



**PERANCANGAN VISUALISASI POSISI ROBOT PADA PENGENDALI  
ROBOT SEPAK BOLA BERODA**

**TUGAS AKHIR**

**Program Studi**

**S1 Teknik Komputer**

**INSTITUT BISNIS  
DAN INFORMATIKA**

**stikom**  
SURABAYA

**Oleh:**

**VICKY INDRARTA**

**15410200071**

---

**FAKULTAS TEKNOLOGI DAN INFORMATIKA**

**INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA**

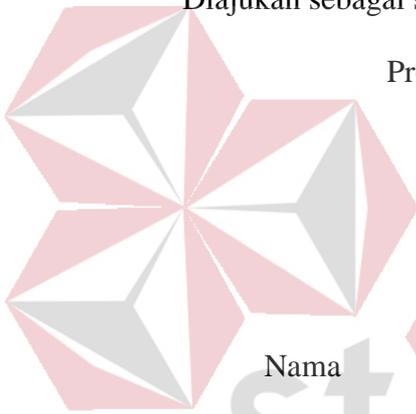
**2019**

**PERANCANGAN VISUALISASI POSISI ROBOT PADA PENGENDALI  
ROBOT SEPAK BOLA BERODA**

**TUGAS AKHIR**

Diajukan sebagai salah satu syarat untuk menyelesaikan

Program Sarjana Teknik



**Disusun Oleh:**

Nama : Vicky Indrarta

NIM : 15410200071

Program : S1 (Strata Satu)

Jurusan : Teknik Komputer

Fakultas : Teknologi Dan Informatika

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

**2019**



**“SETIAP MANUSIA HIDUP SELALU  
MEMILIKI TUJUAN DAN TUJUAN ITU  
HARUS DICAPAI”**

INSTITUT BISNIS  
INFORMATIKA

**stikom**  
SURABAYA

**Kupersembahkan Kepada**

**ALLAH SWT**

**Ayah, Ibu dan semua keluarga tercinta,**

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

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

## TUGAS AKHIR

### PERANCANGAN VISUALISASI POSISI ROBOT PADA PENGENDALI ROBOT SEPAK BOLA BERODA

Dipersiapkan dan disusun oleh

**VICKY INDRARTA**

**NIM : 15410200071**

Telah diperiksa, diuji dan disetujui oleh Dewan Pembahas

Pada : Agustus 2019

#### Susunan Dewan Pembimbing dan Penguji

##### Pembimbing

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

NIDN. 0727097302

II. Ira Puspasari, S.Si., M.T.

NIDN. 0710078601

##### Pembahas

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

NIDN. 0729047501


7/8/19

Tugas Akhir ini telah diterima sebagai salah satu persyaratan



Untuk memperoleh gelar Sarjana

FAKULTAS TEKNOLOGI  
DAN INFORMATIKA

stikom  
SURABAYA

Dr. Jusak

Dekan Fakultas Teknologi dan Informatika

9/8/19

FAKULTAS TEKNOLOGI DAN INFORMATIKA  
INSTITUT BISNIS DAN INFORMATIKA STIKOM SURABAYA

## SURAT PERNYATAAN

### PERSETUJUAN PUBLIKASI DAN KEASLIAN KARYA ILMIAH

Sebagai mahasiswa Institut Bisnis dan Informatika Stikom Surabaya, saya :

Nama : Vicky Indrarta  
NIM : 15410200071  
Program Studi : S1 Teknik Komputer  
Fakultas : Fakultas Teknologi dan Informatika  
Jenis Karya : Tugas Akhir  
Judul Karya : **Perancangan Visualisasi Posisi Robot Pada Pengendali Robot Sepak Bola Beroda**

Menyatakan dengan sesungguhnya bahwa:

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

Demikian surat pernyataan ini saya buat dengan sebenarnya.

Surabaya, Agustus 2019

Yang menyatakan  
  
Vicky Indrarta

Nim : 15410200071

## ABSTRAK

Perancangan visualisasi untuk mengontrol pergerakan robot sepak bola beroda adalah sebuah aplikasi yang digunakan untuk mendukung *base station* robot sepak bola beroda supaya memudahkan user untuk mengontrol robot. Aplikasi ini menggambarkan lapangan dan dua robot milik sendiri yang telah diskala. Selain digunakan untuk mengontrol robot, visualisasi ini bisa untuk memonitoring robot berada pada koordinatnya. Sistem ini sangat baik jika digunakan untuk membangun strategi pada saat pertandingan robot sepak bola beroda.

Strategi dalam perlombaan sangat penting perannya untuk membantu memenangkan lomba. Pada saat pertandingan robot harus berada diposisi awal yakni harus berada dibelakang garis gawang. Untuk pengontrolan robot diperlukan *base station* agar memudahkan user dalam pengontrolan robot sepak bola beroda. Visualisasi adalah nama aplikasi untuk memperkuat *base station*, karena visualisasi akan menampilkan tidak jauh beda antara kondisi lapangan sebenarnya dengan sistem yang telah dirancang.

Pada tugas akhir ini penulis mendapatkan hasil dari dua pengujian yaitu uji komunikasi antara robot dengan visualisasi dan uji ketepatan koordinat robot sepak bola beroda dengan visualisasi posisi robot. Kedua pengujian tersebut mendapatkan hasil yang baik yaitu tidak ada hasil error dalam pengujian.

