\frac{TR}{2L} = \ln\left(1 - \frac{P}{100}\right) \quad (2.7)$$

$$T = \frac{-2L}{R} \ln\left(1 - \frac{P}{100}\right) \quad (2.8)$$

$$f = 1/T \quad (2.9)$$

Untuk menggerakkan motor DC digunakan Timer 4 dengan panjang 10 bit (1023). Sistem minimum robot menggunakan Clock eksternal 11.0592 MHz. Berdasarkan pengujian pada PWM sebesar 10% sampai 100% yang dilakukan pada clock value sebagai berikut :

Tabel 2.3 Frekuensi dan Periode PWM

No	Clock Value (kHz)	Prescaler	Frekuensi PWM (Hz)	Periode PWM (s)
1	11059.2	1	10800	0.0000926
2	1382.4	8	1350	0.00075
3	172.8	64	168.75	0.006

4	43.2	256	42.1875	0.0237
5	10.8	1024	10.547	0.09

Tabel 2.4 Pengujian Variasi Frekuensi PWM

No	Frekuensi PWM (Hz)	10	20	30	40	50	60	70	80	90	100
1	10800	X	X	X	X	X	X	X	X	X	X
2	1350	X	X	X	X	X	X	X	X	X	O
3	168.75	X	X	X	X	X	X	X	O	O	O
4	42.1875	X	X	X	X	X	O	O	O	O	O
5	10.547	X	O	O	O	O	O	O	O	O	O

Keterangan : X = Tidak Bergerak

O = Bergerak

Berdasarkan data pengujian tersebut didapatkan data bahwa terdapat dua nilai *clock value* yang memungkinkan digunakan, yaitu saat 43.2 kHz dan 10.8 kHz. Pertimbangan yang digunakan dalam pemilihan *clock value* adalah waktu periode yang kecil dan gerakan robot yang halus. Akhirnya pilihan jatuh pada *clock value* 10.8 kHz, karena gerakan dan akselerasi yang dihasilkan lebih stabil dan akan sangat membantu pergerakan robot saat beban bertambah, selain itu besarnya periode antara kedua frekuensi PWM tersebut tidak jauh berbeda, yaitu 0.2 μ s untuk frekuensi 43.2 kHz dan 0.9 μ s untuk frekuensi 10.8 kHz. Tujuannya untuk mendapatkan gerakan robot dengan akselerasi baik.

2.1.3 Mikrokontroler

Mikrokontroler adalah sebuah perangkat proses pada robot yang merupakan otak dari robot. Mikrokontroler ini berfungsi sebagai unit pengendali utama pada robot. Mikrokontroler yang digunakan adalah mikrokontroler jenis AVR dengan tipe ATmega2560. Fitur-fitur yang dimiliki oleh mikrokontroler ini yang dimanfaatkan oleh robot ini antara lain adalah :

- a. *PWM generator* yang berfungsi menghasilkan pulsa hasil modulasi dan digunakan untuk menggerakkan *gearhead DC motor*.
- b. *Interrupt system* yang digunakan untuk merespon pemicu dari luar.

- c. *Timer* yang digunakan untuk penghitungan waktu yang akurat. Kecepatan maksimum dari mikrokontroler ini adalah sebesar 16 MIP (*Million Instructions Per Second*) yang menjamin kecepatan dalam pengekseskuan instruksi maupun dalam merespon masukan. Ada dua buah *Timer* yang digunakan, yaitu *Timer* 8 bit dan *Timer* 16 bit.
- d. Port I/O untuk memberikan inisialisasi input dan output dari perangkat lain.
- e. I2C bus yang digunakan untuk komunikasi *two wires*, yaitu berupa pin SDA dan SCL yang digunakan untuk jalur komunikasi antara SRF-08 dengan mikrokontroler.



Gambar 2.18 Mikrokontroler ATmega2560

Mikrokontroler ini memiliki 100 pin yang terdiri atas pin-pin dengan fitur-fitur tertentu. Hal ini cukup memungkinkan penggunaan perangkat yang lebih banyak pada sistem robotika yang akan dibuat. Berikut gambar pinout pada ATMEGA2560 :

2.1.4 Media Debugging

Media debugging digunakan untuk melihat data-data sensor dan sinyal saat pengujian. Media debugging yang digunakan pada robot ini adalah kabel serial USB. Hal ini memungkinkan proses *debugging* melalui personal komputer. Komponennya terdiri dari rangkaian PC DB89 dan sebuah IC MAX232.

2.1.4.1 PC DB89

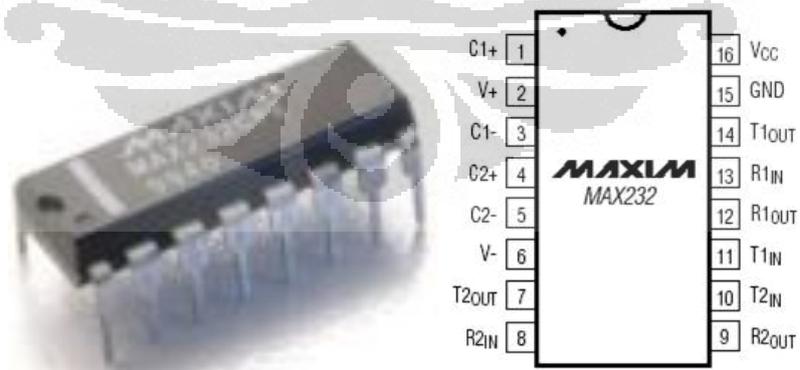
PC DB89 ini adalah sebuah konektor serial yang terdiri dari konektor *female* dan *male*. Berikut gambar PC DB89.



Gambar 2.19 Konektor PC DB89 Male dan Female

2.1.4.2 IC MAX232

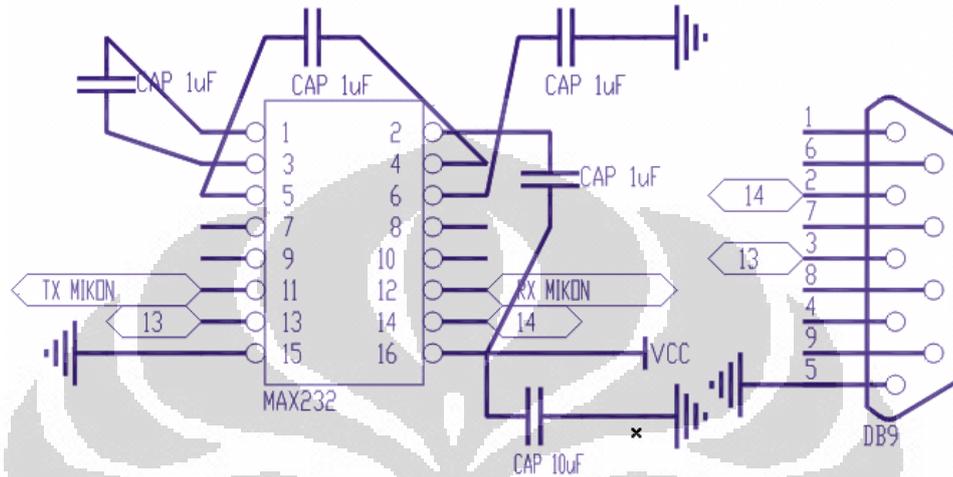
IC MAX232 adalah IC rangkaian antar muka dual RS-232 transmitter/receiver yang memenuhi standar EIA-232-E. IC MAX232 hanya membutuhkan power supply 5V (single power supply) sebagai catu. IC MAX232 berfungsi untuk merubah level tegangan pada COM komputer menjadi level tegangan TTL/CMOS. IC MAX232 terdiri atas tiga bagian yaitu dual charge-pump voltage converter, driver RS232, dan receiver RS232. Berikut gambar IC MAX232 dan pinout IC MAX232.



Gambar 2.20 IC MAX232 dan Pinout

IC MAX232 memiliki dua charge-pump internal yang berfungsi sebagai konverter tegangan +5V menjadi $\pm 10V$ saat tanpa beban untuk operasi driver

RS232. Konverter pertama menggunakan kapasitor C1 untuk menggandakan tegangan input +5V menjadi +10V saat C3 berada pada output V+. Konverter kedua menggunakan kapasitor C2 untuk merubah +10V menjadi -10V saat C4 berada pada output V-. Berikut skematik rangkaian media komunikasi serialnya.



Gambar 2.21 Skematik Komunikasi Serial

2.2 Perangkat Keras Mekanik

2.2.1 Aluminium

Aluminium merupakan bahan yang cukup kuat dan ringan. Berdasarkan penggunaannya pada robot, jenis aluminium dibagi menjadi tiga, yaitu :

a. Aluminium kotak

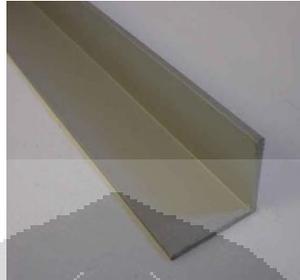
Aluminium kotak digunakan untuk membentuk kerangka dari robot. Ukuran aluminium kotak yang digunakan ada dua macam, yaitu 1x1 inchi dan 2x1cm untuk masing-masing lebar dan tingginya. Berikut gambar aluminium kotak.



Gambar 2.22 Aluminium Kotak

b. Aluminium siku

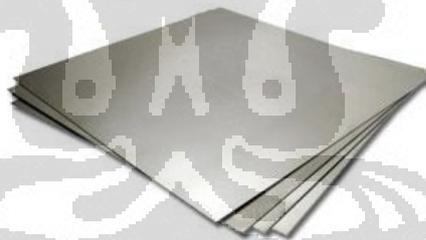
Aluminium siku digunakan untuk menggabungkan dan menahan sambungan antar aluminium kotak akan digabungkan secara siku. Ukurannya adalah 1x1 inchi untuk lebar dan tingginya. Berikut gambar aluminium siku.



Gambar 2.23 Aluminium Siku

c. Aluminium plat

Aluminium plat digunakan untuk menyambung dan menahan sambungan aluminium kotak yang disambung baik secara siku maupun tidak siku. Ukuran panjang dan lebarnya disesuaikan sesuai dengan derajat dan kebutuhan sambungan. Berikut gambar aluminium plat.



Gambar 2.24 Aluminium Plat

2.2.2 Akrilik

Akrilik merupakan bahan seperti plastik yang menyerupai kaca yang merupakan campuran resin dan bahan kimia tertentu. Ketebalan akrilik yang digunakan dalam robot ini ada 4 ukuran, yaitu tebal 2 mm, 3 mm, 5 mm dan 10 mm. Akrilik ketebalan 2 mm digunakan sebagai lapisan penutup roda untuk menahan belt pada roda. Akrilik ketebalan 3 mm digunakan untuk membentuk penutup badan robot, pembentuk lapisan atau layer untuk komponen papan-papan

elektrik dan tempat sensor SRF-08. Akrilik ketebalan 5 mm digunakan untuk layer base robot yang merupakan tempat diletakkannya motor DC dan sistem transmisi robot. Adapun akrilik dengan ketebalan 10 mm digunakan untuk membuat rumah bearing, dudukan motor, dudukan poros dan roda. Berikut contoh akrilik.



Gambar 2.25 Akrilik Dalam Beraneka Warna

2.2.3 Bearing

Bearing merupakan alat untuk menumpu poros dan memberi kemungkinan poros berputar bersamaanya atau berputar padanya. Berikut adalah gambar bagian-bagian bearing.



Gambar 2.26 Bagian Bearing

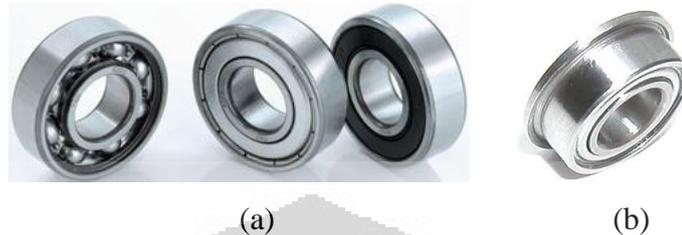
Bearing yang digunakan pada robot ini memiliki diameter dalam 8 mm dan diameter luar 16 mm dengan dua jenis, yaitu :

- a. Bearing standar

Bearing ini digunakan pada tengah roda bebas yang akan dibuat. Bearing ini akan mempermudah perputaran roda bebas.

b. Bearing kuping

Bearing kuping merupakan bearing standar yang memiliki pengait untuk menahan bearing pada rumah bearing. Bearing ini digunakan pada penahan poros dan sisi pinggir roda.



Gambar 2.27 (a) Bearing Standar (b) Bearing kuping

2.2.4 Belt

Belt merupakan material yang digunakan untuk menarik *pulley*. Belt ini digunakan untuk sistem *track* pada robot yang membuat robot seperti tank. Selain itu belt juga digunakan pada sistem transmisi robot, yaitu pada sistem gear. Dalam robot ini terdapat dua ukuran belt yang digunakan, yaitu :

- a. Belt 110 XL yang digunakan pada sistem gear
- b. Belt 320 XL yang digunakan pada sistem perodaan



Gambar 2.28 Belt

Keliling sebuah belt dapat diketahui menurut rumus berikut:

$$\text{Keliling Belt} = (\text{Ukuran Belt}/10) \text{ inchi} = (\text{Ukuran Belt}/10) \times 2.51 \text{ cm} \quad (2.10)$$

Jadi, panjang belt untuk 110XL adalah 276.1 mm dan panjang belt untuk 320XL adalah 802.3 mm.

2.2.5 Poros

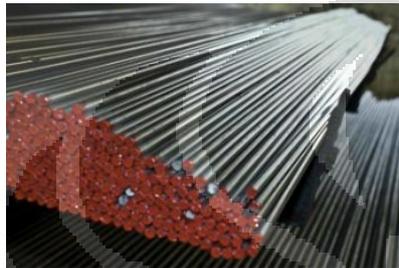
Poros merupakan sebuah besi silinder. Poros digunakan pada sistem transmisi robot. Ada dua jenis poros yang digunakan, yaitu :

a. Poros standar

Poros standar digunakan pada roda bebas dan roda *steering*. Diameternya adalah 8 mm.

b. Poros berulir

Poros berulir merupakan poros standar berulir yang digunakan untuk membuat poros penahan belt pada sistem perodaan. Diameter poros yang digunakan adalah 8 mm.



(a)



(b)

Gambar 2.29 (a) Poros Standar (b) Poros Berulir

2.2.6 Huben Putar

Huben putar merupakan sebuah perangkat mekanik yang berbentuk lingkaran untuk memutar badan robot. Disain badan robot yang dibuat diharapkan dapat berputar, oleh karena itu dalam disainnya digunakan komponen Huben, yaitu sebuah sistem mekanik yang bekerja seperti prinsip bearing, dimana terdapat dua lingkaran yang terhubung melalui barisan bola-bola besi yang jika salah satu lingkaran dibuat statik, maka lingkaran yang satu lagi akan bergerak bebas.



Gambar 2.30 Huben Putar

BAB 3

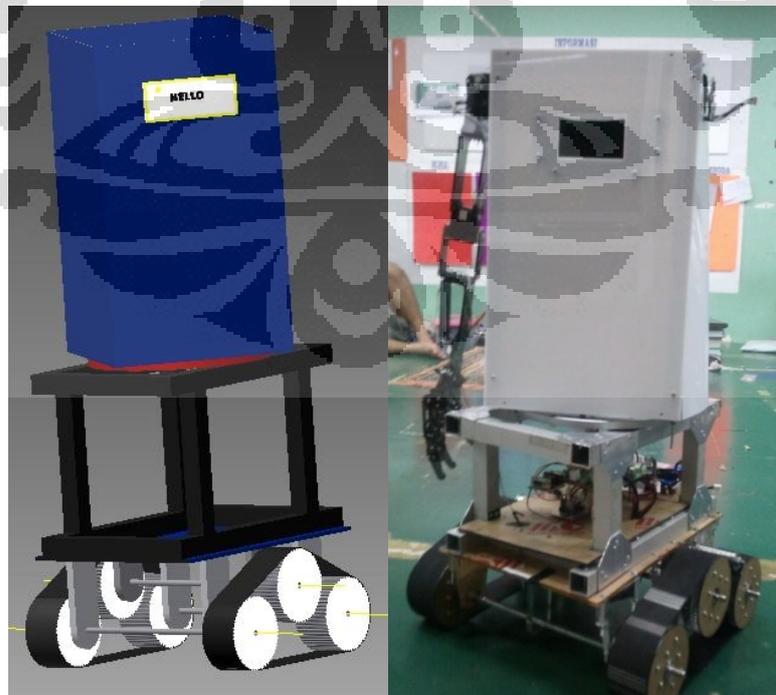
PERANCANGAN SISTEM ROBOT

Dalam menciptakan sebuah robot yang sesuai dengan tujuan pembuatannya, maka ada baiknya dilakukan perancangan robot terlebih dahulu. Dengan adanya perancangan, maka kemungkinan masalah yang akan dihadapi dapat dibayangkan dan dikompensasi. Terlebih lagi jika robot tersebut akan terus dikembangkan, maka perancangan sangatlah diperlukan.

Dalam merancang sebuah robot ada beberapa hal yang harus dipikirkan, yaitu tujuan dan spesifikasi dari robot tersebut, sehingga dalam merancang sebuah sistem robotika tidak terlalu kesulitan. Beberapa tahapan perancangan yang dilakukan dalam pembuatan robot adalah perancangan sistem kendali robot dan perancangan mekanik robot.

3.1 Perancangan Mekanik Robot

Perancangan yang telah dilakukan adalah pada sistem base robot penerima tamu dan sistem mekanik badan robot. Sistem base robot terdiri dari sistem transmisi dan perodaan. Berikut ini hasil perancangan disain robot yang telah dilakukan.

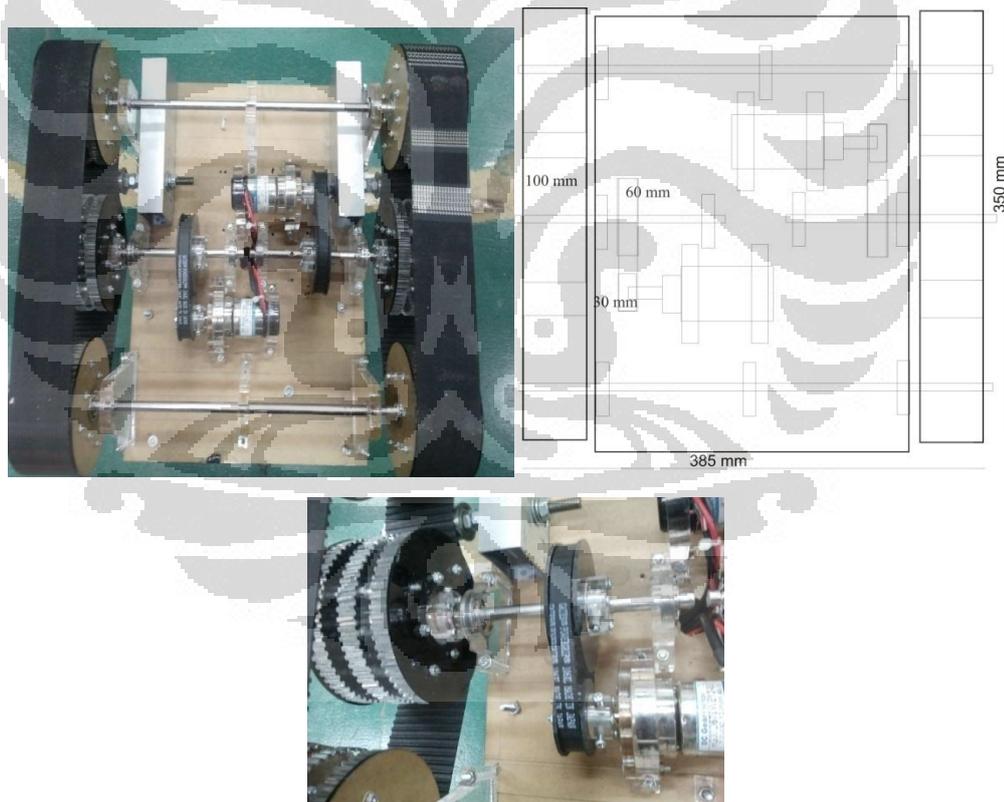


Gambar 3.1 Disain Mekanik Robot Tampak Isometris

Robot ini memiliki dimensi panjang, lebar dan tinggi hingga badan bagian atas berukuran masing-masing 350 mm x 430 mm x 1000 mm dengan berat robot sekitar 10 Kg. Bahan dan material pembuatannya secara umum adalah terbuat dari akrilik, aluminium, belt, PVC, poros besi dan poros ulir besi.

3.1.1 Sistem Transmisi

Sistem transmisi roda menggunakan sistem gear. Hal ini dilakukan untuk meningkatkan torsi motor dan pada dasarnya robot ini tidak didisain untuk bergerak terlalu cepat. Gear 30 mm dihubungkan dengan poros motor dengan sistem penguncian baut, kemudian dihubungkan dengan gear 60 mm menggunakan sistem belting. Penguncian gear ke poros juga menggunakan penguncian baut. Berikut gambar disainnya.



Gambar 3.2 Disain Rancangan Sistem Transmisi

Secara konvensional, gearhead DC motor telah memiliki rasio gear sebesar 1:52 dengan torsi 7.2 Kg/cm. Dengan digunakan sistem transmisi tersebut diharapkan torsi motor akan menjadi lebih besar, yaitu sebesar rasio gear dari

motornya. Robot ini didisain dengan peningkatan torsi sebesar 2 kali torsi sebelumnya.

$$\text{Peningkatan Torsi} = \text{Diameter Gear Akhir} / \text{Diameter Gear Awal} \quad (3.1)$$

$$\text{Peningkatan Torsi} = \frac{6}{3} = 2 \text{ kali torsi awal} \quad (3.2)$$

Berdasarkan pengujian pembebanan yang telah dilakukan didapatkan bahwa robot mampu menahan beban lebih dari 32 Kg termasuk bobot robot dan memiliki kecepatan normal (hanya beban robot) sebesar 0.15 m/s saat PWM 20% dan 0.35 m/s saat PWM 100%.