**Kata kunci :** visualisasi, robot, koordinat.

## KATA PENGANTAR

Segala puji syukur kehadirat Allah SWT, karena dengan rahmatnya dan hidayah-Nya penulis dapat menyelesaikan penyusunan Laporan Tugas Akhir yang berjudul “PERANCANGAN VISUALISASI POSISI ROBOT PADA PENGENDALI ROBOT SEPAK BOLA BERODA”. Laporan Tugas Akhir ini disusun dalam rangka penulisan laporan untuk memperoleh gelar Sarjana Teknik pada program studi S1 Teknik Komputer Stikom Surabaya.

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

1. Orang Tua dan Saudara-saudara saya tercinta yang telah memberikan dorongan dan bantuan baik moral maupun materi sehingga penulis dapat menempuh dan menyelesaikan Tugas Akhir ini.
2. Bapak Dr. Jusak selaku Dekan Fakultas Teknologi dan Informatika (FTI) Institut Bisnis dan Informatika Stikom Surabaya telah membantu proses penyelesaian Tugas Akhir yang dibuat oleh penulis dengan Baik.
3. Bapak Pauladie Susanto, S.Kom., M.T., selaku Ketua Program Studi S1 Sistem Komputer Stikom Surabaya, dan selaku Dosen Penguji atas ijin dan masukkan dalam menyusun Tugas Akhir ini.
4. Bapak Dr.Susijanto Tri Rasmana, S.Kom., M.T. dan Ibu Ira Puspasari, S.Si., M.T. selaku Dosen Pembimbing yang selalu memberi arahan dan bimbingan dalam menyelesaikan Tugas Akhir beserta laporan ini.

5. Seluruh dosen pengajar Porgram Studi S1 Teknik Komputer yang telah mendidik, memberi motivasi kepada penulis selama masa kuliah di Institut Bisnis dan Informatika Stikom Surabaya.
6. Rekan-rekan Komunitas Stikom Robotik yang memberikan motivasi serta bantuan dalam penyelesaian Tugas Akhir ini.
7. Teman-teman seperjuangan Teknik Komputer angkatan 2015 dan semua pihak yang terlibat namun tidak dapat penulis sebutkan satu persatu atas bantuan dan dukungannya.
8. Serta semua pihak lain yang tidak dapat disebutkan secara satu per satu, yang telah membantu dalam menyelesaikan Tugas Akhir ini baik secara langsung maupun tidak langsung,

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

Surabaya, Juli 2019

Penulis

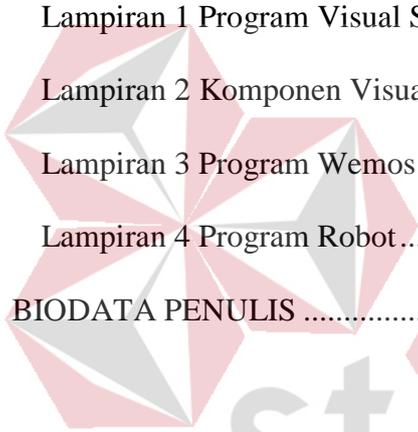
## DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
HALAMAN SYARAT .....	ii
MOTTO.....	iii
HALAMAN PERSEMBAHAN.....	iv
HALAMAN PENGESAHAN.....	v
ABSTRAK.....	vii
KATA PENGANTAR .....	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR .....	xiv
DAFTAR TABEL.....	xvi
DAFTAR LAMPIRAN.....	xviii
BAB I.....	1
PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 RUMUSAN MASALAH .....	3
1.3 BATASAN MASALAH .....	3
1.4 TUJUAN.....	3
1.5 Sistematika Penulisan.....	4
BAB II.....	6
LANDASAN TEORI.....	6
2.1 Robot Omnidirectional .....	6

2.2	Kinematika .....	7
2.3	Odometry .....	9
2.4	Kartesian .....	11
2.5	Visual Studio .....	13
2.6	Base Station .....	14
2.7	Perhitungan Skala Perbandingan Lapangan .....	15
2.8	TCP/IP .....	17
2.9	Arduino .....	18
2.9.1	Arduino Due .....	18
2.10	Wemos .....	20
2.10.1	Pin I/O Wemos .....	21
BAB III	.....	23
METODE PENELITIAN	.....	23
3.1	Metode Penelitian .....	23
3.2	Perancangan Visualisasi .....	26
3.2.1	Algoritma Visualisasi .....	26
3.2.2	Rancang Lapangan Pada Visualisasi .....	33
3.2.3	Rancang Robot Pada Visualisasi .....	35
3.2.4	Rancang Tools Pada Visualisasi .....	37
3.2.5	Rancang Pisah Data Visualisasi .....	39
3.2.6	Perancangan Kartesian Pada Visualisasi dan Robot Sepak Bola Beroda .....	45
3.3	Perancangan Komunikasi Robot .....	46
3.3.1	Terima Data Koordinat .....	46