Tabel 3.1 Kecepatan Robot

PWM	KECEPATAN JELAJAH SEJAUH 3 METER (m/s)
10	0
20	0.014968
30	0.100538
40	0.1777251
50	0.2365931
60	0.2654821
70	0.2838296
80	0.307377
90	0.3324658
100	0.3592814

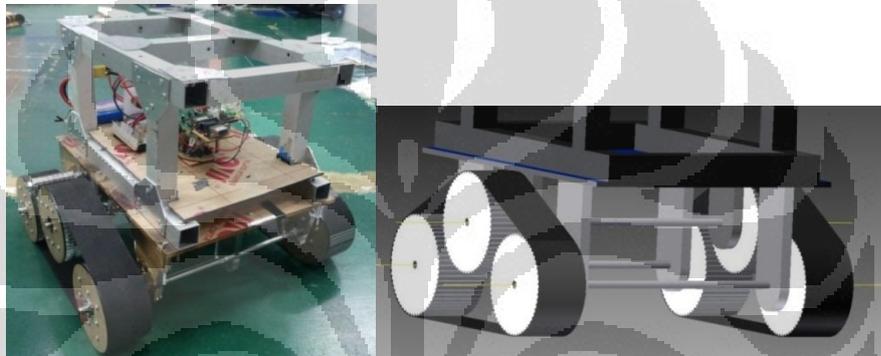
3.1.2 Sistem Perodaan

Sistem ini menggunakan tiga buah roda yang saling berhubungan, dimana satu roda sebagai *steering* dan dua roda lainnya merupakan roda bebas. Salah satu ciri khas robot ini adalah disain rodanya yang menggunakan lapisan-lapisan akrilik dan terdapat pengencang belt. Roda yang dibuat, baik roda *steering* maupun roda bebas, memiliki diameter roda, yaitu 100 mm. Ketiga roda saling terhubung dengan menggunakan sistem *belting*. Jadi, sistem kerja rodanya adalah jika roda *steering* berputar, maka roda *steering* tersebut akan menarik kedua roda bebas, sehingga robot dapat berjalan.



Gambar 3.3 Roda dan Pengencang Belt

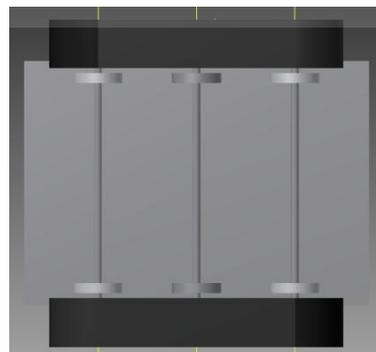
Berikut penggambaran sistem perodaan :



Gambar 3.4 Sistem Perodaan Tampak Isometris



Gambar 3.5 Disain Rancangan Sistem Perodaan Tampak Samping



Gambar 3.6 Disain Sistem Perodaan Tampak Bawah

3.1.3 Sistem Badan

Disain badan robot yang dibuat diharapkan dapat berputar, oleh karena itu dalam disainnya digunakan komponen huben putar, yaitu sebuah sistem mekanik yang bekerja seperti prinsip bearing, dimana terdapat dua lingkaran yang terhubung melalui barisan bola-bola besi yang jika salah satu lingkaran dibuat statik, maka lingkaran yang satu lagi akan bebas.



Gambar 3.7 Disain Badan Robot



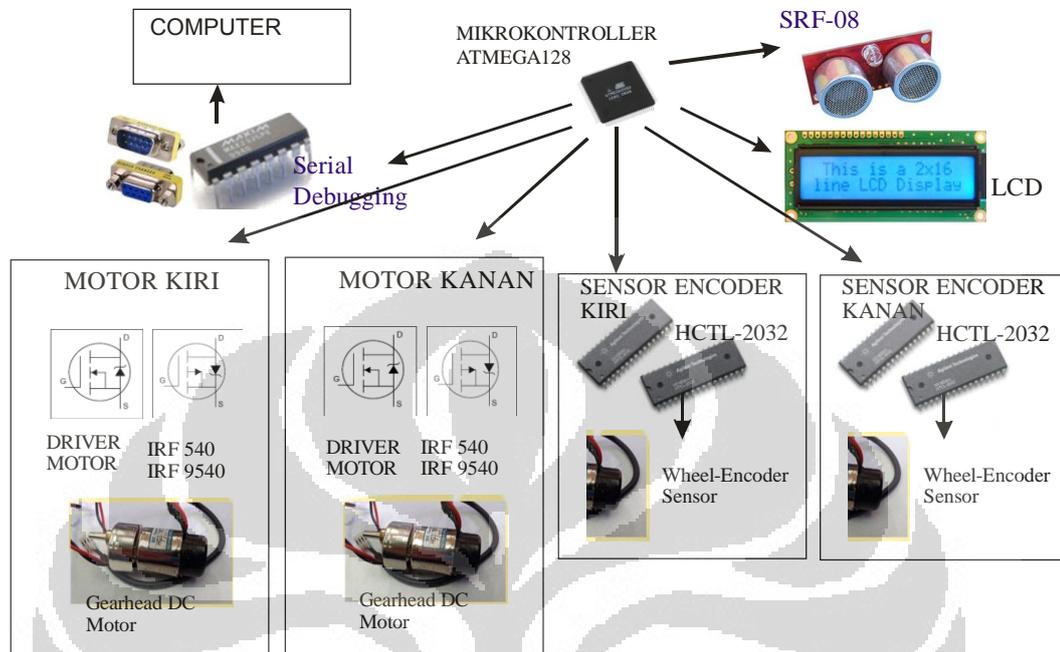
Gambar 3.8 Disain Sistem Putar Badan Dengan Huben Putar

3.2 Perancangan Sistem Kendali Robot

Pada dasarnya sistem kendali robot itu berkaitan erat dengan perangkat elektrik dari robot tersebut. Oleh karena itu, diperlukan perancangan sistem kendali robot yang meliputi perancangan sistem elektrik robot dan perancangan model kinematik robot. Tujuannya adalah agar didapatkan rancangan robot yang stabil.

3.2.1 Sistem Elektrik

Sistem elektrik pada sistem base robot adalah sebagai berikut :



Gambar 3.9 Diagram Alir Sistem Elektrik

3.2.1.1 Sistem Minimum

Sistem Minimum merupakan suatu sistem untuk mengaktifkan fungsi mikrontroller ATmega2560. Sistem minimum ini adalah tempat diletakkannya mikrokontroller. Rangkaiannya terdiri dari resistor, kapasitor, induktor, xtal atau Kristal atau clock eksternal, sistem debugging, sistem downloading, sistem komunikasi serial, sistem komunikasi dan sistem input dan output dari mikrokontroller. Berikut skematik sistem minimum ATmega2560.

Beberapa pin atau port yang digunakan pada sistem minimum ini, yaitu :

Tabel 3.2 Penggunaan Pin atau Port ATmega2560

Port/Pin	Keterangan Port/Pin	Penggunaan
Port E.4 – Port E7	Interupt 4,5,6,7	HCTL-2032
Port A	Port I/O	Data HCTL-2032
Port C	Port I/O	Inisialisasi HCTL-2032
Port F	Port I/O	LCD

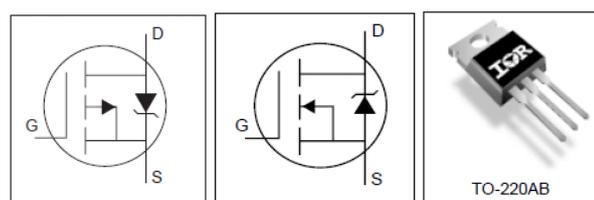
Port E.0, E.1	RX/TX 0	Komunikasi Serial
Port D.2, D.3	RX/TX 1	Komunikasi Serial
Port H.0, H.1	RX/TX 2	Komunikasi Serial
Port J.0, J.1	RX/TX 3	Komunikasi Serial
Port D.0, D.1	SCL/SDA	Komunikasi I2C Untuk SRF-08
Port D.4 – Port D.7	Port I/O	Mengendalikan Arah Putar Motor
Port B.5, B.6	Timer 1 (OCR1A/B)	Timer Gearhead DC Motor
Port B.7	Timer 0 (OCR0A)	Clock HCTL-2032



Gambar 3.10 Papan Sistem Minimum

3.2.1.2 Driver Motor

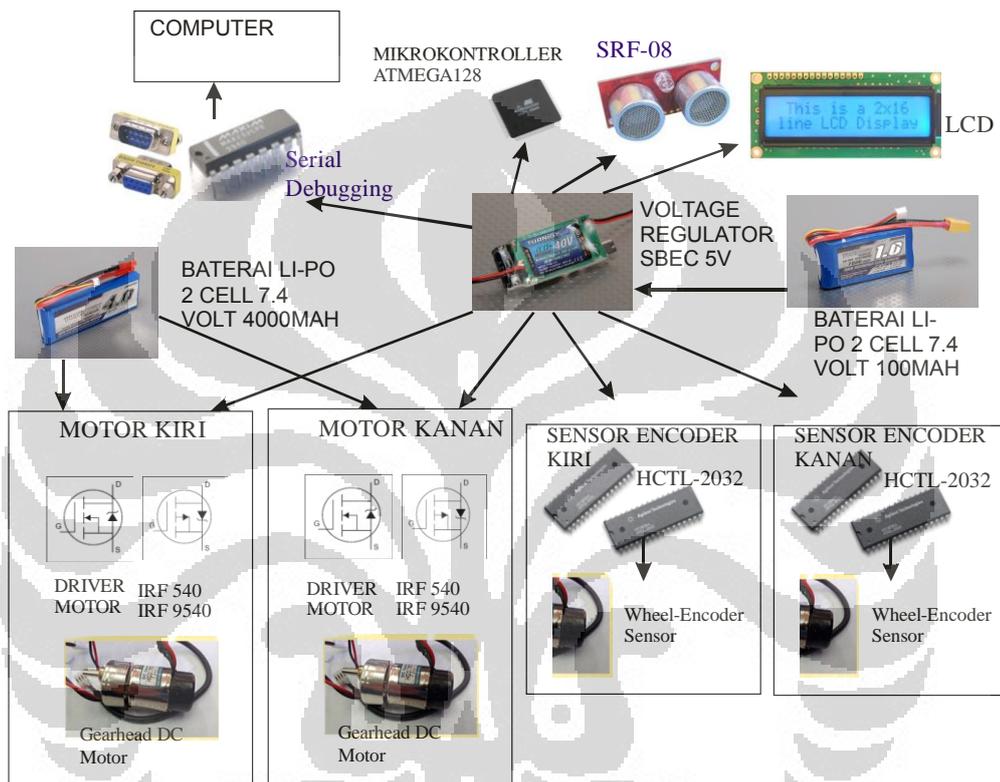
Driver motor yang digunakan adalah IC H-Bridge IRF-540 dan IRF-9540. Penggunaannya pada robot adalah dengan menghubungkannya ke pin OCR ATmega128 dan 1 pin CS. Satu motor dikendalikan dengan satu driver motor. Driver ini dipilih karena mampu menahan arus sebesar 3 Ampere. Berikut gambar IRF540 dan IRF9540 dan layout driver motor.



Gambar 3.11 IRF540 dan IRF9540

3.2.1.3 Sistem Daya

Sistem daya pada robot merupakan sistem pendistribusian daya atau suplai ke komponen elektrik robot. Pada sistem daya, baterai merupakan referensi acuan aliran karena semua komponen disuplai oleh baterai. Berikut diagram alir sistem daya robot.

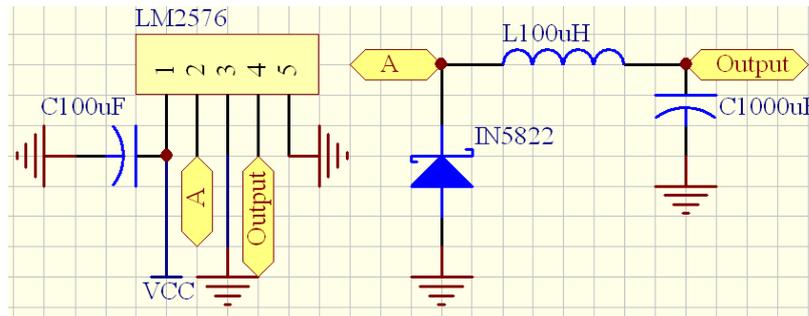


Gambar 3.12 Diagram Alir Sistem Daya Robot

Pada sistem daya terdapat voltage regulator yang merupakan rangkaian LM2576. LM2576 merupakan IC voltage regulator dengan input 7 sampai 40 VDC dan out yang dihasilkan adalah 5 VDC. LM2576 mampu mengalirkan arus hingga 3 Ampere. Berikut bentuk IC dan skematik LM2576.



Gambar 3.13 LM-2576



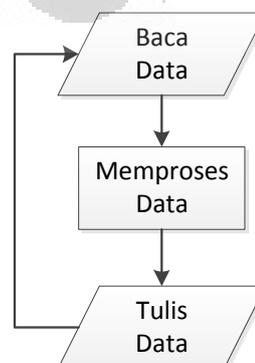
Gambar 3.14 Skematik Rangkaian Voltage Regulator 5 Volt

3.2.2 Sistem Kontrol Robotik

Pengontrolan merupakan bagian yang cukup penting dalam sistem robotika. Tanpa adanya pengontrolan, suatu sistem robotika akan menjadi benda yang mati. Sistem kontrol robotik pada dasarnya terdiri atas dua bagian, yaitu perangkat keras elektronik dalam *software* yang berisi program kemudi dan algoritma pengontrolan (perhatikan gambar 2.2).

Pada gambar 2.2 dapat terlihat bahwa sistem kontrol berfungsi untuk mengkolaborasikan sistem elektronik dan sistem mekanik demi mencapai fungsi tertentu. Kerja sama yang baik antara sistem elektronik, sistem mekanik dan sistem kontrol mampu menghasilkan sistem robotik yang baik sesuai dengan yang diharapkan.

Sistem kontrol itu sendiri bekerja berdasarkan nilai sensor yang didapat, kemudian data sensor tersebut diproses melalui algoritma kontrol dalam program kemudi yang akan menghasilkan nilai aktuasi untuk pergerakan aktuator dan begitu seterusnya secara berulang. Berikut *flow chart* sistem kontrol :

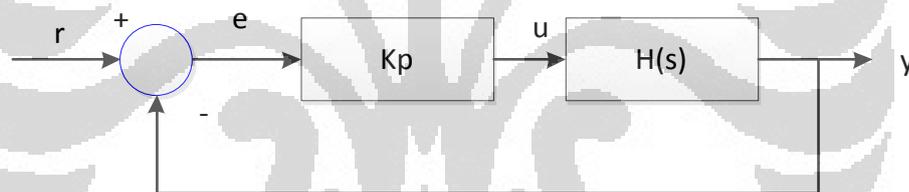


Gambar 3.15 Algoritma Sistem Kontrol

Bagian terpenting dalam sistem kontrol adalah bagian memproses data sensor. Pada bagian ini berbagai ide, konsep dan teori robotika dapat dikembangkan serta diterapkan, yaitu melalui kontrol PD untuk menggerakkan *gearhead DC motor*. Kontrol PD adalah kontrol yang terdiri dari pengendali P dan pengendali D. Berikut penjelasannya.

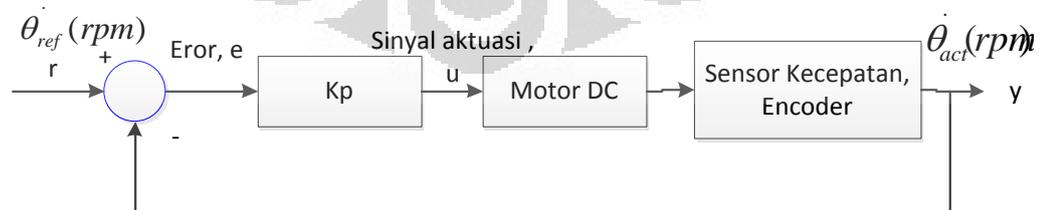
3.2.2.1 Kontrol P (Proporsional)

Kontroler adalah kontrol P jika $G(s) = k$, dengan k adalah sebuah konstanta. Jika $u = G(s) \cdot e$, maka $u = K_p \cdot e$, dengan K_p adalah konstanta proporsional. K_p berlaku sebagai gain atau penguat saja tanpa memberikan efek dinamik kepada kinerja kontroler. Penggunaannya memiliki berbagai keterbatasan karena sifat kontrol yang tidak dinamik ini, namun kontroler P ini mampu mencapai konvergensi meskipun dalam *error steady state* yang besar. Oleh karena itu, kontrol P ini biasanya digunakan untuk permulaan.



Gambar 3.16 Kontrol P

Penerapan kontrol P pada motor DC melalui sensor encoder sebagai sensor kecepatan motor DC digambarkan sebagai berikut :



Gambar 3.17 Kontrol P Pada Motor DC

Nilai output kontroler pada motor DC tergantung pada nilai eror antara kecepatan referensi dengan kecepatan aktual motor DC dan nilai konstanta K_p . Persamaanya dinyatakan sebagai berikut :

$$u(t) = K_p(\theta_{ref}(t) - \theta_{act}(t)) \quad (3.3)$$

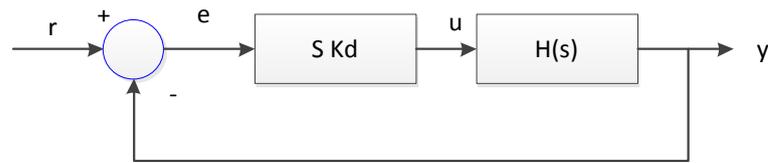
Jika eror bernilai positif, maka kecepatan aktual bernilai lebih kecil dari kecepatan referensi. Jika eror bernilai negatif, maka kecepatan aktual bernilai lebih besar dibandingkan dengan kecepatan referensi. Eror akan bernilai nol jika kecepatan referensi sama dengan kecepatan aktual. Akibatnya sinyal output akan bernilai nol dan menyebabkan motor tidak mendapat sinyal aktuasi lagi, sehingga motor akan berhenti berputar. Begitu putaran motor terdeteksi mulai berkurang, maka sensor akan mendeteksi bahwa kecepatan output tidak lagi sama dengan kecepatan referensi. Kecepatan aktual akan bernilai lebih kecil dari kecepatan referensi. Artinya eror sudah tidak lagi bernilai nol. Disinilah peranan kontroler untuk mengirimkan sinyal aktuasi $u(t)$ positif untuk menambahkan kecepatan motor. Demikian seterusnya secara berulang hingga eror pada kontrol P tidak dapat mempertahankan eror nol ini.

3.2.2.2 Kontrol D (Derivatif)

Sinyal kontrol u yang dihasilkan oleh kontrol D ini dapat dinyatakan sebagai $u = K_d \cdot \dot{e}$ atau $u = K_d \cdot de/dt$, sehingga $G(s)$ dapat dinyatakan :

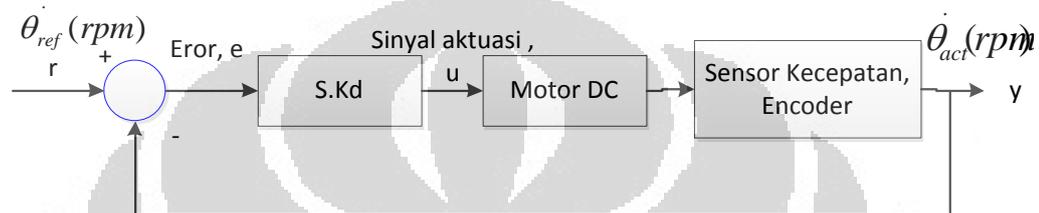
$$G(s) = s \cdot K_d \quad (3.4)$$

Pada persamaan tersebut dapat dinyatakan bahwa kontrol D ini mampu mengendalikan kecepatan atau *rate* dari eror, sehingga kontrol D ini dapat digunakan untuk memprediksi eror yang akan terjadi. Umpan balik yang diberikan sebanding dengan kecepatan perubahan eror, $e(t)$, sehingga eror yang akan terjadi dapat dikompensasi. Dalam blok kontrol D dinyatakan sebagai berikut :



Gambar 3.18 Kontrol D

Penerapan kontrol P pada motor DC melalui sensor encoder sebagai sensor kecepatan motor DC digambarkan sebagai berikut :



Gambar 3.19 Kontrol D Pada Motor DC

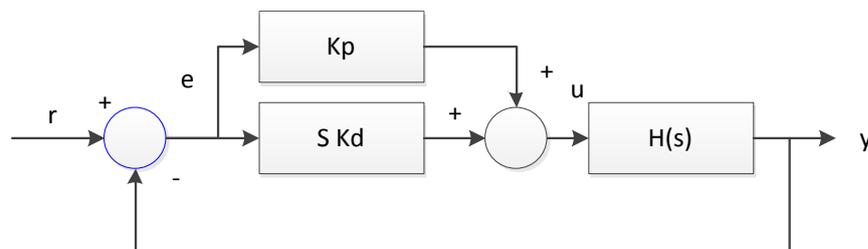
Kontrol D ini bekerja untuk mengurangi efek overshoot (respon berlebih) dalam menuju kondisi tunak, sehingga eror ini seolah-olah mampu memprediksi eror yang akan terjadi akibat efek eror kecepatan sebelumnya.

Berdasarkan diagram blok di atas, dapat dinyatakan persamaan keluaran pengendali D adalah :

$$u(t) = Kd \frac{d(\dot{\theta}_{ref}(t) - \dot{\theta}_{act}(t))}{dt}; s = \frac{de(t)}{dt} \quad (3.5)$$

3.2.2.3 Kontrol Kombinasi PD (Proporsional Derivatif)

Diagram kombinasi kontrol P dan D adalah sebagai berikut :

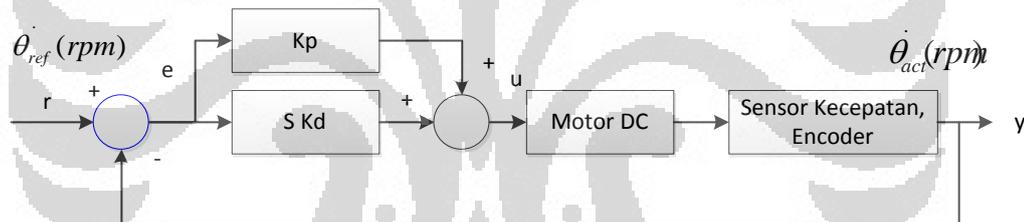


Gambar 3.20 Kontrol PD

Dengan mengkombinasikan kontrol P dan kontrol D, maka kelebihan dari masing-masing kontrol dapat disatukan untuk mendapatkan kontrol yang lebih baik. Namun, kontrol PD ini cenderung akan bekerja dengan baik untuk sistem yang linier dalam fungsi waktu. Artinya, persamaan dinamik dari sistem dianggap konstan, padahal pada kenyataannya persamaan dinamik akan selalu ada dan berubah-ubah.