3.3.2	Pemisah Data.....	48
3.4	Perancangan Komunikasi Data.....	51
BAB IV .....		53
PENGUJIAN.....		53
4.1	Uji Komunikasi Antara Robot Dengan Visualisasi.....	53
4.1.1	Alat Yang Digunakan Pada Uji Komunikasi Antara Robot Dengan Visualisasi.....	53
4.1.2	Prosedur Pengujian Pada Uji Komunikasi Antara Robot Dengan Visualisasi.....	54
4.1.3	Hasil Uji Komunikasi Antara Robot Dengan Visualisasi.....	55
4.1.4	Hasil Akhir Uji Komunikasi Antara Robot Dengan Visualisasi.....	80
4.1.5	Kesimpulan Dari Semua Hasil Uji Komunikasi Antara Robot Dengan Visualisasi.....	80
4.2	Uji Ketepatan Koordinat Robot Sepak Bola Beroda Dengan Visualisasi Posisi Robot.....	81
4.2.1	Alat Yang Digunakan Pada Uji Ketepatan Koordinat Robot Sepak Bola Beroda Dengan Visualisasi Posisi Robot.....	81
4.2.2	Prosedur Pengujian Pada Uji Ketepatan Koordinat Robot Sepak Bola Beroda Dengan Visualisasi Posisi Robot.....	81
4.2.3	Hasil Uji Ketepatan Koordinat Robot Sepak Bola Beroda Dengan Visualisasi Posisi Robot.....	83
4.2.4	Hasil Akhir Uji Ketepatan Koordinat Robot Sepak Bola Beroda Dengan Visualisasi Posisi Robot.....	89

4.2.5 Kesimpulan Dari Semua Hasil Uji Ketepatan Koordinat Robot Dengan Visualisasi .....	89
BAB V.....	90
PENUTUP.....	90
5.1 Kesimpulan.....	90
5.2 Saran.....	91
DAFTAR PUSTAKA .....	92
LAMPIRAN.....	94
Lampiran 1 Program Visual Studio.....	94
Lampiran 2 Komponen Visual Studio.....	101
Lampiran 3 Program Wemos Robot.....	115
Lampiran 4 Program Robot.....	117
BIODATA PENULIS .....	120



INSTITUT BISNIS  
DAN INFORMATIKA

stikom  
SURABAYA

## DAFTAR GAMBAR

	Halaman
Gambar 2. 1 Robot Omnidirectional.....	6
Gambar 2. 2 Konversi Arah Hadap.....	8
Gambar 2. 3 Pemrosesan Odometry Dari Input Rotary Yang Dijadikan Koordinat X, Y, $\Theta$ .....	10
Gambar 2. 4 Sudut Hadap Robot .....	10
Gambar 2. 5 Kartesius.....	12
Gambar 2. 6 Aplikasi Visual Studio .....	14
Gambar 2. 7 Aplikasi Base Station .....	15
Gambar 2. 8 Lapangan KRSBI Beroda.....	16
Gambar 2. 9 Topology Jaringan TCP/IP.....	17
Gambar 2. 10 Arduino Due.....	19
Gambar 2. 11 Wemos D1 mini .....	20
Gambar 3. 1 Rancangan Penelitian .....	23
Gambar 3. 2 Visualisasi .....	26
Gambar 3. 3 Flowchart Visualisasi .....	27
Gambar 3. 4 Tampilan Pilih Robot Pada Visualisasi.....	28
Gambar 3. 5 Tampilan Tombol Kirim Koordinat Pada Visualisasi.....	31
Gambar 3. 6 Tampilan Penampung Data Yang Dikirim Dari Robot.....	32
Gambar 3. 7 Lapangan Sebenarnya Robot Sepak Bola Beroda.....	34
Gambar 3. 8 Lapangan Robot Sepak Bola Beroda Pada Visualisasi .....	35
Gambar 3. 9 Robot Sebenarnya .....	36

Gambar 3. 10 Robot Pada Visualisasi.....	36
Gambar 3. 11 Tampilan Tools Pada Visualisasi .....	37
Gambar 3. 12 Proses Pisah Data Visualisasi.....	40
Gambar 3. 13 Kartesius Visualisasi Dan Robot Sepak Bola Beroda.....	45
Gambar 3. 14 Flowchart Terima Data Koordinat Robot.....	47
Gambar 3. 15 Pemisah Data X, Y, $\Theta$ .....	48
Gambar 3. 16 Flowchart Pemisah Data.....	49
Gambar 3. 17 format pemisah data .....	49
Gambar 3. 18 Blok Diagram Perancangan Komunikasi Data.....	51