Setiap sistem secara nyata selalu berhadapan dengan gangguan, contohnya motor DC yang memiliki permasalahan dengan friksi pada poros, *backslash* pada gear, perubahan karakteristik akibat temperatur dan lain sebagainya, sehingga kontrol PD ini digunakan untuk melakukan pendekatan melalui asumsi-asumsi model secara linier, yaitu mengabaikan faktor-faktor nonlinier yang dianggap sulit jika dimodelkan secara matematik.

Penerapan kontrol P pada motor DC melalui sensor encoder sebagai sensor kecepatan motor DC digambarkan sebagai berikut :



Gambar 3.21 Kontrol PD Pada Motor DC

Berdasarkan diagram blok di atas dapat dinyatakan persamaan keluaran pengendali PD adalah :

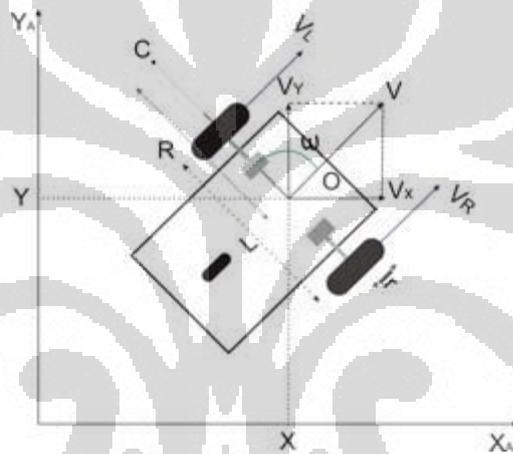
$$u(t) = Kp(\dot{\theta}_{ref}(t) - \dot{\theta}_{act}(t)) + Kd \frac{d(\dot{\theta}_{ref}(t) - \dot{\theta}_{act}(t))}{dt}; s = \frac{de(t)}{dt} \quad (3.6)$$

Pengendali PD ini akan memiliki kelebihan atas pengendali P dan pengendali D yang dikombinasikan. Pengendali PD ini akan meningkatkan waktu respon dalam menuju kondisi tunak tanpa terjadi *overshoot* jika konstanta P dan konstanta D pengendali yang digunakan tepat.

BAB 4

MOBILE ROBOT

Mobile robot adalah sistem robot yang mampu memindahkan dirinya sendiri dari posisi satu ke posisi yang lain, dimana keseluruhan badan robot berpindah tempat. Penerapan umum sistem *mobile robot* ini adalah sistem navigasi pada robot. Sistem navigasi robot merupakan suatu sistem atau cara untuk memindahkan robot dari suatu posisi ke posisi tertentu melalui pembacaan sensor-sensor sebagai umpan balik. Penentuan posisinya melalui sebuah koordinat Cartesian, yaitu posisi terhadap sumbu x , y dan sudut tertentu.



Gambar 4.1 Koordinat Lokal dan Global Mobile Robot

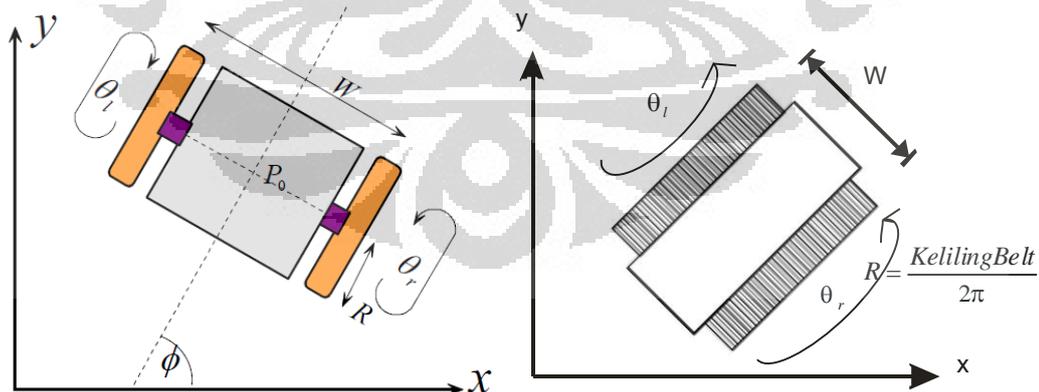
Tujuan dari penerapan sistem *mobile robot* ini adalah untuk mengetahui posisi robot dari awal bergerak hingga posisinya saat itu sehingga robot mampu membuat rencana untuk menuju ke posisi berikutnya dengan tepat, sedangkan jika sebuah robot tidak mengetahui posisinya saat memulai sebuah trajektori, maka robot tersebut akan memiliki masalah dalam menentukan posisi tujuan. Dengan demikian sistem *mobile robot* dirasakan amat penting diterapkan dalam sistem navigasi robot.

Aplikasi sederhana dari penerapan *mobile robot* adalah pada robot beroda, yaitu pada sebuah *Differential Mobile Robot* (DMR). DMR adalah sebuah disain robot beroda yang memiliki dua buah roda untuk berpindah tempat. DMR adalah disain yang paling sering digunakan dalam aplikasi *mobile robot*.

Agar robot penerima tamu yang dibuat mampu melakukan navigasi, maka diperlukan suatu model kinematika dari robot penerima tamu yang dibuat dan algoritma *mobile robot*-nya melalui suatu *trajectory planning* tertentu. Berikut akan dijelaskan model kinematika robot penerima tamu dan algoritma *mobile robot* yang akan digunakan pada robot penerima tamu.

4.1 Model Kinematika Robot

Base robot penerima tamu ini pada dasarnya merupakan sebuah robot beroda yang berbentuk seperti tank. Oleh karena itu, *mobile robot* yang dibuat adalah *mobile robot* berjenis TMR (*tracked mobile robot*), dimana terdapat *track* yang meliliti roda depan dan belakang sehingga memiliki bentuk roda seperti tank. Pada disain yang dibuat, *track* tersebut merupakan sebuah belt yang berfungsi sebagai alas roda dan sekaligus berfungsi pada sistem perodaan. TMR itu sendiri adalah sebuah *differential mobile robot* (DMR) yang menggunakan prinsip *skid steering*, yaitu variasi dari kecepatan relatif *track* yang menghasilkan *slippage* dan *soil-shearing* yang pada akhirnya menghasilkan *steering*. Oleh karena itu, terdapat kesamaan antara TMR dan DMR dari segi lokomotif. Dengan demikian model kinematika robot penerima tamu didapatkan dengan melakukan pendekatan kepada *differential mobile robot* (DMR).



Gambar 4.2 Model DMR dan TMR

Perbedaannya terletak pada bentuk rodanya, dimana DMR bulat sedangkan TMR berbentuk tank atau elip. Pendekatan yang dilakukan adalah

dengan menjadikan keliling roda tank menjadi keliling roda bulat untuk didapatkan nilai jari-jari roda robot melalui persamaan berikut :

$$R = \frac{\text{KelilingBdt}}{2\pi} \quad (4.1)$$

Pada pemodelan kinematikanya, robot diasumsikan berada pada koordinat Cartesian x dan y . Parameter-parameter yang terdapat dalam memodelkan kinematika robot adalah :

- a. ϕ yang merupakan sudut arah hadap robot
- b. W yang merupakan lebar robot yang diukur dari garis tengah roda ke roda
- c. R yang merupakan jari-jari roda
- d. (x,y) yang merupakan koordinat global Cartesian posisi awal robot.
- e. $P_o(x,y)$ yang merupakan titik kontrol robot
- f. θ_l yang merupakan sudut putar roda kiri
- g. θ_r yang merupakan sudut putar roda kanan

Walaupun memiliki dua buah aktuator, namun DMR memiliki tiga buah parameter yang dapat dikontrol, yaitu arah x , arah y dan arah hadap robot ϕ . Ketiga parameter tersebut dikontrol untuk mendapatkan gerakan nonholonomic.

Gerak holonomic merupakan gerak robot untuk berpindah dari satu posisi ke posisi yang lainnya dengan banyak arah gerakan, sedangkan gerak nonholonomic merupakan gerak robot untuk berpindah melalui satu arah gerakan. Dengan kata lain sistem gerak holonomic memiliki banyak arah gerakan sedangkan sistem gerak nonholonomic memiliki keterbatasan arah gerakan.

Dalam memodelkan kinematikanya, robot diasumsikan bergerak relatif pelan dan roda tidak slip terhadap permukaan jalan. Dengan demikian didapatkan komponen x dan y dalam suatu persamaan nonholonomic akibat robot tidak dapat bergerak ke samping sebagai berikut :

$$\dot{y}_o \cos \phi - \dot{x}_o \sin \phi = 0 \quad (4.2)$$

Saat kedua roda berputar, didapatkan persamaannya sebagai berikut :

$$\dot{x}_o \cos \phi + \dot{y}_o \sin \phi + \frac{W}{2} \dot{\phi} = R \dot{\theta}_r \quad (4.3)$$

$$\dot{x}_o \cos \phi + \dot{y}_o \sin \phi - \frac{W}{2} \dot{\phi} = R \dot{\theta}_l \quad (4.4)$$

Berdasarkan persamaan di atas didapatkan model forward Jacobian dari DMR, yaitu :

$$\dot{x} = J(\phi) \dot{\theta} \quad (4.5)$$

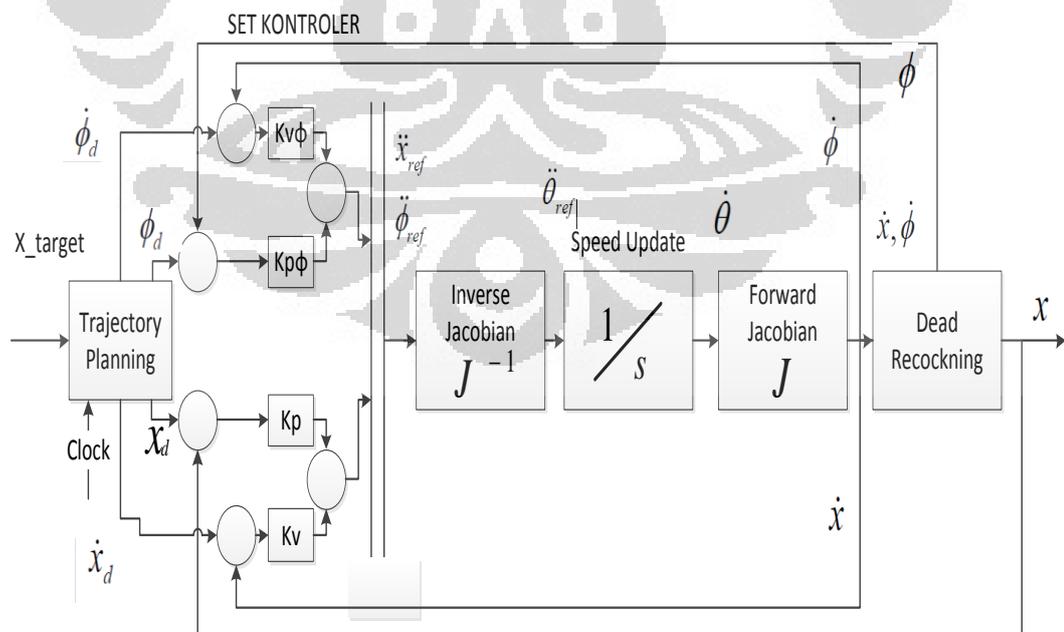
$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \frac{R}{2} \cos \phi & \frac{R}{2} \cos \phi \\ \frac{R}{2} \sin \phi & \frac{R}{2} \sin \phi \\ \frac{R}{W} & -\frac{R}{W} \end{bmatrix} \begin{bmatrix} \dot{\theta}_r \\ \dot{\theta}_l \end{bmatrix} \quad (4.6)$$

Didapatkan persamaan Inverse Jacobiannya sebagai berikut :

$$\begin{bmatrix} \dot{\theta}_r \\ \dot{\theta}_l \end{bmatrix} = \frac{1}{R} \begin{bmatrix} \cos \phi & \sin \phi & \frac{w}{2} \\ \cos \phi & \sin \phi & -\frac{w}{2} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} \quad (4.7)$$

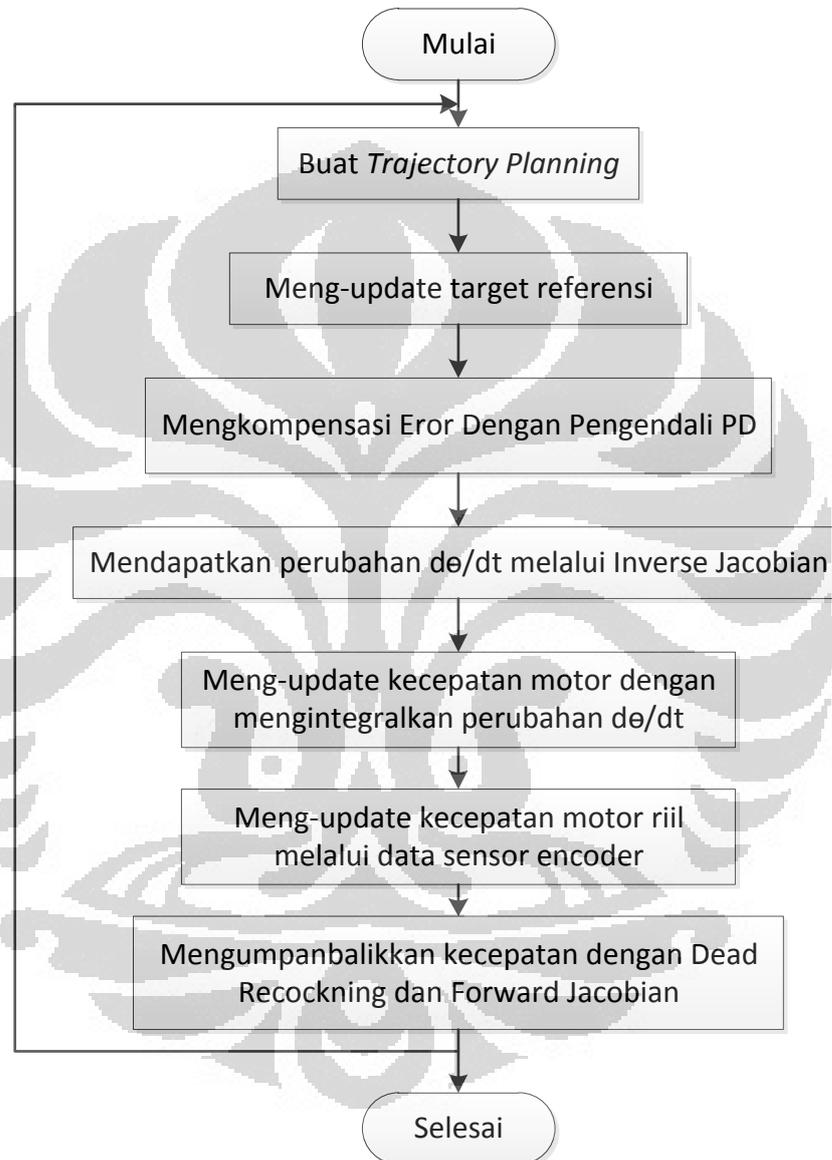
4.2 Algoritma Mobile Robot

Algoritma *mobile robot* dari robot penerima tamu diilustrasikan sebagai berikut :



Gambar 4.3 Algoritma Mobile Robot

Algoritma *mobile robot* yang digunakan terdiri dari enam blok, yaitu blok *trajectory planning*, pengesetan kontroler, Inverse Jacobian, *speed update*, Forward Jacobian dan Dead Recockning. Berikut diagram alir *mobile robot* pada robot penerima tamu :



Gambar 4.4 Diagram Alir Robot Service Mobile Robot

Berikut adalah penjelasan masing-masing blok pada algoritma *mobile robot* :

a. Blok *Trajectory Planning*

Blok ini berisi rencana trajektori atau pergerakan robot penerima tamu dalam bernavigasi atau berpindah posisi. Trajektori digambarkan melalui persamaan-persamaan yang diinisialisasi sebagai nilai x target, y target dan psi target (x_{target} , y_{target} dan ϕ_{target} dalam program).

b. Blok Set Kontroler

Blok ini adalah blok pengendali PD, dimana besarnya eror yang terjadi antara nilai referensi dengan nilai aktual dikompensasi melalui pengendali PD. Pengendali PD ini juga berfungsi untuk mengkompensasi laju eror. Sebelum masuk ke blok ini, nilai x target, y target dan psi target terlebih dahulu disimpan dalam variabel referensi agar tidak terganggu dengan nilai target berikutnya. Blok diagram ini berisi persamaan PD kontroler seperti yang dijelaskan pada Bab 3. Persamaannya adalah :

$$\ddot{x}_x = (x_{xref} - x_x)Kp_x + (\dot{x}_{xref} - \dot{x}_x)Kd_x \quad (4.8)$$

$$\ddot{x}_y = (x_{yref} - x_y)Kp_y + (\dot{x}_{yref} - \dot{x}_y)Kd_y \quad (4.9)$$

$$\ddot{\phi} = (\phi_{ref} - \phi)Kp_\phi + (\dot{\phi}_{ref} - \dot{\phi})Kd_\phi \quad (4.10)$$

, dimana KP adalah konstanta proporsional dan KV adalah konstanta derivatif.

Blok ini menerima inputan berupa perubahan posisi yang diumpanbalikkan oleh blok Forward Jacobian untuk mendapatkan eror antara nilai referensi dengan nilai aktual.

c. Blok Inverse Jacobian

Blok ini berisi persamaan Jacobian yang diubah dari persamaan *state space* menjadi persamaan umum. Fungsi blok ini adalah untuk mendapatkan besarnya $\dot{\theta}$ referensi, sehingga dapat diketahui besarnya perubahan PWM.

$$\ddot{\theta}_r = \frac{(\ddot{x}_x \cos\theta + \ddot{x}_y \sin\theta + \frac{W}{2} \ddot{\phi})}{R} \quad (4.11)$$

$$\ddot{\theta}_r = \frac{(\ddot{x}_x \cos\theta + \ddot{x}_y \sin\theta - \frac{W}{2} \ddot{\phi})}{R} \quad (4.12)$$

d. Blok Integrator

Blok ini adalah sebuah integrator yang berfungsi untuk memperbaharui kecepatan aktual dari motor. Keluaran dari integrator merupakan umpan balik dari kecepatan putar motor DC.

e. Blok Forward Jacobian

Blok ini digunakan untuk mendapatkan perubahan posisi dari kecepatan real motor DC melalui data dari sensor encoder. Perubahan posisi yang didapatkan akan diumpanbalikkan ke blok set kontrol untuk dilakukan *update*-an posisi.

$$\dot{x}_x = \frac{R \cos \theta (\dot{\theta}_{r,real} + \dot{\theta}_{l,real})}{2} \quad (4.13)$$

$$\dot{x}_y = \frac{R \sin \theta (\dot{\theta}_{r,real} + \dot{\theta}_{l,real})}{2} \quad (4.14)$$

$$\dot{\phi} = \frac{R(\dot{\theta}_{r,real} + \dot{\theta}_{l,real})}{W} \quad (4.15)$$

f. Blok Dead Recockning

Blok ini berfungsi untuk meng-*update* posisi robot, data posisi yang didapatkan akan diumpanbalikkan ke set kontrol. Blok ini cukup penting untuk navigasi robot karena dengan adanya *dead recockning* ini robot akan mengetahui posisinya dan mampu melakukan pemberhentian pada posisi yang dituju.

Dead recockning ini terdiri atas persamaan-persamaan sebagai berikut :

$$v = \dot{x} \cos \phi + \dot{y} \sin \phi \quad (4.16)$$

$$\phi_k = \frac{R}{W} (\theta_r - \theta_l) + \phi_{init} \quad (4.17)$$

$$x_k = x_{k-1} + v \cos \left(\frac{\phi_k + \phi_{k-1}}{2} \right) \Delta t \quad (4.18)$$

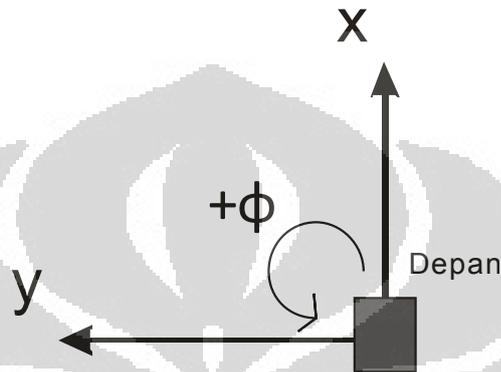
$$y_k = y_{k-1} + v \sin \left(\frac{\phi_k + \phi_{k-1}}{2} \right) \Delta t \quad (4.19)$$

4.3 Trajectory Planning

Penerapan trajektori sebagai sistem navigasi pada robot penerima tamu adalah dengan mengkombinasikan trajektori lurus, putar dan elip sebagai gerakan

dasar pada sistem navigasi robot penerima tamu. Robot diharapkan mampu bernavigasi ke segala arah dan posisi. Oleh karena itu, gerakan-gerakan dasar tersebut diharapkan sudah teruji untuk mendapatkan sistem navigasi yang sesuai dengan yang diharapkan.