## DAFTAR TABEL

	Halaman
Tabel 2. 1 Ukuran Lapangan KRSBI Beroda Dalam Satuan Meter .....	16
Tabel 2. 2 Spesifikasi Teknik.....	21
Tabel 2. 3 Fungsi Pin .....	22
Tabel 4. 1 Uji Komunikasi Antara Robot 1 Ke Visualisasi Dengan Gerak Robot Ke Samping.....	55
Tabel 4. 2 Uji Komunikasi Antara Robot 2 Ke Visualisasi Dengan Gerak Robot Ke Samping.....	59
Tabel 4. 3 Uji Komunikasi Antara Robot 1 Ke Visualisasi Dengan Gerak Robot Maju .....	62
Tabel 4. 4 Uji Komunikasi Antara Robot 2 Ke Visualisasi Dengan Gerak Robot Maju .....	67
Tabel 4. 5 Uji Komunikasi Antara Robot 1 Ke Visualisasi Dengan Gerak Robot Serong .....	70
Tabel 4. 6 Uji Komunikasi Antara Robot 2 Ke Visualisasi Dengan Gerak Robot Serong .....	77
Tabel 4. 7 Hasil Akhir Uji Komunikasi Antara Robot Dengan Visualisasi .....	80
Tabel 4. 8 Uji Ketepatan Gerak Robot 1 Ke Samping Kanan Pada Visualisasi ...	83
Tabel 4. 9 Uji Ketepatan Gerak Robot 2 Ke Samping Kanan Pada Visualisasi ...	84
Tabel 4. 10 Uji Ketepatan Gerak Robot 1 Maju Pada Visualisasi Posisi Robot...	85
Tabel 4. 11 Uji Ketepatan Gerak Robot 2 Maju Pada Visualisasi.....	86
Tabel 4. 12 Uji Ketepatan Gerak Robot 1 Serong Pada Visualisasi .....	87

Tabel 4. 13 Uji Ketepatan Gerak Robot 2 Serong Pada Visualisasi ..... 88

Tabel 4. 14 Hasil Uji Ketepatan Koordinat Robot Sepak Bola Beroda Dengan  
Visualisasi ..... 89



## DAFTAR LAMPIRAN

	Halaman
Lampiran 1 Program Visual Studio.....	94
Lampiran 2 Komponen Visual Studio.....	101
Lampiran 3 Program Wemos Robot .....	115
Lampiran 4 Program Robot.....	117



# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Kontes Robot Sepak Bola Indonesia Beroda diadakan untuk meningkatkan keilmuan mahasiswa dalam bidang robotika. Dalam kontes ini mahasiswa dituntut untuk mengembangkan kemampuan pada bidang mekanika, manufaktur, elektronika, pemrograman, artificial intellegent, image processing, komunikasi digital, strategi, kemampuan meneliti dan menulis artikel, sekaligus diperlukan pengembangan ke arah disiplin, toleransi, sportifitas, kerjasama, saling menghargai, kontrol emosi dan kemampuan *softskill* lainnya (Kementrian Riset, 2017).

Pada saat pertandingan robot harus di posisi awal yakni harus ada di belakang garis gawang. Agar tidak perlu memindahkan robot dengan cara manual atau mengangkat diperlukan pengontrolan terhadap robot agar pada saat sebelum pertandingan dimulai tidak perlu mengangkat robot dan menaruhnya pada posisi awal (Amin, 2019). *Reeferee box* adalah teknologi yang digunakan untuk mengatur jalannya pertandingan. Fungsi *reeferee box* yakni membantu wasit dalam menyampaikan keputusan wasit kepada robot yang sedang ditengah-tengah pertandingan (Ardhiansyah & Syarifuddin, 2017). Supaya *reeferee box* dapat terhubung ke robot maka perlu adanya *base station* sebagai penghubung antara *reeferee box* dengan robot. Fungsi *base station* untuk mengontrol pergerakan robot secara otomatis dilapangan tanpa harus mengangkat. Belum ada pengontrolan

dalam *base station* yang menentukan pergerakan robot secara detail sesuai dengan koordinat dalam skala lapangan yang sebenarnya.

Untuk mengatasi beberapa permasalahan di atas maka penulis ingin mengembangkan pada robot sepak bola beroda dengan cara membuat suatu visualisasi posisi robot yang dirancang untuk mengirim dan menerima informasi sesuai keadaan yang sebenarnya. Dimana visualisasi ini bekerja dibawah kendali operator dalam menentukan segala pergerakan robot. Tidak hanya untuk mengendalikan robot, tetapi juga dapat memonitor pergerakan robot yang sedang aktif dengan mengambil data umpan balik dari proses odometry. Visualisasi ini menerima data paket pulsa yang dikirim secara langsung dari robot berupa data pulsa putaran (rotary) kedua roda untuk kemudian diolah menjadi data koordinat yang digunakan untuk memvisualisasikan pergerakan robot pada bidang kartesian (Wahyuni, 2015).

Sistem kendali yang dilakukan untuk menentukan titik koordinat yang dituju dapat dilakukan dengan cara memasukkan koordinat angka pada komputer sehingga pengguna tidak melakukan kendali pada robot secara manual dan satu persatu. data perhitungan titik koordinat yang dihasilkan mempunyai jarak antar titik yang cukup jauh sehingga perlu mencari titik-titik lain yang dapat digunakan sebagai pembantu untuk memperhalus transisi antar posisi. Interpolasi Visualisasi ini berfungsi sebagai alat pembelajaran mengkonversi data-data kecepatan Angular menjadi data pergerakan robot dalam bidang kartesian (Wahyuni, 2015).

## 1.2 RUMUSAN MASALAH

Adapun rumusan masalah sesuai yang telah diuraikan dalam latar belakang sebagai berikut:

1. Bagaimana merancang visualisasi agar sesuai keadaan di lapangan sepak bola robot.
2. Bagaimana merekam gerak robot sesuai lokasi yang sebenarnya dan menampilkan di visualisasi.