Trajektori-trajektori ini dirancang pada koordinat Cartesian x dan y menurut skema sebagai berikut :



Gambar 4.5 Arah Koordinat Positif Trajektori Robot

4.3.1 Trajektori Lurus

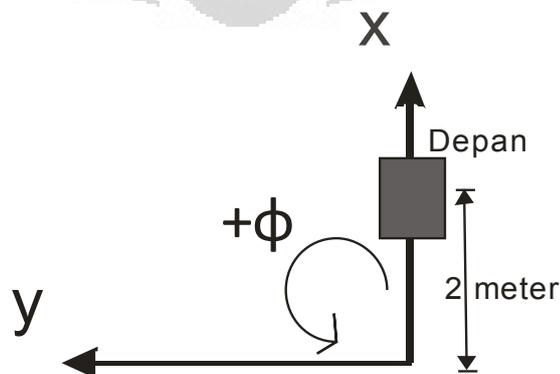
Trajektori ini dibuat dengan membuat target pada posisi sumbu x dan menginisialisasi sebagai nilai nol untuk target pada posisi sumbu y dan derajat ψ . Berikut contoh penerapan trajektori lurus pada robot.

$X_{target} = \text{koordinat } x \text{ target};$

$Y_{target} = 0;$

$\phi_{target} = 0;$

Kode di atas menyatakan bahwa robot akan bernavigasi lurus pada sumbu x sejauh 2 meter.



Gambar 4.6 Trajektori Lurus Sejauh 2 Meter

4.3.2 Trajektori Putar

Trajektori putar bekerja dengan memutar arah hadap robot menurut sumbu ϕ . Satuan arah hadap robot diinisialisasi dalam satuan radian. Konversinya dalam bentuk derajat adalah :

$$\text{NilaiDerajat} = \frac{\text{NilaiRadian}}{\pi} \times 180 \quad (4.12)$$

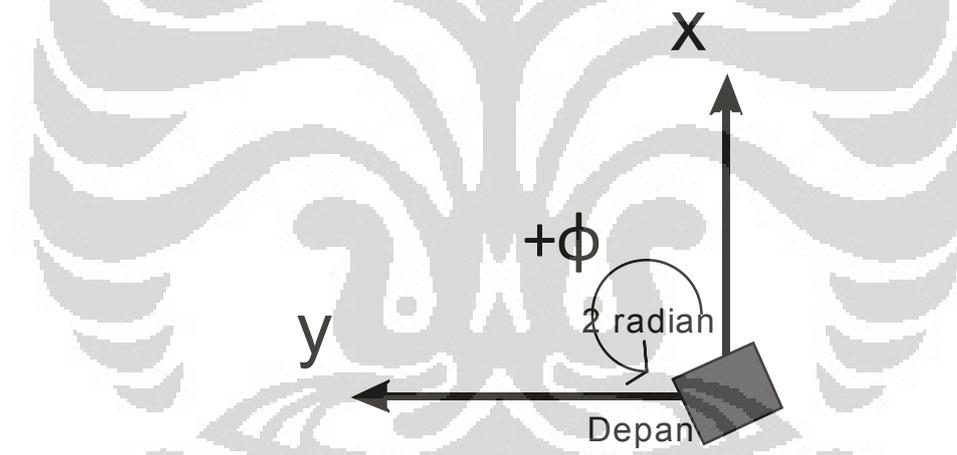
Cara yang dapat dilakukan untuk membuat trajektori putar adalah dengan menginisialisasi besar sudut putar robot dalam bentuk radian sebagai berikut :

$X_{\text{target}} = 0;$

$Y_{\text{target}} = 0;$

$\phi_{\text{target}} = \text{sudut dalam radian};$

Kode di atas menandakan bahwa robot akan berputar sebesar 2 radian menurut sumbu psi positif. Berikut trajektori yang dihasilkan :



Gambar 4.7 Trajektori Putar Sebesar 2 Radian

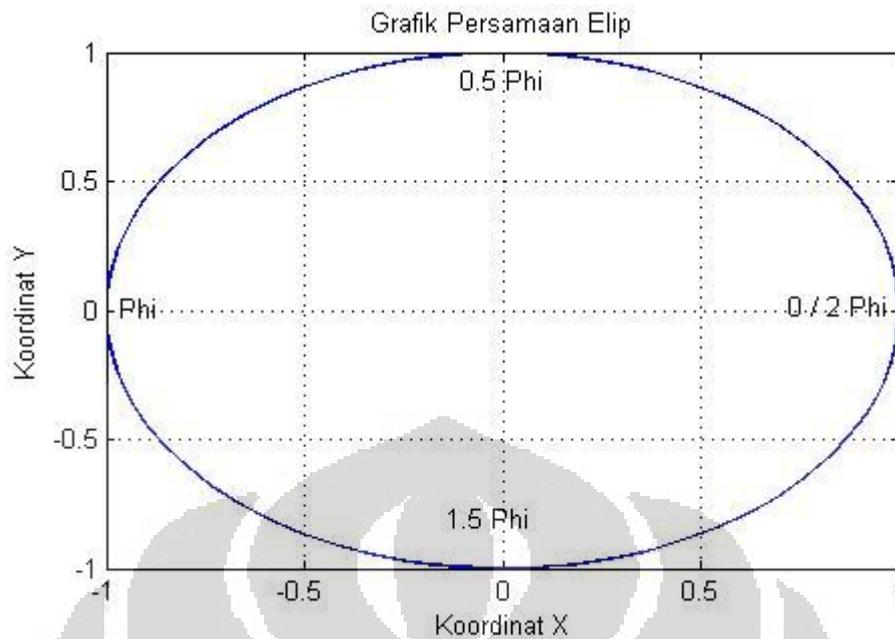
4.3.3 Trajektori Elip

Trajektori elip ini merupakan trajektori kombinasi dari trajektori lurus dan trajektori putar dengan memanfaatkan persamaan fungsi elip sebagai trajektorinya. Trajektori elip dapat dilakukan jika trajektori dasar telah benar dilakukan.

Fungsi elip pada suatu koordinat direpresentasikan dengan fungsi sinus dan cosinus sebagai berikut :

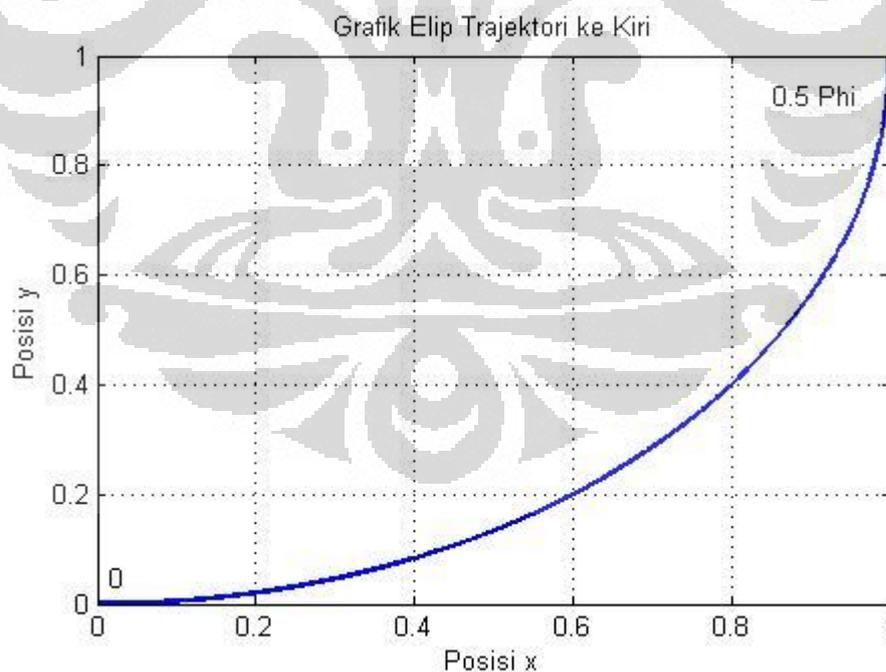
$$x = a \cos t \text{ dan } y = a \sin t; 0 \leq t \leq 2\pi \quad (4.13)$$

Untuk nilai $x = \cos t$ dan $y = \sin t$ dengan $0 \leq t \leq 2\pi$, maka didapatkan grafik :



Gambar 4.8 Grafik Elip Untuk $0 \leq t \leq \pi$

Berdasarkan grafik tersebut dibuatlah trajektori kurva dengan $0 \leq t \leq \frac{\pi}{2}$ dan dengan memodifikasi posisi awal kurva menjadi posisi 0 untuk trajektori elip ke kiri dan ke kanan :



Gambar 4.9 Trajektori ke Kiri

Pada gambar 4.9, posisi awal grafik dinaikkan sepanjang koordinat y yang diinginkan agar posisi awal robot didapatkan koordinat (0,0) dengan

trajektori ke arah kiri. Berikut persamaan fungsinya dengan x_i dan y_i adalah koordinat pergerakan robot.

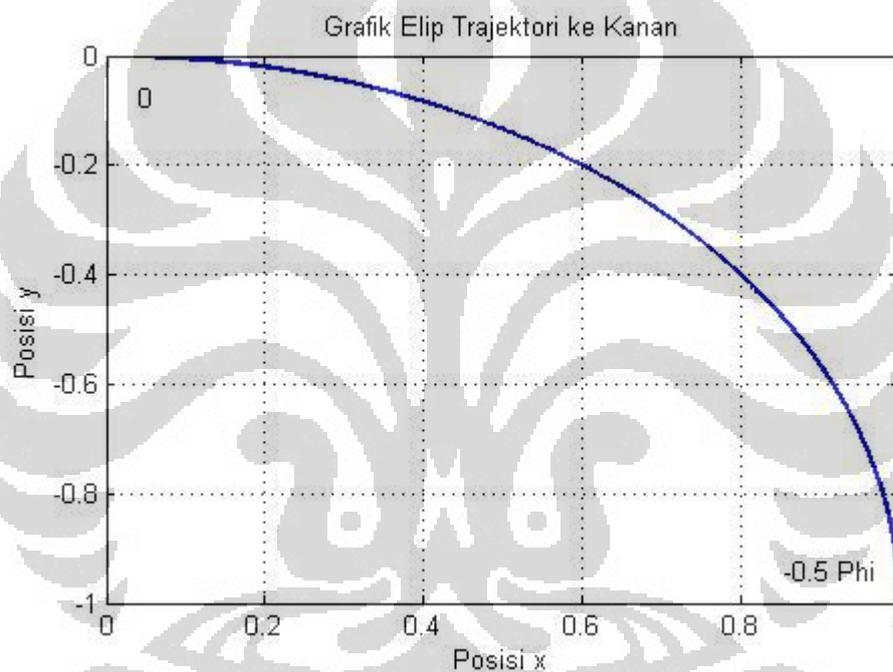
$$x = x_i \sin(t) \quad (4.14)$$

$$y = y_i \cos(t) + y_i \quad (4.15)$$

Selanjutnya, untuk trajektori ke kanan dilakukan dengan mengurangi koordinat y yang diinginkan sehingga didapatkan posisi awal robot dengan koordinat $(0,0)$ dan trajektori ke kanan.

$$x = x_i \sin(t) \quad (4.14)$$

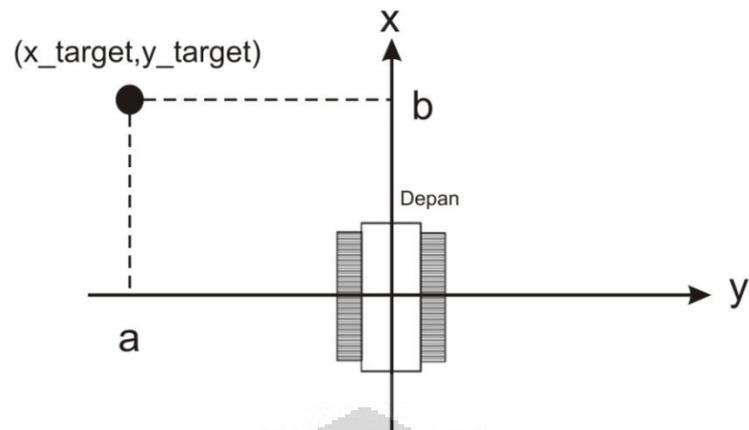
$$y = y_i \cos(t) - y_i \quad (4.15)$$



Gambar 4.10 Trajektori ke Kanan

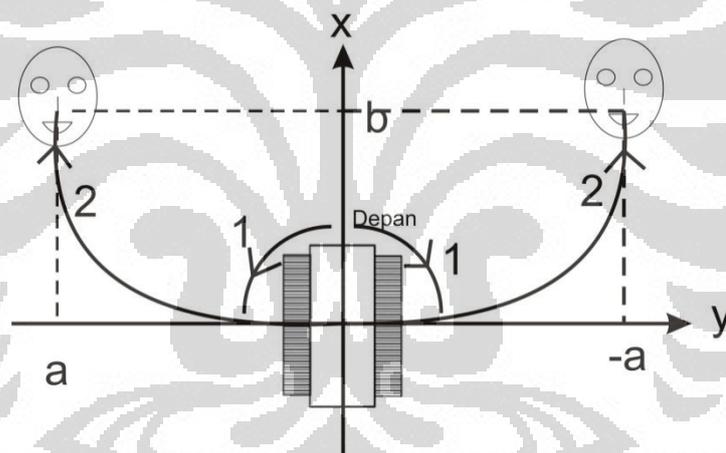
4.3.4 Trajektori Navigasi Robot Penerima Tamu

Robot Penerima tamu ini akan bergerak mendekati objek yang dituju berdasarkan nilai koordinat X dan Y tertentu.

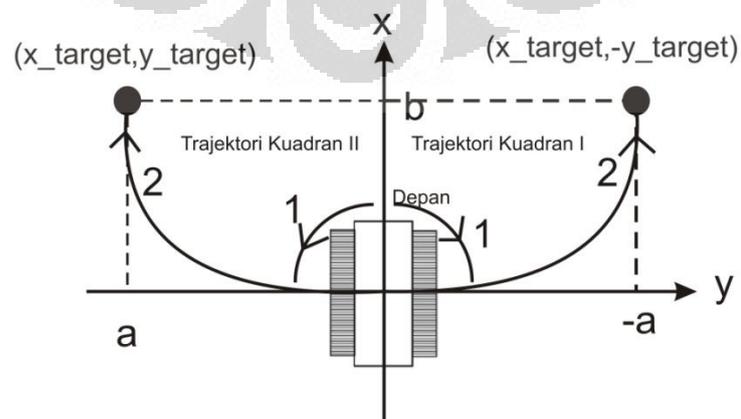


Gambar 4.11 Navigasi Robot Penerima Tamu

Selanjutnya pergerakan robot mendekati objek tersebut di representasikan dengan sebuah trajektori, yaitu melalui persamaan kombinasi antara trajektori putar dan trajektori elip. Perhatikan gambar berikut :



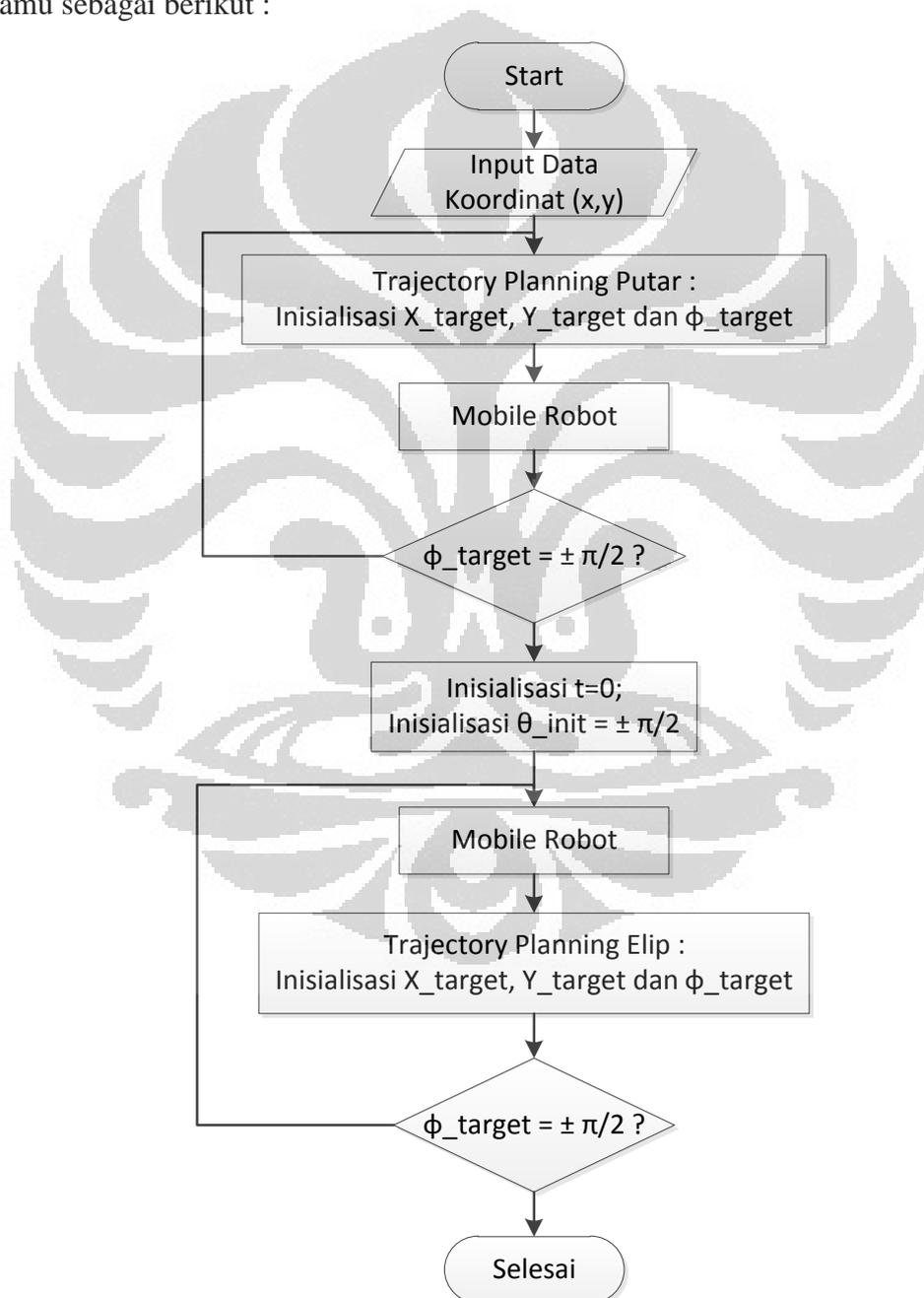
Gambar 4.12 Trajektori dan Orientasi Objek



Gambar 4.13 Trajektori Navigasi Robot Penerima Tamu

Trajektori kombinasi tersebut digunakan untuk orientasi objek menghadap ke arah robot seperti pada gambar 4.12. Algoritmanya adalah dengan melakukan trajektori putar sebesar $\pm 0.5 \pi$, kemudian melakukan trajektori elip.

Pada saat memulai trajektori baru besarnya nilai t , diinisialisasi menjadi nol karena setiap trajektori gerakan dasar dimulai dari $t = 0$. Inputan data yang dimasukkan ke robot adalah berupa data koordinat robot. Untuk orientasi secara umum terlihat pada gambar 4.12. Diagram alir sistem navigasi robot penerima tamu sebagai berikut :

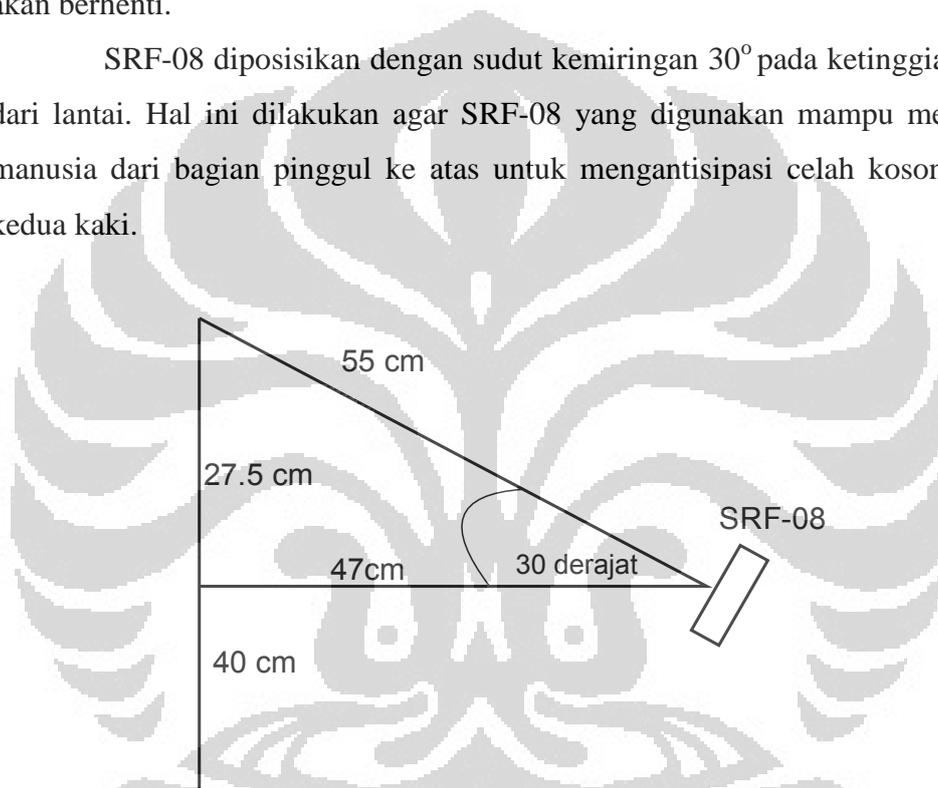


Gambar 4.14 Diagram Alir Trajektori Navigasi Robot Penerima Tamu

4.3.5 Trajektori Navigasi Robot Penerima Tamu Dengan SRF-08

Sistem navigasi robot penerima tamu dengan *trajectory planning* tidak selama tepat, dalam artian masih terdapat eror. Selain itu, untuk mengantisipasi agar robot tidak menabrak objek baik yang diam maupun yang bergerak, maka diperlukan suatu data sensor jarak sebagai acuan jarak robot terhadap objek. Sensor SRF-08 ini akan digunakan sebagai umpan balik bahwa robot sudah dekat dengan objek. Jika robot sudah mencapai jarak tertentu pada robot, maka robot akan berhenti.

SRF-08 diposisikan dengan sudut kemiringan 30° pada ketinggian 40 cm dari lantai. Hal ini dilakukan agar SRF-08 yang digunakan mampu mendeteksi manusia dari bagian pinggul ke atas untuk mengantisipasi celah kosong antara kedua kaki.