## 1.3 BATASAN MASALAH

Agar tidak menyimpang dari tujuan yang akan dicapai maka diberi batasan masalah sebagai berikut:

1. Hanya menggambarkan lapangan dan kondisi gerak robot.
2. Robot berjalan tanpa memperdulikan halangan yang ada.
3. Kondisi lapangan datar.
4. Menganggap pergerakan robot benar.
5. Tidak membahas jaringan komunikasi yang digunakan.

## 1.4 TUJUAN

Adapun tujuan dari Penelitian ini sebagai berikut:

1. Visualisasi dapat mensimulasikan sesuai keadaan di lapangan.
2. Mendapatkan data lokalisasi dari robot yang akan menampilkan pergerakannya.

## **1.5 Sistematika Penulisan**

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

### **BAB 1 PENDAHULUAN**

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

### **BAB II LANDASAN TEORI**

Pada bab ini membahas teori-teori yang berhubungan dan mendukung dalam pembuatan tugas akhir seperti robot, mikrokontroler, wemos, odometry, dan literatur yang menunjang dalam pembuatan tugas akhir ini.

### **BAB III METODE PENELITIAN**

Pada bab ini membahas tentang metode penelitian baik dalam perancangan sistem yang meliputi perangkat keras maupun perangkat lunak pada tugas akhir ini.

### **BAB IV PENGUJIAN**

Pada bab ini menjelaskan tentang hasil dari pengujian robot. Pengujian yang dilakukan yaitu uji komunikasi antara robot dengan visualisasi, dan uji ketepatan koordinat robot dengan koordinat gambar visual robot pada visualisasi.

## **BAB V PENUTUP**

Pada bab ini menjelaskan tentang kesimpulan dan saran. Kesimpulan akan dijelaskan berdasarkan dari hasil pengujian tugas akhir ini, serta saran-saran untuk perkembangan.



## BAB II

### LANDASAN TEORI

#### 2.1 Robot Omnidirectional

*Omnidirectional* merupakan salah satu konsep pergerakan robot yang dapat bergerak ke segala arah, dimana konsep pergerakan ini mempunyai kelebihan dibandingkan dengan konsep pergerakan robot lainnya.

Roda omni atau roda poli adalah roda dengan cakram kecil di sekitar lingkaran yang tegak lurus terhadap arah pengerolan. Efeknya adalah bahwa roda akan roll dengan kekuatan penuh, tetapi juga akan geser lateral dengan sangat mudah. Roda ini sering digunakan dalam sistem penggerak Holonomic. Terdapat contoh robot omnidirectional yang ditunjukkan pada gambar 2.1.



Gambar 2. 1 Robot *Omnidirectional*

## 2.2 Kinematika

Dalam melakukan kontrol pergerakan robot, penggunaan sistem kinematika menurut teori yang didapat adalah menggunakan *inverse* kinematika dengan mengacu pada persamaan 2.1, berikut adalah penjabaran persamaan berdasarkan teori yang didapat dengan implementasi rumus pada kontrol pergerakan robot *holonomic* beroda 3:

$$\begin{pmatrix} Vm_1 \\ Vm_2 \\ Vm_3 \end{pmatrix} = \begin{pmatrix} -\sin\theta_1 & \cos\theta_1 & 1 \\ -\sin\theta_2 & \cos\theta_2 & 1 \\ -\sin\theta_3 & \cos\theta_3 & 1 \end{pmatrix} \begin{pmatrix} V_x \\ V_y \\ R\omega \end{pmatrix} \quad (2.1)$$

Hasil perkalian matriks dari persamaan 2.1 adalah sebagai berikut:

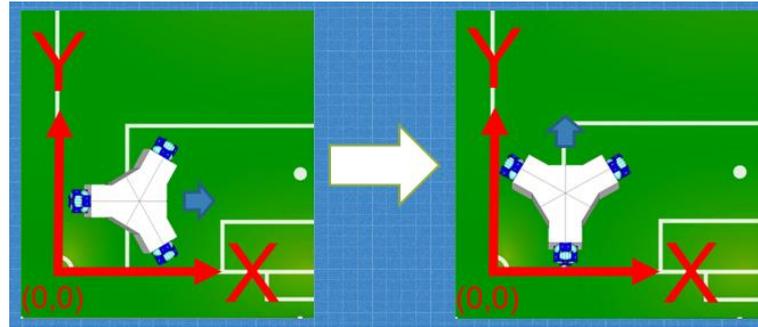
$$\begin{pmatrix} Vm_1 \\ Vm_2 \\ Vm_3 \end{pmatrix} = \begin{pmatrix} -V_x \sin\theta_1 + V_y \cos\theta_1 + R\omega \\ -V_x \sin\theta_2 + V_y \cos\theta_2 + R\omega \\ -V_x \sin\theta_3 + V_y \cos\theta_3 + R\omega \end{pmatrix} \quad (2.2)$$

$$Vm_1 = -V_x \sin\theta_1 + V_y \cos\theta_1 + R\omega \quad (2.3)$$

$$Vm_2 = -V_x \sin\theta_2 + V_y \cos\theta_2 + R\omega \quad (2.4)$$

$$Vm_3 = -V_x \sin\theta_3 + V_y \cos\theta_3 + R\omega \quad (2.5)$$

Berdasarkan desain mekanisme yang telah dibuat dan ketentuan persepsi arah hadap robot yang menghadap koordinat (Y) pada pembahasan bab sebelumnya, maka perlu dilakukan konversi arah hadap agar sesuai dengan ketepatannya dikarenakan pada rumus *inverse* kinematika mempunyai standard normal arah hadap robot yang menghadap koordinat (X) (Abseno, 2019).