Gambar 4.15 Derajat Kemiringan SRF-08

Robot akan berhenti saat jarak manusia dengan robot sekitar 47 cm. Jarak tersebut diperoleh berdasarkan kesesuaian dengan panjang tangan robot ketika berjabat tangan dengan seseorang.

BAB 5

PENGUJIAN

Setelah sistem mekanik dan sistem elektrik selesai dibuat, langkah berikutnya adalah memasukkan algoritma *mobile robot* ke dalam mikrokontroler. Memasukkan algoritma *mobile robot* dengan menggunakan bahasa pemrograman C menggunakan perangkat lunak Code Vision AVR versi 2.03.4. Sebelum dilakukan pengujian trajektori, perlu dilakukan pengujian terhadap kontroler PD yang digunakan, yaitu dengan mencari konstanta proporsional dan konstanta derivatif yang tepat. Setelah didapatkan nilai Kp dan Kd yang sesuai barulah dilakukan pengujian terhadap trajektori. Pengujian dilakukan pada trajektori dasar robot, yaitu trajektori lurus, putar dan melingkar. Tujuan dilakukan pengujian pada trajektori dasar ini adalah untuk melihat kemampuan navigasi robot. Selanjutnya dilakukan pada trajektori kombinasi, yaitu gabungan trajektori dasar yang merupakan navigasi untuk robot penerima tamu dengan umpan balik SRF-08.

5.1 Kondisi Robot dan Lingkungan Pengujian

Robot yang akan diuji ini memiliki spesifikasi sebagai berikut :

- a. Dimensi panjang x lebar x tinggi adalah 350 mm x 380 mm x 1000 mm
- b. Bobot robot adalah 10.6 Kg
- c. Memiliki 2 sensor encoder dan 4 sensor SRF-08
- d. Robot terdiri atas sistem base dan badan robot

Pengujian dilakukan di gedung Engineering Center Fakultas Teknik Universitas Indonesia dengan lantai terbuat dari keramik dan pencahayaan dinilai cukup.

5.2 Perangkat Pendukung Pengujian

Pengujian yang dilakukan berhubungan dengan sistem navigasi robot penerima tamu dengan menggunakan pengendali PD ini menggunakan perangkat pendukung seperti :

- a. Kamera untuk mendokumentasikan foto atau video

- b. USB Downloader untuk memasukkan program ke mikrokontroler
- c. USB Serial untuk komunikasi pada proses *debugging* data pengujian
- d. Laptop untuk menyimpan hasil *debugging*
- e. Spidol dan lem isolasi untuk validasi trajektori

5.3 Pengujian Pengendali PD

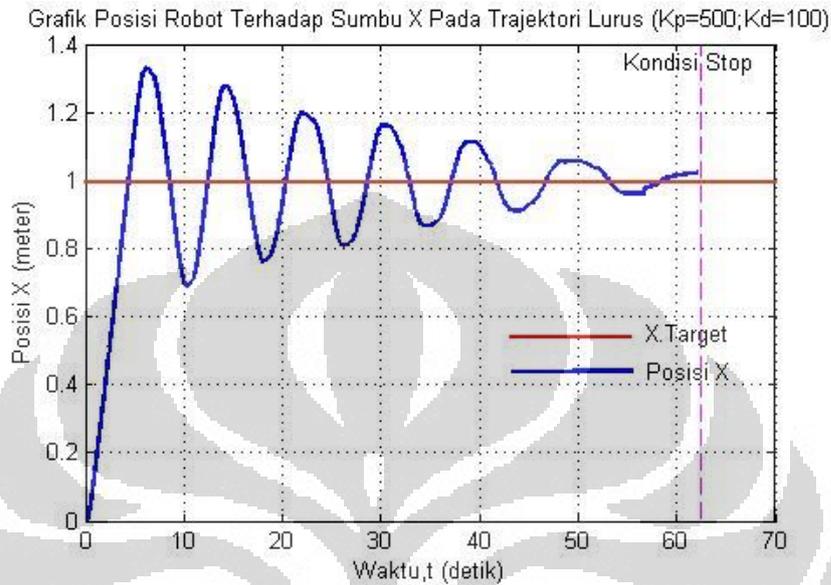
Pengujian ini dilakukan untuk mendapatkan validasi nilai konstanta proporsional dan konstanta derivatif yang sesuai. Pengujiannya dilakukan dengan memvariasikan konstanta P dan konstanta D pada blok kontroler. Tujuan validasi nilai Kp dan Kd adalah agar navigasi pada trajektori yang telah dibuat memiliki respon yang baik dan tidak beresilasi sehingga robot mampu berpindah posisi secara baik.

Pengujian validasi nilai Kp dan Kd melalui metode *trial error*, yaitu dengan memvariasikan nilai Kp dan Kd pada nilai-nilai tertentu dan melihat pergerakan pada robot. Kp dan Kd terpilih adalah yang memiliki waktu respon baik dan tidak beresilasi. Berikut pemvariasian nilai Kp dan Kd yang telah dilakukan :

Tabel 5.1 Validasi Kp dan Kd

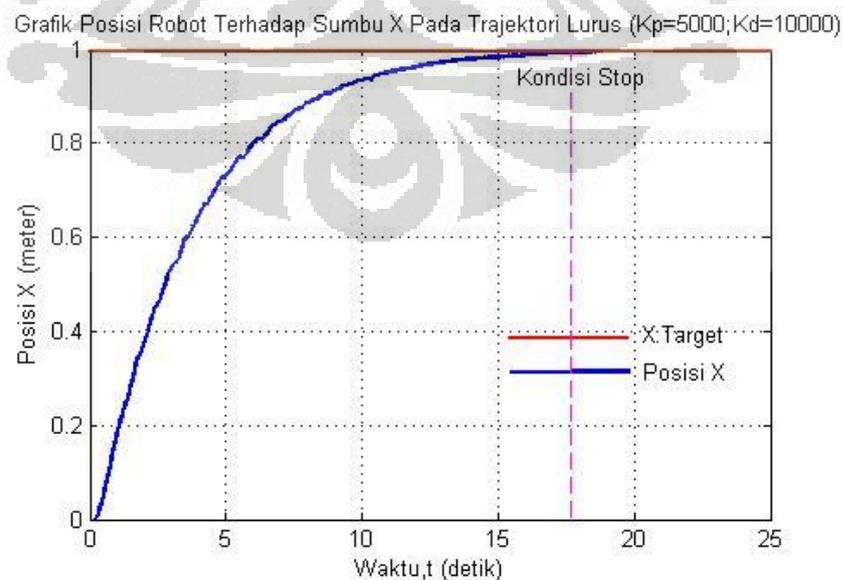
No	Nilai Kp	Nilai Kd	Keluaran
1	500	100	Respon time cukup baik terlihat dari akselerasi robot, namun osilasi yang terjadi terlalu banyak (terlalu lama konvergensinya)
2	5000	10000	Transien robot tidak stabil
3	2350	10000	Transien robot tidak stabil
4	2350	100	Respon time cukup baik, namun terlalu banyak osilasi bahkan dapat dikatakan mendekati divergensi
5	10000	1300	Respon time baik namun masih ada osilasi beberapa cm dari posisi sebenarnya
6	100	1300	Respon time sangat lambat
7	2350	1300	Bagus, respon cukup cepat tanpa osilasi dan robot langsung menuju posisi yang dituju

Berdasarkan percobaan-percobaan di atas dapat digambarkan grafik hasil percobaan, yaitu Nomor Data terhadap Posisi X aktual. Grafik yang diplot hanya untuk pengujian dengan K_p 500 dan K_d 100, K_p 5000 dan K_d 10000 dan K_p 2350 dan K_d 1300. Berikut grafiknya :



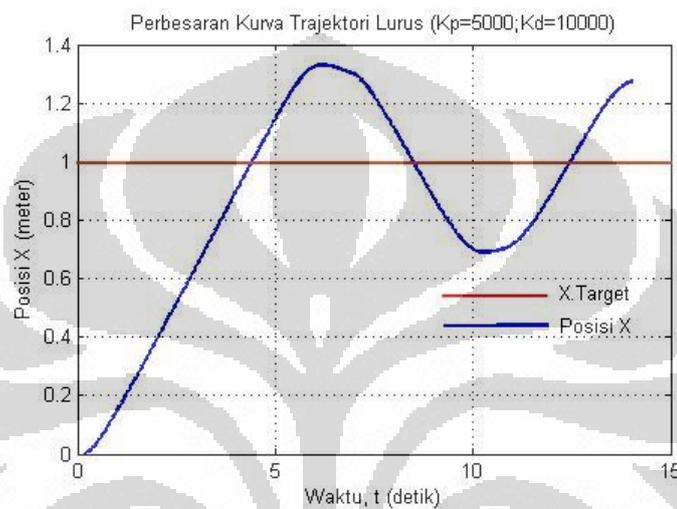
Gambar 5.1 Grafik Posisi Robot Pada Sumbu X ($K_p=500$ dan $K_d=100$)

Pada gambar 5.1 di atas terlihat bahwa robot mampu mencapai set point sebesar 1 meter, namun terjadi osilasi di sekitar nilai 1 meter tersebut yang semakin lama osilasi tersebut akan mencapai 1 meter. Hal ini menandakan bahwa nilai K_p dan K_d tersebut masih belum tepat digunakan oleh robot.

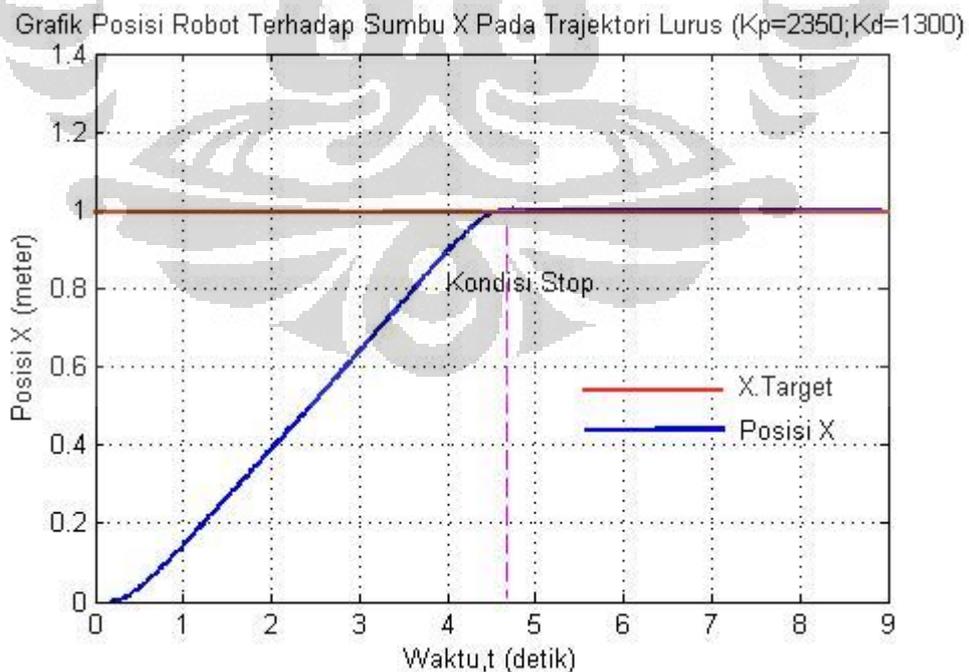


Gambar 5.2 Grafik Posisi Robot Pada Sumbu X ($K_p= 5000$ dan $K_d=10000$)

Berdasarkan grafik pada gambar 5.2. di atas terlihat bahwa grafik posisi mengalami perlambatan saat mendekati set point sejauh 1 meter. Selain itu pergerakan robot terlihat tidak stabil, dalam artian akselerasi robot tidak stabil, terkadang cepat terkadang lambat. Hal ini terlihat dari grafik pada gambar 5.2 yang diperbesar pada gambar 5.3. Pada gambar tersebut grafik terlihat tidak stabil, yaitu berosilasi. Oleh karena itu nilai K_p dan K_d tersebut masih belum tepat.



Gambar 5.3 Grafik Posisi Robot Pada Sumbu X ($K_p=5000$ dan $K_d=10000$)



Gambar 5.4 Grafik Posisi Robot Pada Sumbu X ($K_p=2350$ dan $K_d=1300$)

Pada gambar 5.4 memperlihatkan bahwa akselerasi robot cukup stabil dan konstan. Hal ini terlihat dari bentuk grafik yang membentuk garis lurus yang cukup linier dan respon terhadap set point 1 meter juga terlihat bagus. Hal ini menandakan bahwa penggunaan K_p dan K_d tersebut cukup baik diterapkan pada pengendalian robot. Dengan demikian, dapat disimpulkan bahwa nilai K_p yang digunakan adalah 2350 dan K_d yang digunakan adalah 1300.

Berdasarkan pengujian nilai K_p dan K_d yang dilakukan dapat disimpulkan bahwa pengaruh nilai K_p dan K_d terhadap pergerakan robot penerima tamu adalah :

- a. Konstanta K_p mempengaruhi waktu respon sistem, dimana nilai K_p yang rendah akan memperlambat respon sistem sedangkan nilai K_p yang terlalu tinggi mampu mempercepat respon sistem, namun jika pengendali D bersifat resesif, maka akan meningkatkan osilasi. Hal ini sesuai dengan teori bahwa pengendali P akan mengendalikan kecepatan robot dengan mengkompensasi eror kecepatan yang terjadi
- b. Konstanta K_d mempengaruhi besarnya *overshoot* pada sistem, dimana nilai K_d yang rendah akan meningkatkan *overshoot* sedangkan nilai K_d yang tinggi akan mengurangi *overshoot* yang ditandai dengan gerakan osilasi, namun jika nilai K_d terlalu tinggi membuat gerakan robot seperti tersendat (*error rate* tinggi)
- c. Penggunaan pengendali PD yang tepat pada robot mampu meningkatkan waktu respon dan mengurangi tingkat *overshoot*, sehingga robot dapat bergerak lebih stabil.

5.4 Pengujian Trajektori Dasar Dengan Pengendali PD

Pengujian ini dilakukan dengan tujuan untuk mengetahui kemampuan robot dalam bernavigasi dan mengetahui besarnya slip yang terjadi pada roda robot. Pengujian gerakan dasarnya, yaitu melalui trajektori lurus dan putar, karena kedua gerakan tersebut adalah gerakan dasar untuk bernavigasi. Trajektori lurus dilakukan berdasarkan sumbu x dan gerakan putar mengkombinasikan sumbu x dan sumbu y untuk menghasilkan sudut ψ tertentu. Oleh karena itu, pengendalian PD dilakukan pada posisi x , y dan ψ , yaitu dengan memberikan

nilai konstanta K_p dan K_d pada ketiga parameter tersebut. Pengujian ini dilakukan di lantai berkeramik. Pada pengujian ini diplot grafik posisi yang menandakan posisi pergerakan robot pada pusat koordinat yang terletak di depan robot yang bernilai setengah dari lebar robot.

5.4.1 Pengujian Trajektori Lurus

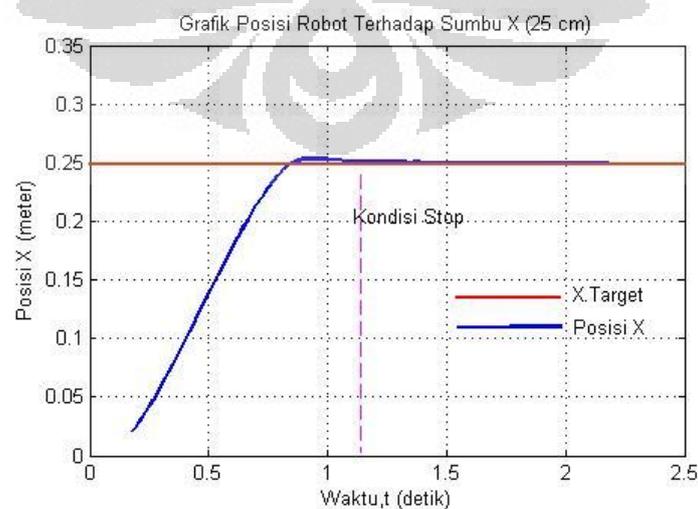
Pada bab 4 telah dijelaskan tentang trajektori lurus yang digunakan pada robot penerima tamu. Pengujian ini dilakukan sebanyak tiga kali, yaitu pada jarak 25 cm, 83 cm dan 300 cm. Pemvariasian jarak yang dilakukan adalah untuk mengetahui seberapa besar tingkat kesalahan robot pada variasi jarak, baik jarak dekat maupun jarak jauh. Langkah yang dilakukan adalah dengan menginisialisasi posisi x_{target} sebesar 0.25, 0.83 dan 3 untuk membuat robot bergerak lurus sepanjang 25 cm, 83 cm dan 3 meter.

5.4.1.1 Hasil Pengujian Trajektori Lurus

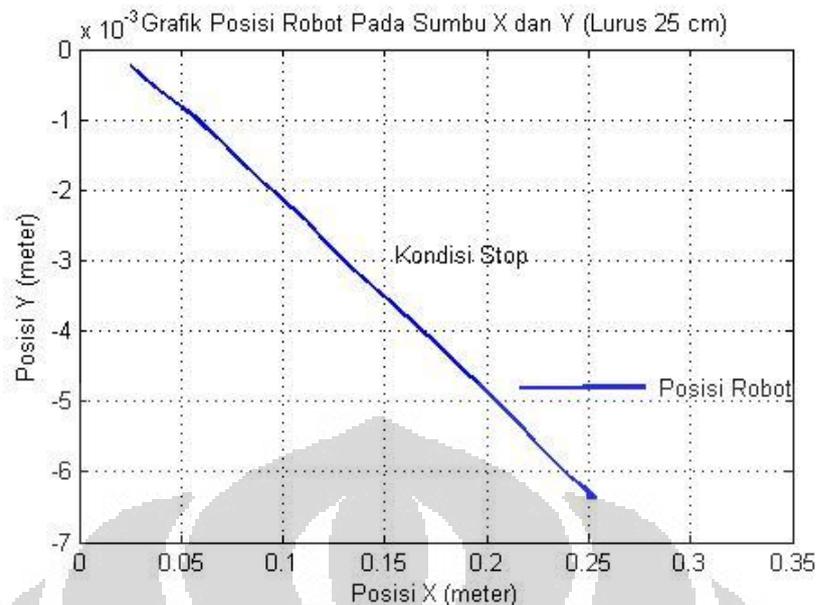
a. Percobaan lurus 25 cm



Gambar 5.5 Validasi dan Error Trajektori Lurus 25 cm



Gambar 5.6 Grafik Kecepatan Trajektori Lurus 25 cm

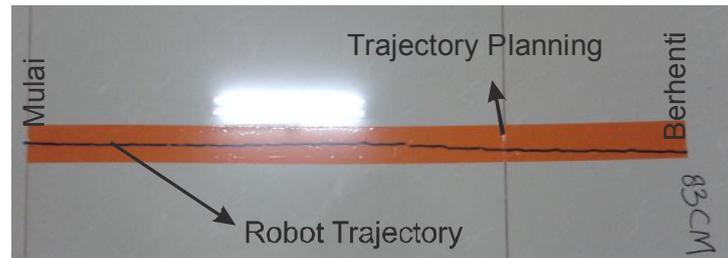


Gambar 5.7 Grafik Posisi X-Y Trajektori Lurus 25 cm

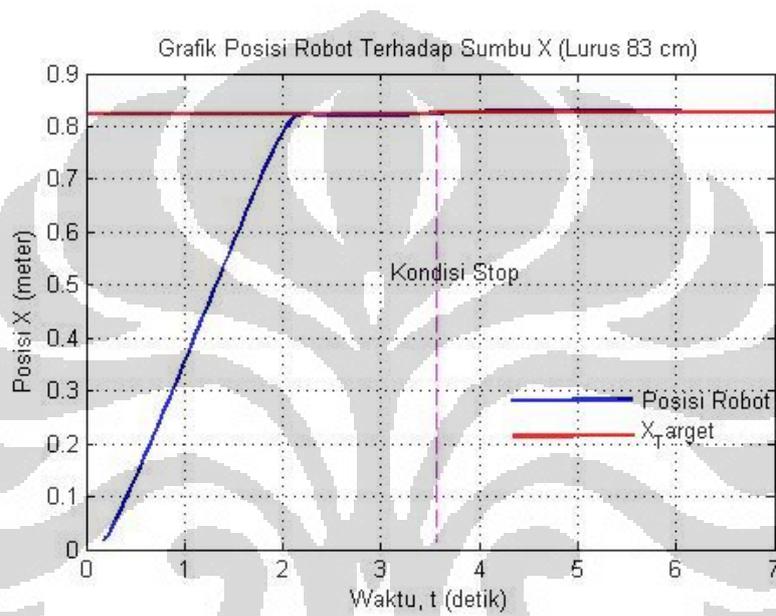
Pada percobaan ini tampak bahwa robot mampu bergerak lurus sejauh 25 cm dengan nilai eror yang kecil, yaitu sebesar 2 mm atau sebesar 0.8 % dari 25 cm. Gambar 5.6 juga menggambarkan kecepatan robot yang konstan untuk menuju posisi target, namun terjadi sedikit overshoot pada puncak 0.253497 m. Overshoot yang terjadi sebesar 1.39 %. Overshoot terjadi akibat nilai nilai K_p dan K_d yang diberikan ternyata belum sangat tepat, namun robot masih mampu mencapai target dengan baik. Hal ini menandakan bahwa robot mampu bergerak pada jarak dekat dengan nilai K_p dan K_d tersebut.

Pada gambar 5.7 diilustrasikan posisi robot pada koordinat Cartesian. Pada gambar tersebut terlihat bahwa robot tidak bergerak lurus, namun belok ke kiri. Hal ini juga terlihat pada gambar 5.5 yang menunjukkan trajektori yang dibentuk oleh robot terlihat lurus ke arah kanan. Hal ini disebabkan karena besarnya kecepatan putar antara kedua roda yang berbeda yang menyebabkan robot bergerak miring. Oleh karena itu, diperlukan suatu pengendali kecepatan untuk mengendalikan kecepatan kedua motor. Namun, dalam skripsi ini robot belum menggunakan pengendali kecepatan karena robot dirasakan masih mampu mengikuti trajektori yang diberikan.