Gambar 2. 2 Konversi Arah Hadap

Pada gambar 2.2 adalah analogi dari konversi arah hadap. Cara melakukan konversi arah hadap cukup dengan mengurangi sudut pemasangan seluruh roda dengan  $90^\circ$ .

Berikut adalah penulisan persamaan rumus untuk masing-masing kecepatan motor sesuai bentuk basis penempatan roda yang telah dirancang:

$$Vm_1 = -V_x \sin(\theta_1 - 90^\circ) + V_y \cos(\theta_1 - 90^\circ) + R\omega \quad (2.6)$$

$$Vm_2 = -V_x \sin(\theta_2 - 90^\circ) + V_y \cos(\theta_2 - 90^\circ) + R\omega \quad (2.7)$$

$$Vm_3 = -V_x \sin(\theta_3 - 90^\circ) + V_y \cos(\theta_3 - 90^\circ) + R\omega \quad (2.8)$$

Keterangan:

$V_m$  = Kecepatan motor (rpm)

$V_y$  = Kecepatan robot vertikal (m/s)

$V_x$  = Kecepatan robot horizontal (m/s)

$R\omega$  = Kecepatan sudut robot (rad/s)

$\theta$  = Sudut pemasangan roda ( $\theta_1=60^\circ$ ,  $\theta_2=180^\circ$ ,  $\theta_3=300^\circ$ )

Dengan adanya persamaan 2.6, 2.7, 2.8 maka telah diketahui proporsi masing masing kecepatan motor yang telah ditentukan oleh persamaan tersebut sesuai dengan besar/kecil *input* yang diberikan terhadap kecepatan X ( $V_x$ ),

kecepatan Y ( $V_y$ ), dan kecepatan sudut ( $R\omega$ ). Rumus kinematika yang digunakan sangatlah fleksibel, rumus ini dapat disesuaikan dengan sudut pada tata letak pemasangan roda dan banyaknya roda yang digunakan, sehingga cukup mudah dalam memodifikasi tata letak roda sesuai kebutuhan pengguna. Untuk menentukan kecepatan robot pada Tugas Akhir ini maka diperlukan persamaan pembalik dari persamaan 2.6, 2.7, 2.8, sehingga dapat dituliskan menjadi persamaan 2.9, 2.10, 2.11 sebagai berikut:

$$V_x = Vm_2 - Vm_1 \cos 60 - Vm_3 \cos 300 \quad (2.9)$$

$$V_y = Vm_1 \sin 60 - Vm_3 \sin 300 \quad (2.10)$$

$$V_\theta = \frac{Vm_1 + Vm_2 + Vm_3}{L} \quad (2.11)$$

Keterangan:

$V_x$ =Kecepatan robot horizontal (m/s)

$V_y$ =Kecepatan robot vertikal (m/s)

$V_\theta$ =Kecepatan sudut robot (rad/s)

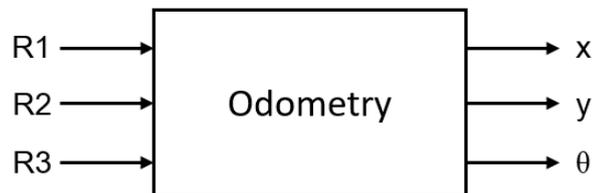
$Vm$  =Kecepatan motor (rpm)

$L$  =Jarak roda dengan titik pusat robot (cm)

### 2.3 Odometry

Metode untuk melakukan pengukuran dan perekaman putaran *rotary* yang digunakan sebagai penentuan suatu posisi adalah menggunakan metode umpan balik *odometry*. Seperti pada gambar 2.3 *odometry* bekerja dengan cara memasukkan parameter berupa pulse dari ketiga *rotary encoder* sehingga dapat