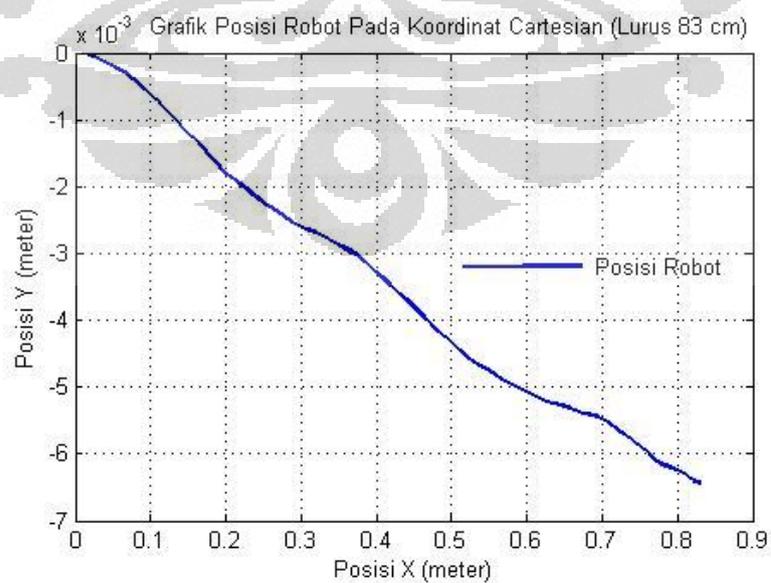
b. Percobaan lurus 83 cm



Gambar 5.8 Validasi Trajektori Lurus 83 cm



Gambar 5.9 Grafik Kecepatan Trajektori Lurus 83 cm

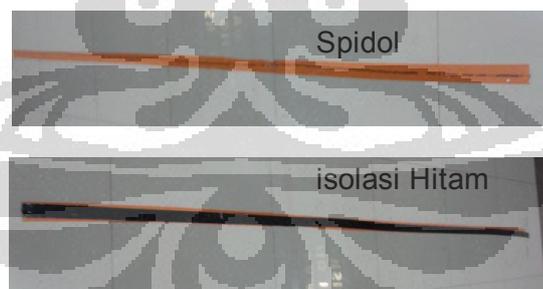


Gambar 5.10 Grafik Posisi Trajektori Lurus 83 cm

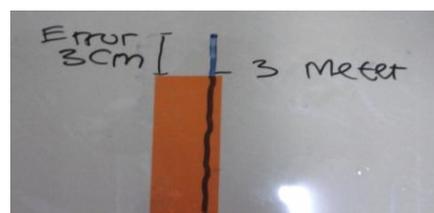
Pada percobaan ini tampak bahwa robot mampu bergerak lurus sejauh 83 cm dengan nilai eror mendekati nol. Gambar 5.9 juga menggambarkan kecepatan robot yang konstan untuk menuju posisi target, namun terdapat overshoot yang sangat sedikit pada puncak 0.831150 m. Overshoot yang terjadi sebesar 0.13 %. Overshoot terjadi akibat nilai nilai K_p dan K_d yang diberikan ternyata belum sangat tepat, namun robot masih mampu mencapai target dengan baik. Hal ini menandakan bahwa robot juga mampu bergerak pada jarak 83 cm.

Pada gambar 5.10 diilustrasikan posisi robot pada koordinat Cartesian. Pada gambar tersebut terlihat bahwa robot tidak bergerak lurus, namun belok ke kiri. Hal ini juga terlihat pada gambar 5.8 yang menunjukkan trajektori yang dibentuk oleh robot terlihat lurus ke arah kanan. Hal ini disebabkan karena besarnya kecepatan putar antara kedua roda yang berbeda yang menyebabkan robot bergerak miring. Dalam hal ini motor kanan sedikit lebih cepat dari motor kiri. Oleh karena itu, diperlukan suatu pengendali kecepatan untuk mengendalikan kecepatan kedua motor. Namun, dalam skripsi ini robot belum menggunakan pengendali kecepatan karena robot dirasakan masih mampu mengikuti trajektori yang diberikan.

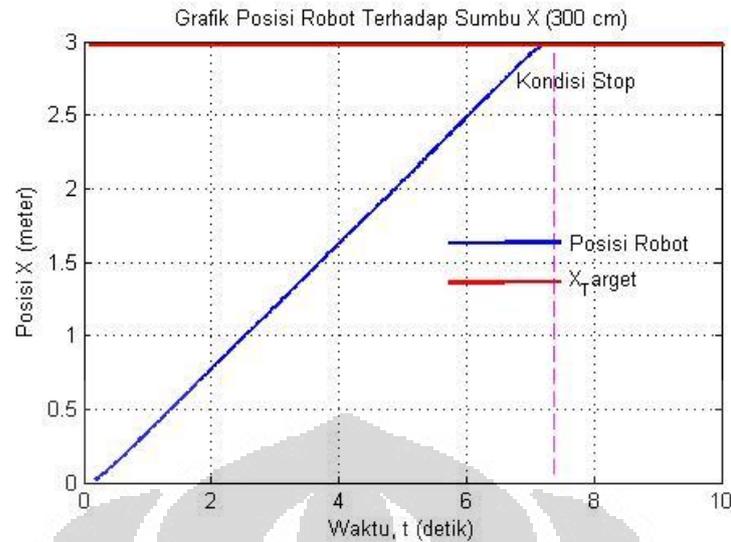
c. Percobaan lurus 300 cm



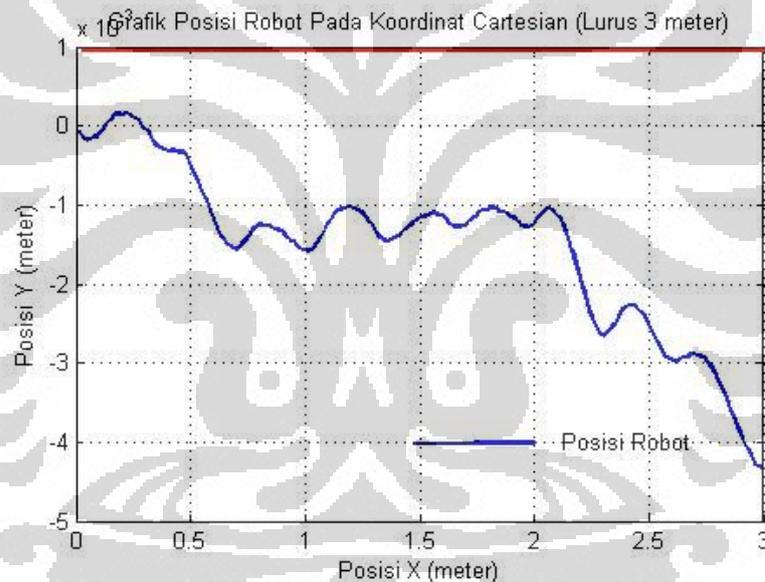
Gambar 5.11 Validasi Trajektori Lurus 300 cm



Gambar 5.12 Error yang Terjadi 300 cm



Gambar 5.13 Grafik Kecepatan Trajektori Lurus 300 cm



Gambar 5.14 Grafik Posisi Trajektori Lurus 300 cm

Pada percobaan ini tampak bahwa robot mampu bergerak lurus sejauh 83 cm dengan nilai eror mendekati nol. Gambar 5.13 menggambarkan kecepatan robot yang konstan untuk menuju posisi target sebesar 3 meter dan tidak terjadi overshoot. Hal ini diperlihatkan dari puncak grafik yang bernilai 2.997025, sehingga masih terdapat error sebesar 0.01 %. Jika dibandingkan dengan data pada navigasi nyata pada gambar 5.12 robot memiliki eror sebesar 3 cm atau 1%. Hal ini terjadi akibat terdapatnya slip pada robot dan kemampuan robot untuk menyamakan posisinya akibat adanya perbedaan kecepatan antara kedua motor

akibatnya saat motor kanan telah mencapai target, motor kiri belum mencapai target sehingga masih berputar dan mengakibatkan roda yang berhenti menjadi tergeser. Namun, nilai eror sebesar 1% tersebut dinilai masih cukup aman untuk navigasi robot. Hal ini menandakan bahwa robot juga mampu bergerak pada jarak 3 meter.

Pada gambar 5.14 diilustrasikan posisi robot pada koordinat Cartesian. Pada gambar tersebut terlihat bahwa robot tidak bergerak lurus, namun bergerak berbelok-belok ke arah kanan. Hal ini juga terlihat pada gambar 5.11 yang menunjukkan trajektori yang dibentuk oleh robot terlihat lurus ke arah kanan dengan trajektori agak berbelok-belok. Hal ini disebabkan karena besarnya kecepatan putar antara kedua roda yang berbeda dan besarnya slip yang terjadi semakin besar saat jarak target semakin jauh. Oleh karena itu, diperlukan suatu pengendali kecepatan untuk mengendalikan kecepatan kedua motor untuk memperkecil besar eror yang terjadi.

5.4.1.2 Analisis Pengujian Trajektori Lurus

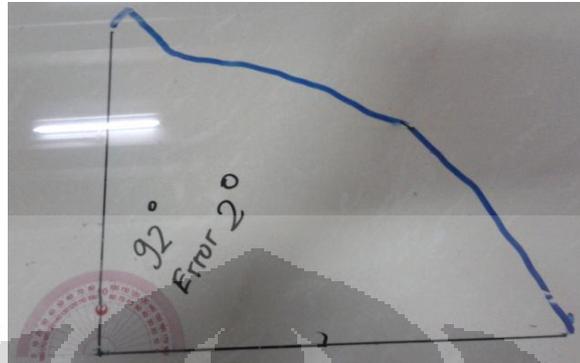
Berdasarkan data tersebut di atas dapat disimpulkan bahwa navigasi dasar robot untuk trajektori lurus sudah cukup baik. Hal ini juga terlihat dari trajektori yang diperlihatkan oleh spidol yang hampir membentuk suatu garis lurus, namun tetap memiliki eror yang disebabkan akibat terjadinya slip dan perbedaan kecepatan putar dari kedua motor. Salah satu cara untuk memperkecil eror adalah dengan menggunakan pengendali kecepatan pada kedua motor, namun untuk navigasi robot penerima tamu yang dirancang belum menggunakan pengendali kecepatan.

5.4.2 Pengujian Trajektori Putar

Pada bab 4 telah dijelaskan tentang trajektori putar yang digunakan pada robot penerima tamu. Pengujian ini dilakukan sebanyak empat kali, yaitu pada putaran sebesar 0.5π radian, π radian, 1.5π radian dan 2π radian. Pemvariasian derajat putar yang dilakukan adalah untuk mengetahui seberapa besar tingkat kesalahan dan slip yang terjadi saat robot berputar karena saat berputar akan memungkinkan terjadinya slip yang lebih besar dibandingkan dengan jarak lurus.

5.4.2.1 Hasil Pengujian Trajektori Putar

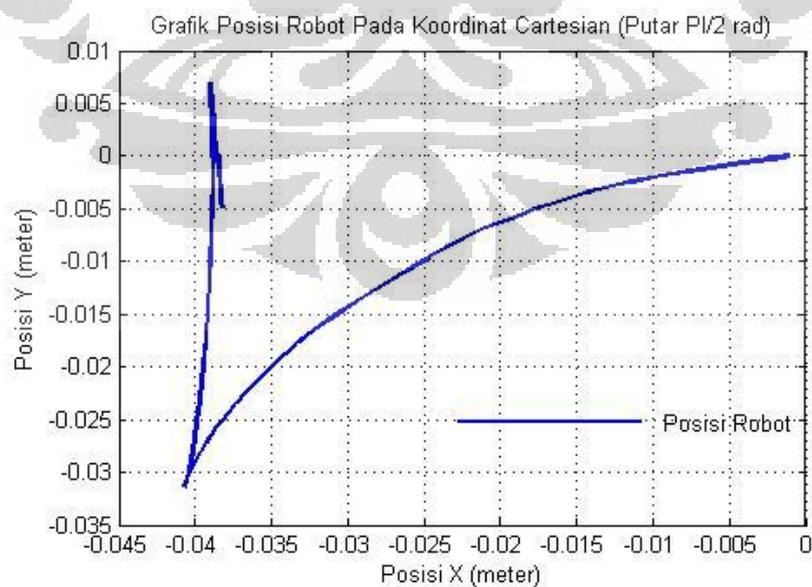
a. Percobaan putar sebesar $\pi/2$ rad



Gambar 5.15 Gerakan dan Error Trajektori Putar $\pi/2$ rad



Gambar 5.16 Posisi Trajektori Putar $\pi/2$ rad



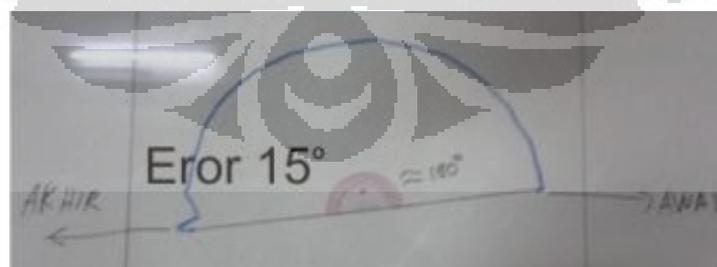
Gambar 5.17 Pergerakan Trajektori Putar $\pi/2$ rad

Pada percobaan ini robot berputar sebesar 90° berlawanan arah jarum jam. Pada percobaan ini robot mampu bergerak sebesar 90° pada data posisi percobaan yang terlihat pada gambar 5.17, namun pada kenyataannya robot berputar sebesar 92° atau mengalami eror sebesar 2° (2.22%). Eror ini disebabkan karena adanya slip pada pergerakan putar robot. Hal ini terlihat dari gambar 5.17 yang menggambarkan posisi robot pada saat berputar. Pada gambar tersebut digambarkan bahwa terdapat perubahan posisi pada menjelang sudut 90° yang disebabkan akibat slip pada roda. Namun, melihat besarnya eror pada robot dan pergerakan robot secara nyata didapatkan kesimpulan bahwa robot mampu berputar sebesar 90° .

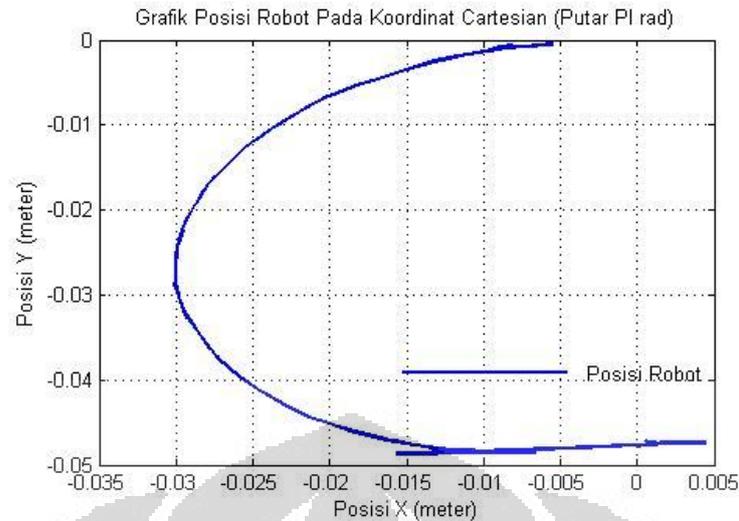
b. Percobaan PI rad



Gambar 5.18 Posisi Trajektori Putar PI rad



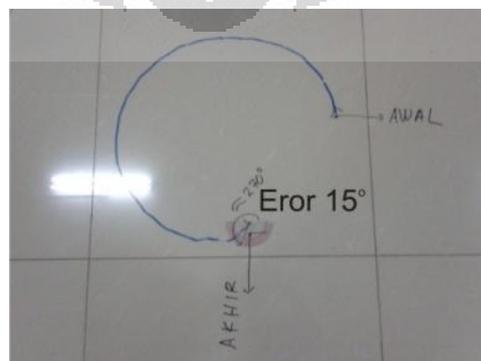
Gambar 5.19 Gerakan dan Eror Trajektori Putar PI rad



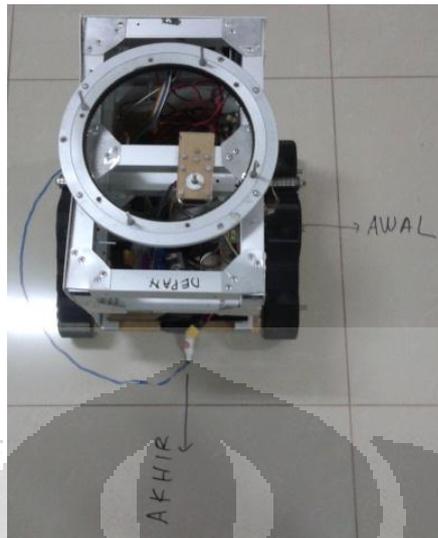
Gambar 5.20 Pergerakan Trajektori Putar PI rad

Pada percobaan ini robot berputar sebesar 180° berlawanan arah jarum jam. Pada percobaan ini robot mampu bergerak sebesar 195° pada data posisi percobaan yang terlihat pada gambar 5.20. Besarnya eror yang terjadi adalah sebesar 15° atau 8.33%. Nilai eror ini lebih disebabkan akibat posisi spidol yang terletak tidak pada koordinat pusat robot sehingga garis yang terbentuk oleh spidol memiliki posisi awal yang agak miring. Akibatnya trajektori yang terbentuk tidak membentuk 180° . Selain itu besarnya slip yang terjadi juga mampu mempengaruhi besarnya sudut putar robot, serta adanya nilai K_p dan K_d yang benar-benar belum tepat sehingga terjadi sedikit overshoot pada trajektorinya. Namun, dengan eror sebesar 8.33% yang dapat dianggap sebagai eror maksimum, dapat dinyatakan bahwa robot mampu berputar sebesar 180° .

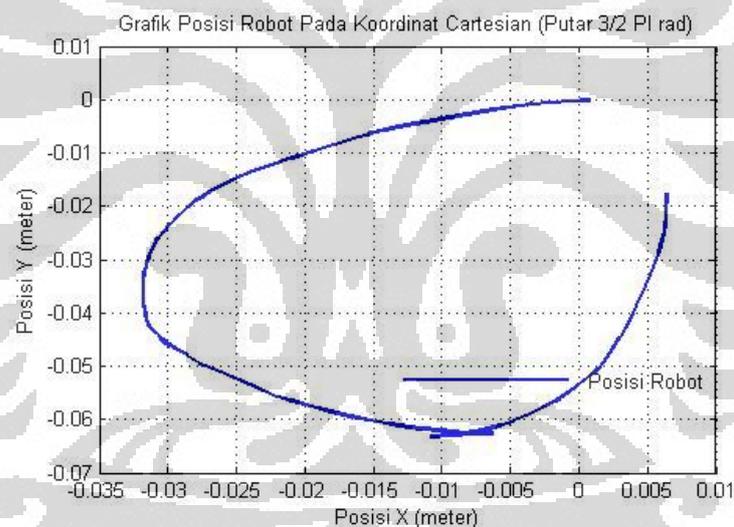
c. Percobaan $3/2$ PI rad



Gambar 5.21 Pergerakan dan Eror Trajektori Putar $3/2$ PI rad



Gambar 5.22 Posisi Trajektori Putar $3/2 \text{ PI rad}$

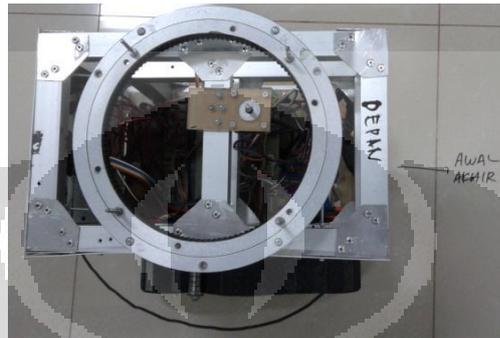


Gambar 5.23 Grafik Posisi untuk 2 PI rad

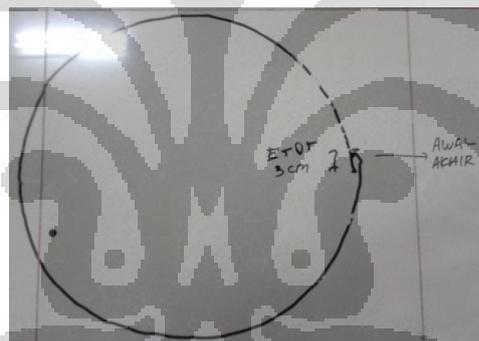
Pada percobaan ini robot berputar sebesar 270° berlawanan arah jarum jam. Pada percobaan ini robot mampu bergerak sebesar 285° pada data posisi percobaan yang terlihat pada gambar 5.23. Besarnya eror yang terjadi adalah sebesar 15° atau 8.33%. Nilai eror ini lebih disebabkan akibat posisi spidol yang terletak tidak pada koordinat pusat robot sehingga garis yang terbentuk oleh spidol memiliki posisi awal yang agak miring. Akibatnya trajektori yang terbentuk tidak membentuk 270° . Selain itu besarnya slip yang terjadi juga mampu mempengaruhi besarnya sudut putar robot, hal ini terlihat dari bentuk

perputaran robot yang tidak melingkar pada gambar 5.23. Selain itu adanya nilai K_p dan K_d yang benar-benar belum tepat sehingga terjadi sedikit overshoot pada tajektorinya. Namun, dengan eror sebesar 8.33% yang dapat dianggap sebagai eror maksimum, dapat dinyatakan bahwa robot mampu berputar sebesar 270° .