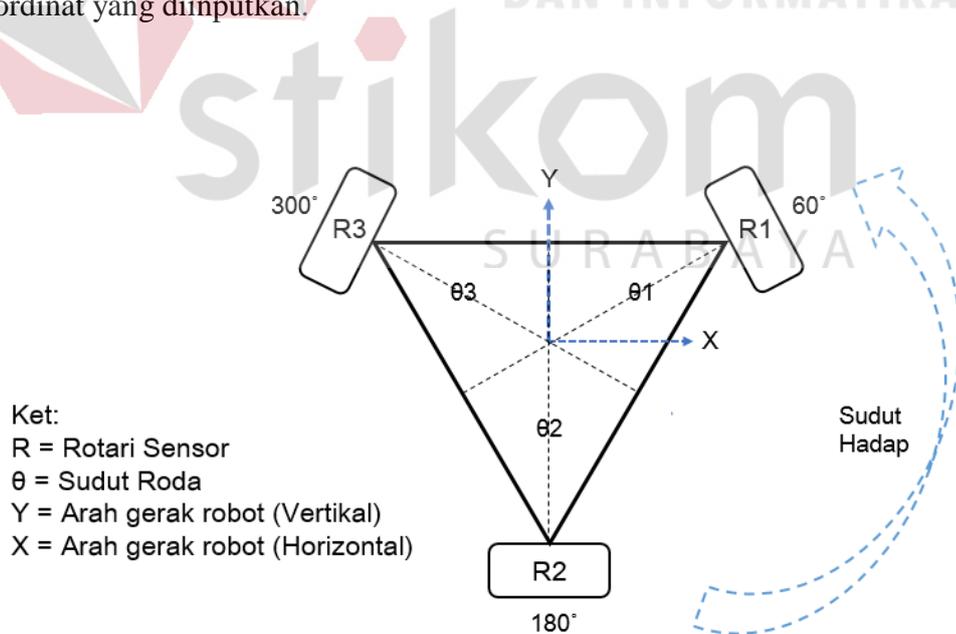
menghasilkan nilai keluaran berupa  $x$ ,  $y$ ,  $\theta$  untuk diolah sebagai koreksi umpan balik pergerakan. Pada gambar 2.4 ditentukan sudut dan posisi masing-masing rotary sebagai acuan dalam menentukan persamaan rumus odometry.



Gambar 2. 3 Pemrosesan *Odometry* Dari Input Rotary Yang Dijadikan Koordinat

$X, Y, \theta$

Gambar 2.3 merupakan hasil dari R1, R2, R3 yang diolah menggunakan odometry agar dapat menjadi koordinat  $X, Y, \theta$  supaya robot dapat berjalan dengan koordinat yang diinputkan.



Gambar 2. 4 Sudut Hadap Robot

Gambar 2.4 merupakan sudut hadap robot yang terdiri dari R1, R2, dan R3.

*Odometry* adalah metode yang digunakan sebagai sistem umpan balik dari pembacaan rotary encoder untuk merekam seluruh pergerakan robot sehingga dapat menghasilkan titik koordinat dan arah hadap robot.

$$X = 2/3 R_2 - 1/3 R_1 - 1/3 R_3 \dots\dots\dots(2.12)$$

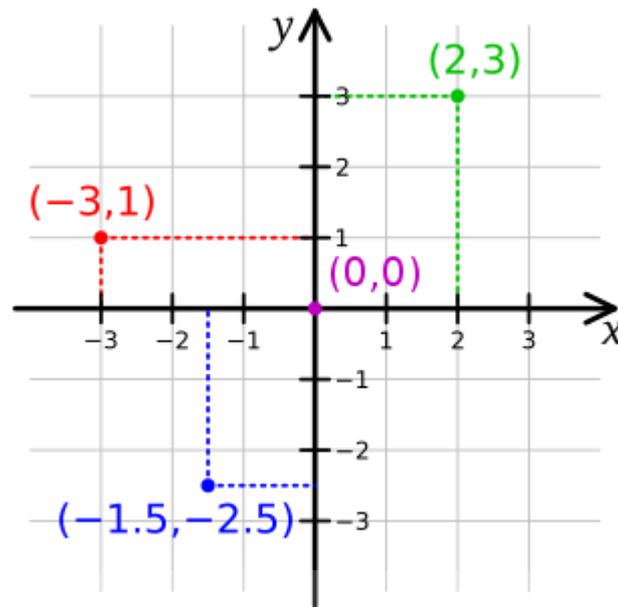
$$Y = 1/\sqrt{3} R_3 - 1/\sqrt{3} R_1 \dots\dots\dots(2.13)$$

$$\text{Sudut Hadap} = \frac{R_1+R_2+R_3}{3} \dots\dots\dots(2.14)$$

Dengan Persamaan diatas dapat diperoleh rumusan pembacaan titik koordinat yang didapat melalui kombinasi perhitungan 3 rotary encoder (Jiraphan dan Chirdpong,2014: 3).

## 2.4 Kartesius

Koordinat kartesius digunakan untuk menentukan tiap titik dalam bidang dengan menggunakan dua bilangan yang biasa disebut koordinat x dan koordinat y dari titik tersebut. Untuk mendefinisikan koordinat diperlukan dua garis berarah lurus satu sama lain antara sumbu x dengan sumbu y. Diagram kartesius dapat dilihat pada gambar 2.5.