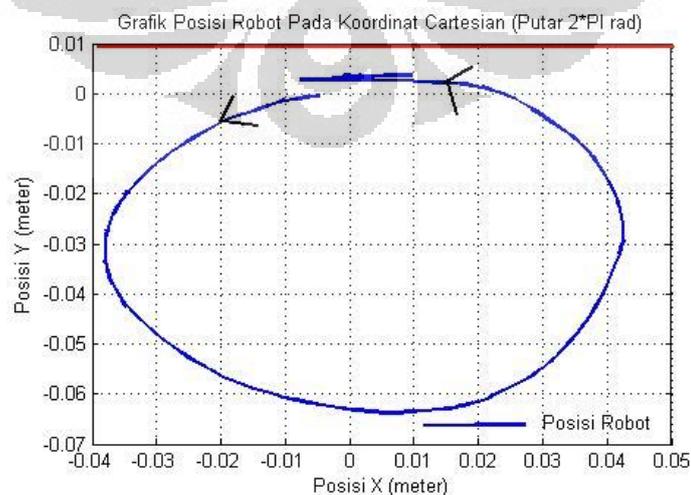
d. Percobaan 2 PI rad



Gambar 5.24 Posisi Trajektori Putar 2 PI rad



Gambar 5.25 Pergerakan dan Error Trajektori Putar 2 PI rad



Gambar 5.26 Grafik Posisi untuk 2 PI rad

Pada percobaan ini robot berputar sebesar 360° berlawanan arah jarum jam. Pada percobaan ini robot mampu bergerak sebesar 375° pada data posisi percobaan yang terlihat pada gambar 5.26. Besarnya eror yang terjadi adalah sebesar 15° atau 8.33%. Nilai eror ini lebih disebabkan akibat posisi spidol yang terletak tidak pada koordinat pusat robot sehingga garis yang terbentuk oleh spidol memiliki posisi awal yang agak miring. Akibatnya trajektori yang terbentuk tidak membentuk 270° . Selain itu besarnya slip yang terjadi juga mampu mempengaruhi besarnya sudut putar robot, serta adanya nilai Kp dan Kd yang benar-benar belum tepat sehingga terjadi sedikit overshoot pada trajektorinya. Namun, dengan eror sebesar 8.33% yang dapat dianggap sebagai eror maksimum, dapat dinyatakan bahwa robot mampu berputar sebesar 270° .

5.4.2.2 Analisis Pengujian Trajektori Putar

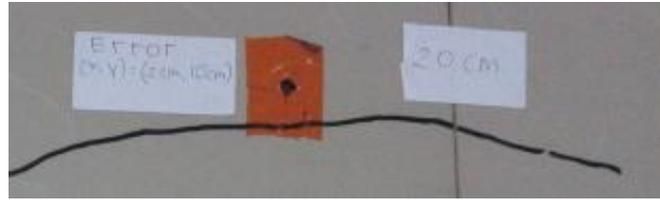
Berdasarkan data hasil percobaan di atas dapat disimpulkan bahwa robot mampu melakukan pergerakan dengan trajektori putar dengan baik meskipun masih terdapat eror yang kecil. Eror yang terjadi memiliki eror maksimum sebesar 8.33% dan dirasa robot masih mampu mengikuti trajektori putar. Eror yang terjadi disebabkan oleh slip akibat kurang baiknya perancangan mekanik dalam perodaan yang berbentuk tank dan kecepatan putar pada kedua roda yang tidak sama.

5.4.3 Pengujian Trajektori Elip

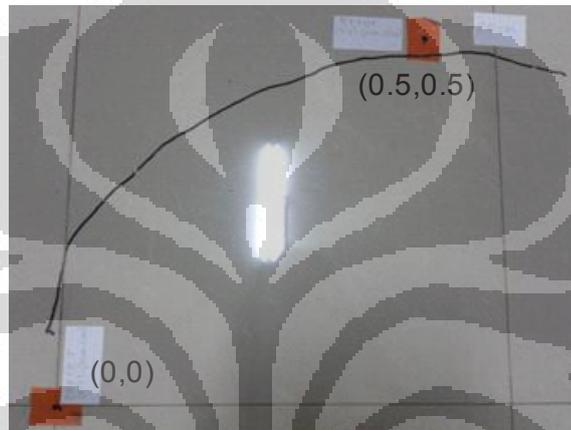
Pada percobaan ini robot bernavigasi dengan membentuk trajektori elip berdasarkan fungsi elip yang telah dijelaskan pada bab 4. Percobaan dilakukan pada koordinat (0.5,0.5) dan (1,2) untuk trajektori elip ke kiri dan ke kanan. Pada percobaan digunakan isolasi berwarna jingga sebagai penanda koordinat awal dan koordinat target dari trajektori robot elip dari robot penerima tamu. Robot diharapkan mampu berhenti di koordinat target dengan baik. Berikut data hasil percobaan yang dilakukan :

5.4.3.1 Hasil Pengujian Trajektori Elip

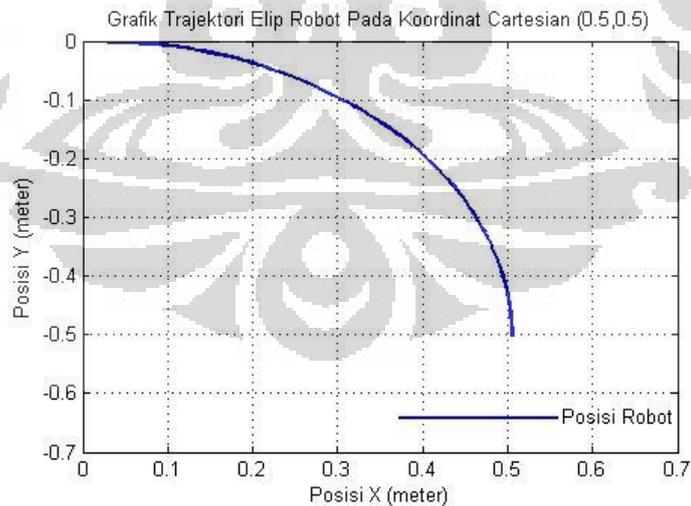
a. Percobaan koordinat (0.5,0.5)



Gambar 5.27 Error Trajektori Elip Koordinat (0.5,0.5)



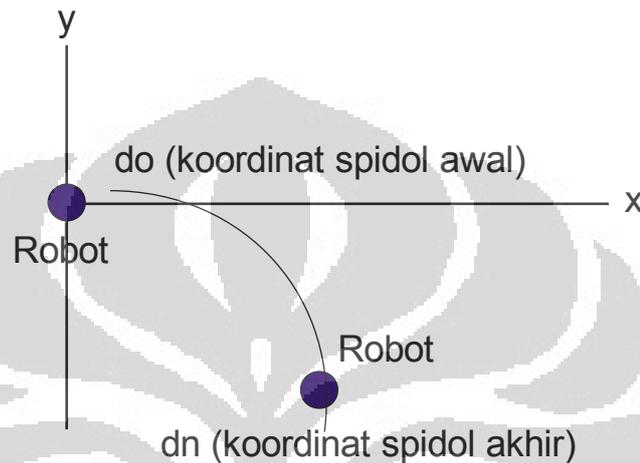
Gambar 5.28 Trajektori Elip Pada Koordinat (0.5,0.5)



Gambar 5.29 Grafik Trajektori Elip di Koordinat (0.5,0.5)

Pada percobaan ini robot bergerak membentuk kurva elip dan berhenti saat koordinat (0.5,0.5). Pada gambar 5.28 terlihat bahwa robot berhasil

mendekati koordinat target, namun ternyata terdapat kelebihan trajektori sebesar 20 cm. Hal ini disebabkan karena kondisi spidol saat pengambilan data tidak terletak pada koordinat pusat robot, akibatnya spidol mencatat trajektori berlebih. Posisi spidol adalah sekitar (x_0, y_0) dari koordinat pusat. Berikut penggambarannya :

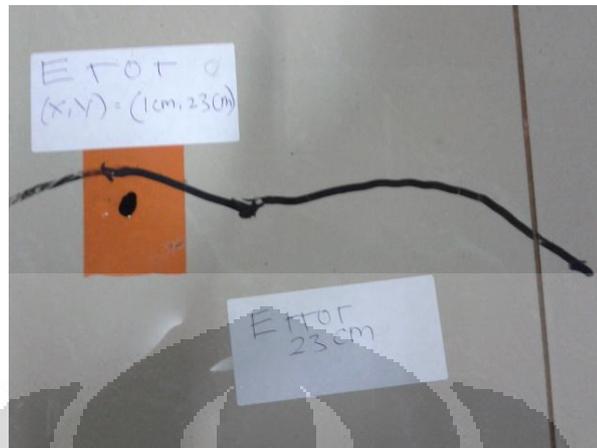


Gambar 5.30 Koordinat Spidol Terhadap Robot

Akibatnya, trajektori robot yang terukir memiliki kelebihan sebesar koordinat d_0 (x_0, y_0) , dimana (x_0, y_0) adalah $(0.01, 0.1)$. Dengan demikian didapatkan besarnya eror yang sebenarnya adalah $(0 \text{ cm}, (23-10) \text{ cm})$, yaitu sebesar 13 cm. Besarnya eror yang terjadi diakibatkan oleh slip dan penggunaan nilai K_p dan K_d untuk posisi sudut ($K_p\phi$ dan $K_d\dot{\phi}$) yang masih kurang tepat, namun dapat K_p tersebut masih dapat digunakan karena berhasil mendekati target dan hal ini masih memungkinkan untuk dikompensasi.

Fungsi elip yang dibuat berhasil diterapkan dalam navigasi dasar robot penerima tamu, terlihat dari gambar 5.28 dan 5.29 yang mampu menghasilkan trajektori elip dan mampu mendekati target dengan cukup baik. Dengan demikian dapat dinyatakan bahwa robot mampu melakukan navigasi untuk jarak dekat.

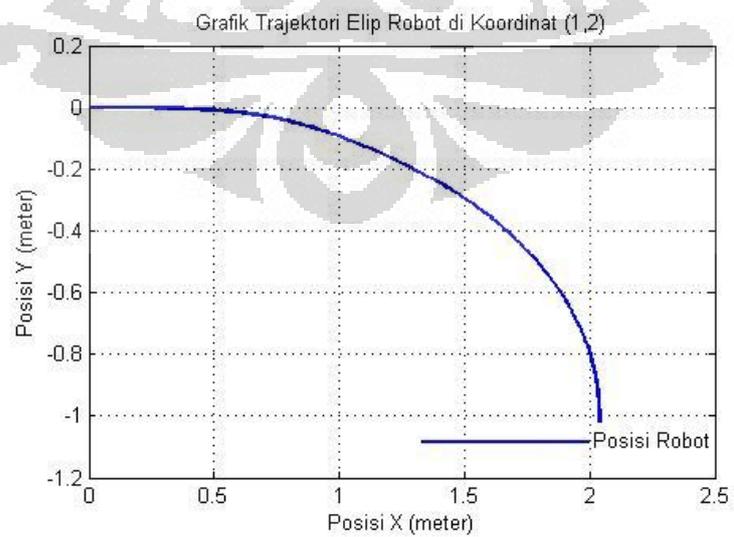
b. Percobaan koordinat (1,2).



Gambar 5.31 Error Trajektori Elip Koordinat (1,2)



Gambar 5.32 Trajektori Elip Pada Koordinat (1,2)

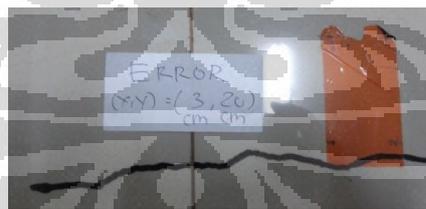


Gambar 5.33 Grafik Trajektori Elip di Koordinat (1,2)

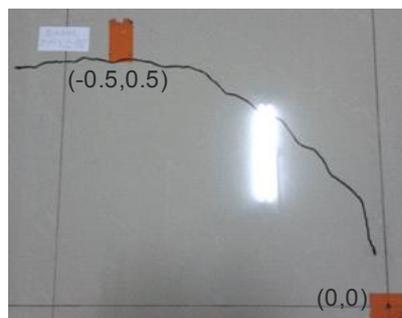
Pada percobaan ini robot bergerak membentuk kurva elip dan berhenti saat koordinat (1,2). Pada gambar 5.31 terlihat bahwa robot berhasil mendekati koordinat target, namun ternyata terdapat kelebihan trajektori sebesar 23 cm. Hal ini disebabkan karena kondisi spidol saat pengambilan data tidak terletak pada koordinat pusat robot, akibatnya spidol mencatat trajektori berlebih. Posisi spidol adalah sekitar (x_0, y_0) dari koordinat pusat. Akibatnya, trajektori robot yang terukur memiliki kelebihan sebesar koordinat $d_0(x_0, y_0)$, dimana (x_0, y_0) adalah $(0.01, 0.1)$. Dengan demikian didapatkan besarnya eror yang sebenarnya adalah $(0 \text{ cm}, (23-10) \text{ cm})$, yaitu sebesar 13 cm. Besarnya eror yang terjadi diakibatkan oleh slip dan penggunaan nilai K_p dan K_d untuk posisi sudut ($K_p\phi$ dan $K_d\dot{\phi}$) yang masih kurang tepat, namun dapat K_p tersebut masih dapat digunakan karena berhasil mendekati target dan hal ini masih memungkinkan untuk dikompensasi.

Fungsi elip yang dibuat berhasil diterapkan dalam navigasi dasar robot penerima tamu, terlihat dari gambar 5.32 dan 5.33 yang mampu menghasilkan trajektori elip dan mampu mendekati target dengan cukup baik. Dengan demikian dapat dinyatakan bahwa robot mampu melakukan navigasi untuk koordinat tersebut.

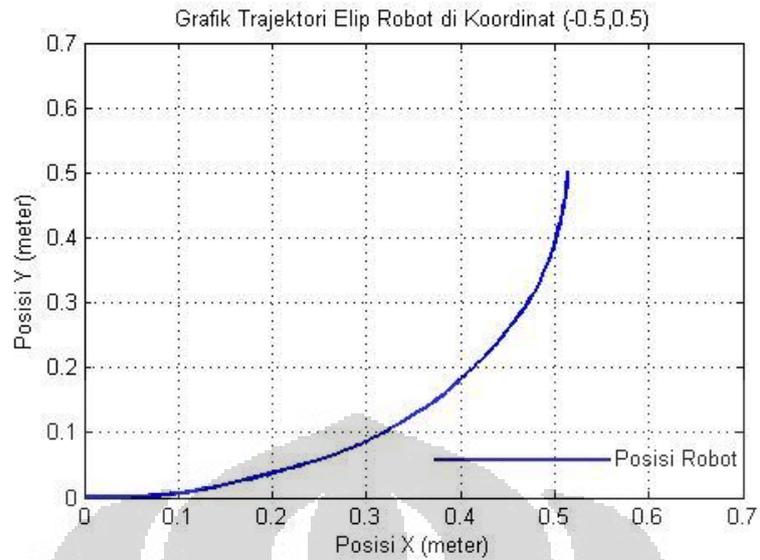
c. Percobaan koordinat $(-0.5, 0.5)$



Gambar 5.34 Eror Trajektori Elip Koordinat $(-0.5, 0.5)$



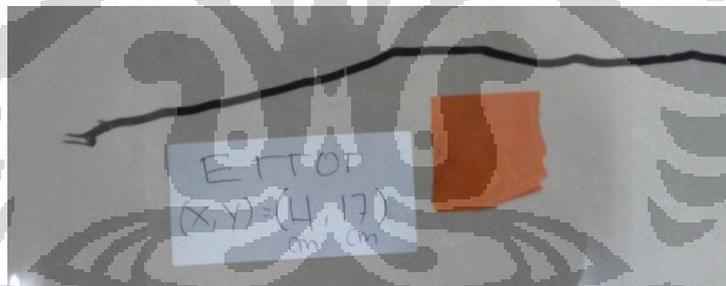
Gambar 5.35 Trajektori Elip Pada Koordinat $(-0.5, 0.5)$



Gambar 5.36 Grafik Trajektori Elip di Koordinat (-0.5,0.5)

Pada percobaan ini besarnya eror yang terjadi juga disebabkan akibat masih kurang tepatnya pemilihan $K_p\phi$ dan $K_d\phi$ untuk pengendali posisi sudut dan akibat terjadinya slip.

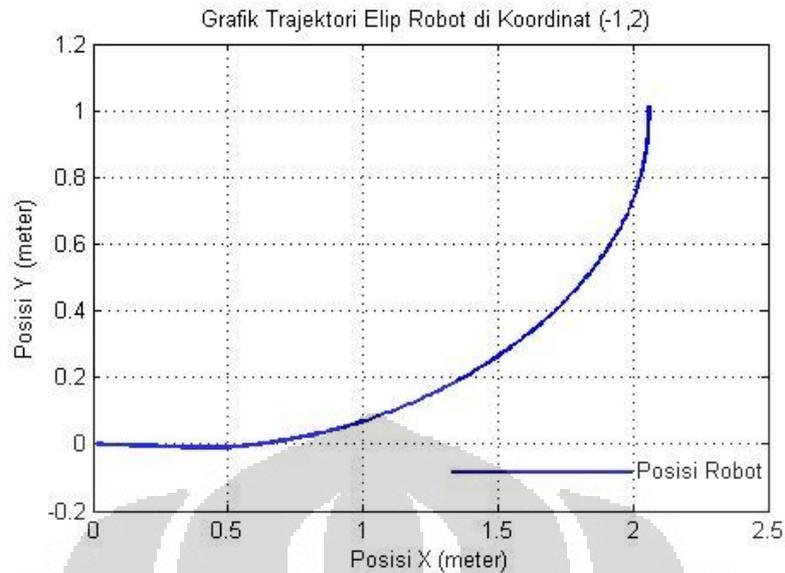
d. Percobaan koordinat (-1,2)



Gambar 5.37 Error Trajektori Elip Koordinat (-1,2)



Gambar 5.38 Trajektori Elip Pada Koordinat (-1,2)



Gambar 5.39 Grafik Trajektori Elip di Koordinat (-1,2)

Pada percobaan ini besarnya eror yang terjadi juga disebabkan akibat masih kurang tepatnya pemilihan $K_{p\phi}$ dan $K_{d\phi}$ untuk pengendali posisi sudut dan akibat terjadinya slip.

5.4.3.2 Analisis Pengujian Trajektori Elip

Pada percobaan ini robot dinyatakan berhasil melakukan navigasi dengan trajektori elip, namun masih belum mendapatkan konstanta pengendali PD untuk sudut yang sangat tepat. Selain itu, kemungkinan slip yang terjadi akan semakin besar untuk trajektori melingkar sehingga eror yang akan terjadi juga akan menjadi besar. Untuk mendapatkan hasil yang lebih baik dapat digunakan pengendalian secara dinamik saat robot bergerak, yaitu melalui pengendalian gaya gravitasi dan momen inersia dari robot, namun dalam skripsi ini pengendalian secara dinamik masih belum digunakan.

5.5 Pengujian Navigasi Robot Penerima Tamu

Pada percobaan ini robot melakukan navigasi untuk menghampiri suatu objek berdasarkan koordinat tertentu dengan arah orientasi ke arah robot. Penjelasan lebih jelas mengenai trajektori navigasi dari robot penerima tamu telah dijelaskan pada bab 4. Percobaan ini dilakukan sebanyak empat kali, yaitu dengan memvariasikan koordinat navigasi pada koordinat (0.5,0.5) dan (1,2) untuk

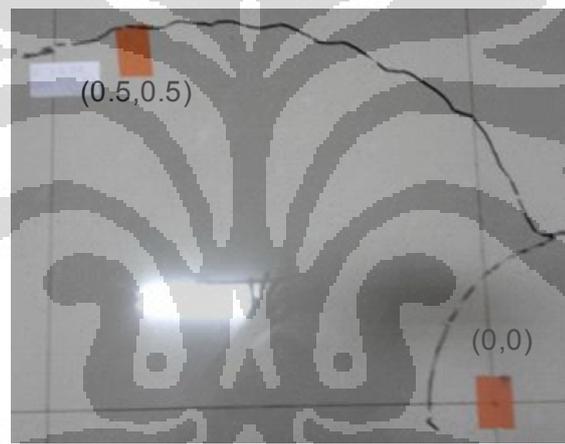
kuadran I dan kuadran II robot. Tujuan pengujian ini adalah untuk mendapatkan trajektori navigasi robot penerima tamu berdasarkan orientasi objek yang akan dituju.

5.5.1 Hasil Pengujian Trajektori Navigasi Robot Penerima Tamu

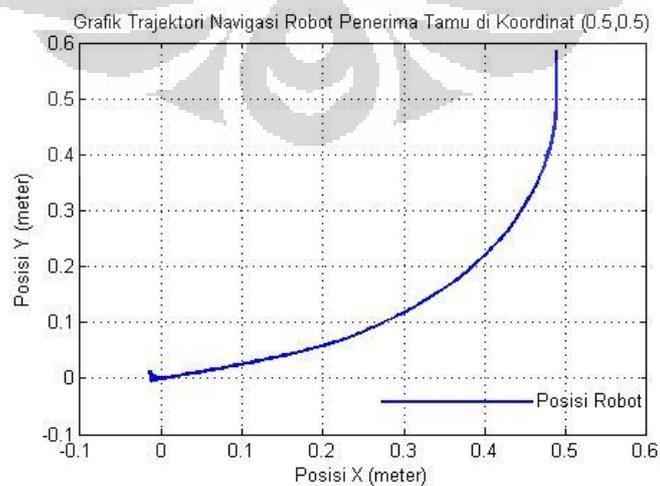
a. Percobaan koordinat (0.5,0.5)



Gambar 5.40 Error Trajektori Navigasi Robot Pada Koordinat (0.5,0.5)



Gambar 5.41 Trajektori Navigasi Robot Pada Koordinat (0.5,0.5)



Gambar 5.42 Grafik Trajektori Navigasi Robot Pada Koordinat (0.5,0.5)

Pada percobaan ini robot dinyatakan mampu mendekati objek yang dituju pada koordinat (0.5,0.5). Hal ini terlihat pada gambar 5.40, 5.41 dan 5.42, dimana baik kurva data maupun kurva aktual membentuk trajektori navigasi yang diinginkan dan juga mampu mendekati kondisi objek, namun masih terdapat eror yang relatif disebabkan oleh slip saat bernavigasi dan pemilihan $K_p\phi$ dan $K_d\phi$ yang masih belum tepat. Dengan demikian sistem navigasi yang dibuat dapat diterapkan pada robot penerima tamu untuk jarak atau koordinat dekat.

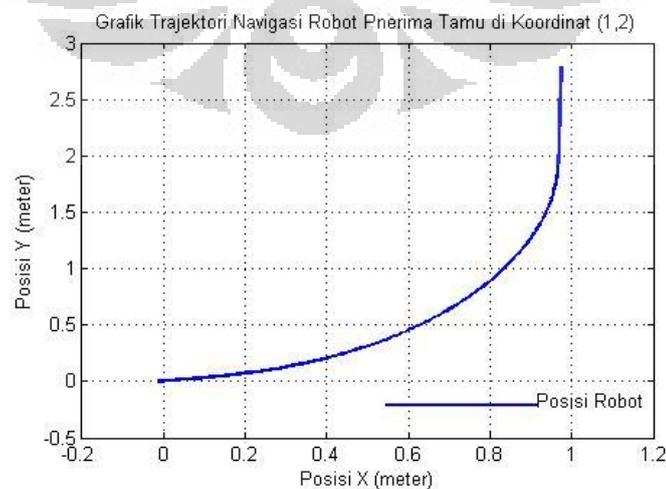
b. Percobaan koordinat (1,2)



Gambar 5.43 Eror Trajektori Navigasi Robot Pada Koordinat (1,2)



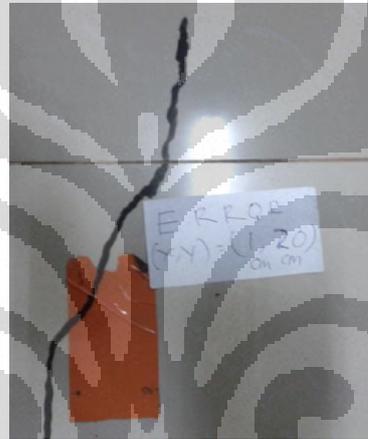
Gambar 5.44 Trajektori Navigasi Robot Pada Koordinat (1,2)



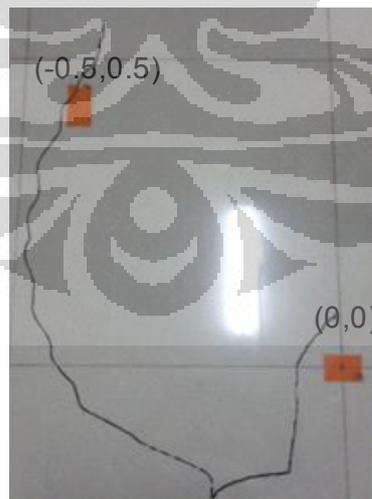
Gambar 5.45 Grafik Trajektori Navigasi Robot Pada Koordinat (1,2)

Untuk percobaan ini, analisis yang didapatkan sama halnya dengan koordinat $(0.5,0.5)$, dimana pada percobaan ini robot dinyatakan mampu mendekati objek yang dituju pada koordinat $(1,2)$. Hal ini terlihat pada gambar 5.43, 5.44 dan 5.45, dimana baik kurva data maupun kurva aktual membentuk trajektori navigasi yang diinginkan dan juga mampu mendekati kondisi objek, namun masih terdapat eror yang relatif disebabkan oleh slip saat bernavigasi. Dengan demikian sistem navigasi yang dibuat dapat diterapkan pada robot penerima tamu untuk jarak atau koordinat dekat.

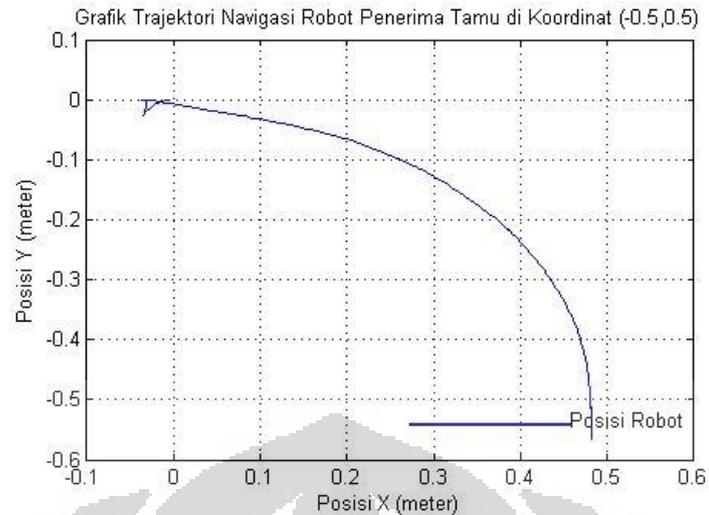
c. Percobaan koordinat $(-0.5,0.5)$



Gambar 5.46 Error Trajektori Navigasi Robot Pada Koordinat $(-0.5,0.5)$



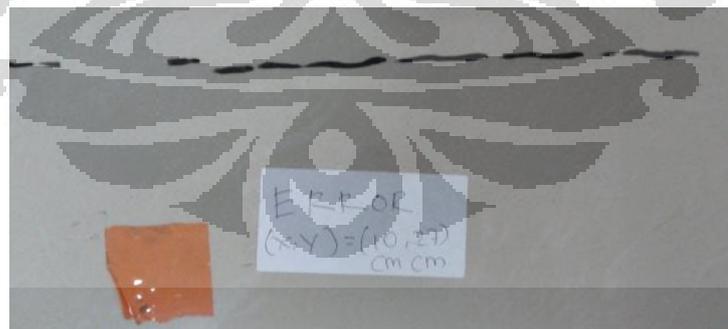
Gambar 5.47 Trajektori Navigasi Robot Pada Koordinat $(-0.5,0.5)$



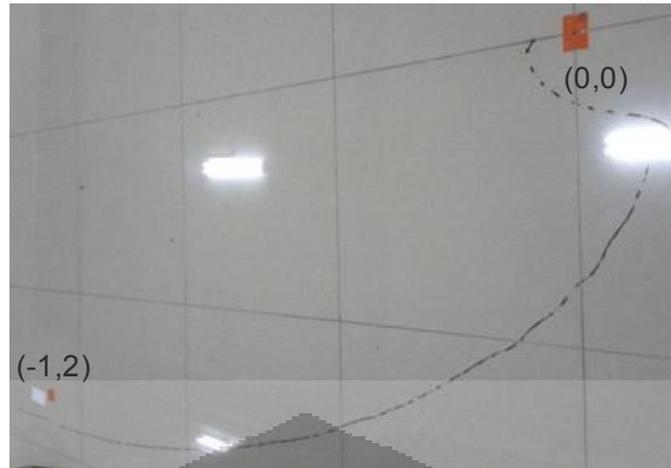
Gambar 5.48 Grafik Trajektori Navigasi Robot Pada Koordinat (-0.5,0.5)

Pada percobaan untuk kuadran II ini juga dapat dinyatakan bahwa robot mampu mendekati target dengan eror posisi sumbu y sebesar 10 cm akibat slip dan pemilihan konstanta pengendali sudut yang masih belum tepat. Hal ini terlihat dari bentuk kurva trajektori yang ditunjukkan pada gambar 5.47 yang terlihat berbelok-belok yang dapat diartikan bahwa pengendali posisi dan sudut yang digunakan masih belum sangat sesuai. Namun, robot masih mampu digunakan untuk bernavigasi dengan jarak dekat pada kuadran II robot.

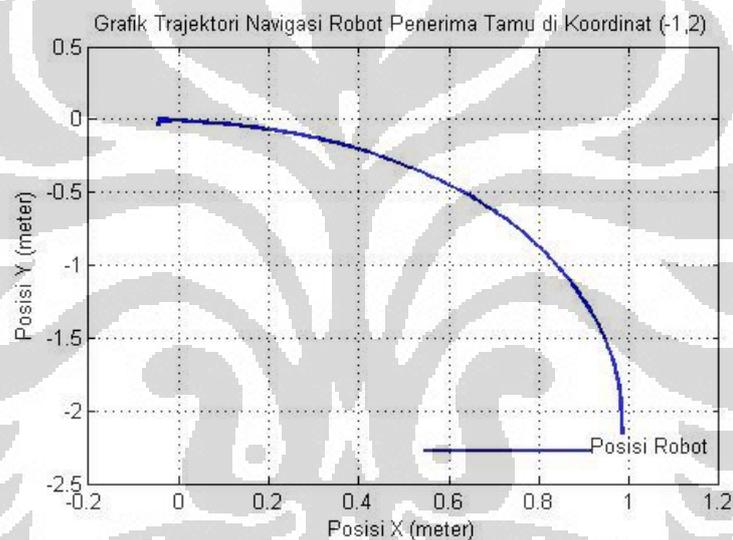
d. Percobaan koordinat (-1,2)



Gambar 5.49 Error Trajektori Navigasi Robot Pada Koordinat (-1,2)



Gambar 5.50 Trajektori Navigasi Robot Pada Koordinat (-1,2)



Gambar 5.51 Grafik Trajektori Navigasi Robot Pada Koordinat (-0.5,0.5)

Untuk percobaan ini, robot juga mampu mengikuti trajektori yang diberikan terlihat dari bentuk kurva yang dibentuk untuk kurva aktual pada gambar 5.50 dan kurva data pada gambar 5.51 yang sama-sama membentuk trajektori navigasi sesuai yang diharapkan.

5.5.2 Analisis Pengujian Trajektori Navigasi Robot Penerima Tamu

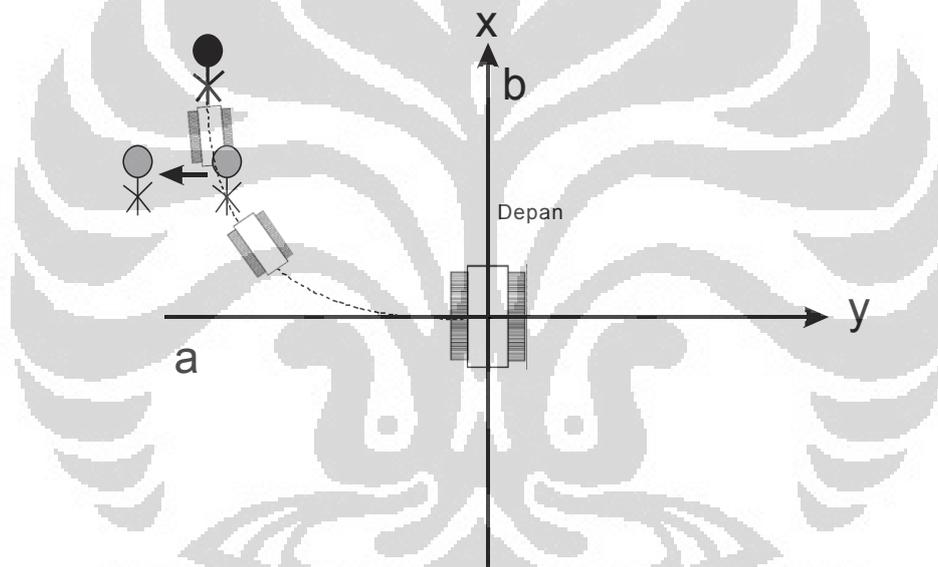
Berdasarkan keempat percobaan yang dilakukan dapat disimpulkan bahwa robot penerima tamu mampu melakukan navigasi dengan baik dengan eror yang masih bias ditoleransi dan dikompensasi dengan menggunakan sensor tambahan seperti sensor jarak agar robot tidak menabrak objek. Besarnya eror

yang terjadi tidak terlepas akibat adanya slip, perbedaan kecepatan putar motor dan konstanta pengendali sudut yang masih belum sesuai.

5.6 Pengujian Navigasi Robot Penerima Tamu Dengan SRF-08

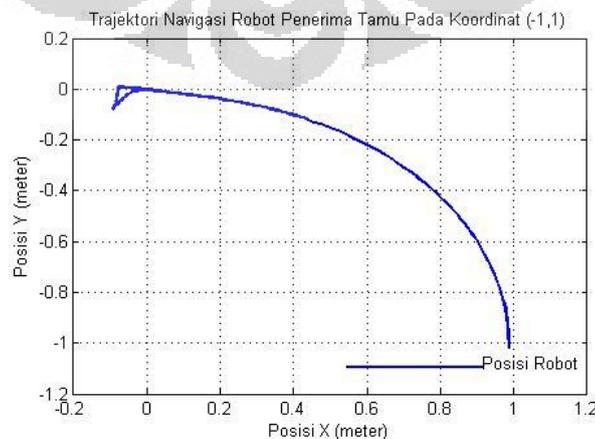
Pengujian ini dilakukan dengan melakukan navigasi pada robot penerima tamu pada koordinat (-1,1) untuk menuju objek. Saat terdapat orang lewat, maka robot akan berhenti sejenak hingga orang tersebut sudah tidak terdeteksi berada di depan robot, kemudian robot akan melanjutkan navigasinya hingga ke koordinat yang dituju.

Berikut ilustrasi navigasi robot penerima tamu dengan SRF-08 :

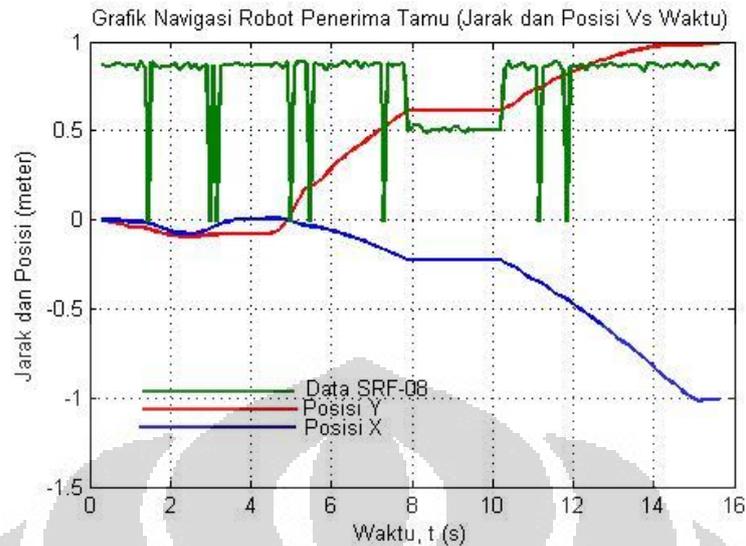


Gambar 5.52 Trajektori Navigasi Robot Penerima Tamu Dengan SRF-08

5.6.1 Hasil Pengujian Navigasi Robot Penerima Tamu Dengan SRF-08



Gambar 5.53 Grafik Trajektori Navigasi Robot Pada Koordinat (-1,1)



Gambar 5.54 Grafik Jarak dan Posisi Terhadap Waktu

5.6.2 Analisis Pengujian Navigasi Robot Penerima Tamu Dengan SRF-08

Pada percobaan ini robot cukup berhasil bernavigasi untuk mendekati objek yang dituju dan ketika ada seseorang berjalan di depannya, maka robot akan berhenti dan melanjutkan navigasinya kembali menuju objek yang dituju.

Pada gambar 5.54 yang menggambarkan grafik posisi dan jarak terhadap waktu dinyatakan bahwa pada detik ke 7.8 hingga 10.1 terdapat orang sedang berjalan dan robot melakukan pemberhentian, terlihat dari grafik posisi X dan Y yang nilainya konstan dan juga grafik jarak yang cukup konstan. Pada grafik jarak masih terdapat noise untuk datanya sehingga terbaca -1 cm dan robot akan bergerak sesuai dengan kecepatan sebelumnya.

Berdasarkan pengujian-pengujian di atas dapat diketahui bahwa pengujian sistem navigasi robot penerima tamu dapat dinyatakan berhasil dengan parameter mampu mendekati objek dengan trajektori yang telah dibuat.

BAB 6

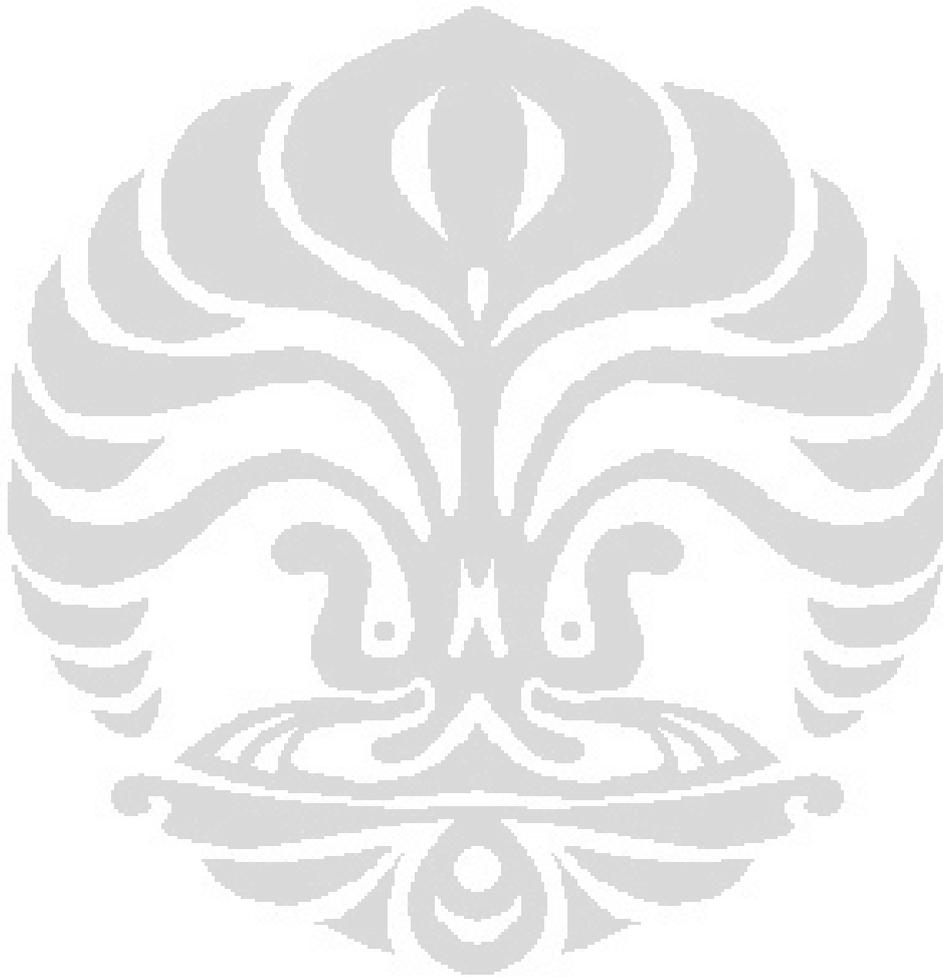
KESIMPULAN

Berdasarkan pengujian yang dilakukan terhadap sistem navigasi robot penerima tamu dengan $K_p=2350$ dan $K_d=1300$, maka dapat disimpulkan bahwa :

1. Sistem navigasi robot penerima tamu melalui trajektori dapat dikatakan berhasil bernavigasi untuk menuju suatu objek pada koordinat tertentu, namun masih terdapat eror yang terjadi yang diakibatkan oleh :
 - a. Adanya perbedaan kecepatan motor kiri dan motor kanan robot yang menyebabkan pergerakan robot tidak halus
 - b. Adanya slip pada roda saat bernavigasi yang menyebabkan terjadinya eror pada posisi robot
 - c. Konstanta pengendali sudut yang masih belum sesuai sehingga mengakibatkan eror pada perputaran robot
2. Robot mampu mendekati objek tanpa terjadinya tabrakan dengan objek yang dituju dan objek lain yang melintas di depan robot dengan memanfaatkan data sensor jarak SRF-08.
3. Untuk memperkecil eror yang terjadi pada navigasi robot penerima tamu diperlukan pengendalian tambahan, yaitu pengendalian kecepatan motor.

DAFTAR ACUAN

- [1] Pitowarno, Epit. 2006. *Robotika Desain, Kontrol Dan Kecerdasan Buatan*. Yogyakarta:Andi.
- [2] Nugroho, Rizky Prasetya Ade. 2011. *Strategi Lokalisasi Mobile Robot Dengan Menggunakan Extended Kalman Filter Pada Lingkungan Terstruktur*. Universitas Indonesia



DAFTAR REFERENSI

- [1] Braunl, Thomas. 2006. *Embedded Robotics 2nd Edition*. Perth : Springer.
- [2] Cang, Ya-Chun, et. al. "Sensor-based Trajectory Planning Strategy for Non-holonomic Mobile Robot with Laser Range Sensors", *IEEE International Symposium on Industrial Electronics (ISIE 2009) Seoul Olympic Parktel*, Seoul, Korea July 5-8, 2009, page 1755-1760.
- [3] Chapman, Stephen J. 2002. *Electric Machinery and Power System Fundamental 1st Editions*. New York : The McGraw-Hill Companies, Inc.
- [4] Cook, David. 2009. *Robot Building for Beginners 2nd Edition*. New York : [http://www. Après.com](http://www.Après.com).
- [5] Dewanto, Vektor. 2009. *Kalibrasi Odometri Pada Tracked Mobile Robot Al-Fathyrss*. Fakultas Teknik Universitas Indonesia
- [6] J.L. Martinez, A. Mandow, et. al. "Kinematics Modelling of Tracked Vehicles by Experimental Identification", *The 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, 2004, Page. 1487-1492.
- [7] Laumond, Jean-Paul, et. al. "A Motion Planner for Nonholonomic Mobile Robots", *IEEE Transactions on Robotics and Automation Vol. 10 No. 5*, 1994, page 577-593.
- [8] Ljung, Lennart. 1994. *Modeling of Dynamic Sistem*. New Jersey : PTR Prentice Hall, Inc.
- [9] Mertinez-Alfaro, Horacio and Sergio Uribe-Gutierrez. "Designing, Making and Using a Mobile Robot", *IEEE 0-7803-4778-1/9 tk*, Monterrey, Mexico, 1998. Page 3307-3312.
- [10] Nugroho, Rizky Prasetya Ade. 2011. *Strategi Lokalisasi Mobile Robot Dengan Menggunakan Extended Kalman Filter Pada Lingkungan Terstruktur*. Universitas Indonesia
- [11] Pitowarno, Endro. 2006. *Robotika Desain, Kontrol Dan Kecerdasan Buatan*. Yogyakarta : Andi.
- [12] Tournassoud, P. and O. Jehl. "Motion Planning for a Mobile Robot with a Kinematic Constraint", *IEEE CH2555-1/88/0000/1785*, Lee Chesnay Cedex, France, 1988, Page 1785-1790.
- [13] <http://nobitakitoyo.blogspot.com/2010/12/robot-pelayan-sabar-dan-tak-doyan-tip.html>. Diakses pada tanggal 7 Juni 2012 Pukul 19.00 WIB