Gambar 2. 5 Kartesius

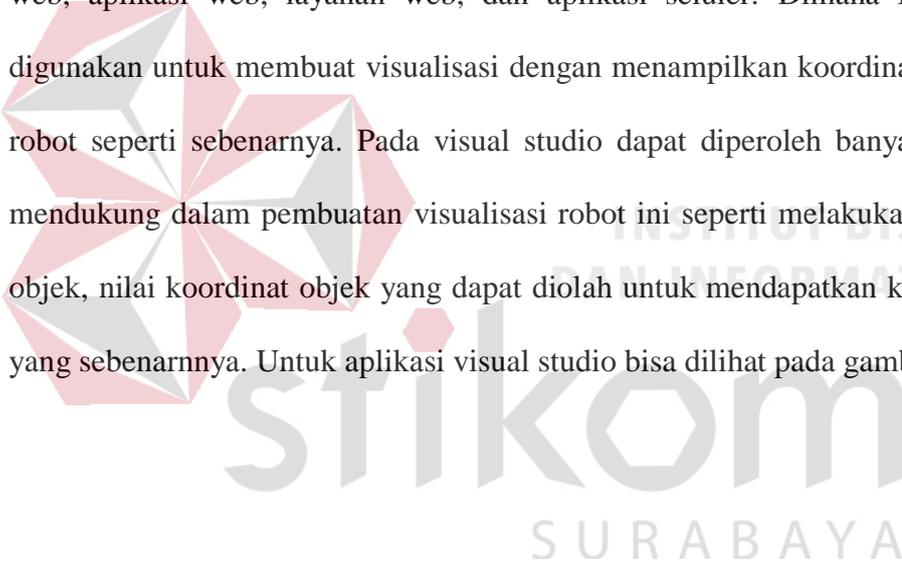
Dari gambar 2.5 diatas dapat dilihat bahwa ada 4 titik yang telah ditandai diantaranya:  $(-3,1)$ ,  $(2,3)$ ,  $(-1.5, -2.5)$ ,  $(0,0)$ . Untuk titik  $(0,0)$  disebut titik asal.

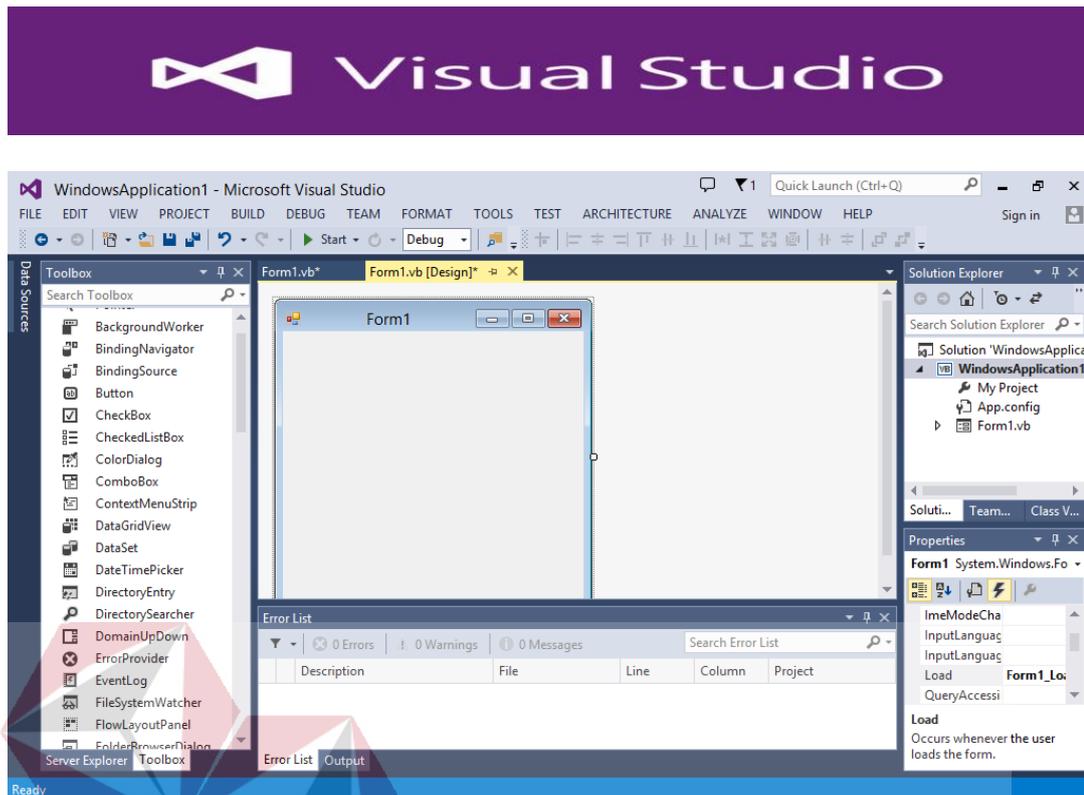
Dari gambar 2.4 diatas juga dapat kita lihat bahwa: karena kedua sumbu bertegak lurus satu sama lain, bidang  $xy$  terbagi menjadi empat bagian yang disebut kuadran, yang pada Gambar di atas ditandai dengan titik  $(-3,1)$ , titik  $(2,3)$ , titik  $(-1.5,-2.5)$ . Menurut konvensi yang berlaku, keempat kuadran diurutkan mulai dari yang kanan atas (kuadran I), melingkar melawan arah jarum jam. Pada kuadran I, kedua koordinat ( $x$  dan  $y$ ) bernilai positif. Pada kuadran II, koordinat  $x$  bernilai negatif dan koordinat  $y$  bernilai positif. Pada kuadran III, kedua koordinat bernilai negatif, dan pada kuadran IV, koordinat  $x$  bernilai positif dan  $y$  negatif . Titik  $(2,3)$  terletak pada kuadran I, titik  $(-3,1)$  terletak pada kuadran II dan titik  $(-1.5,-2.5)$  terletak pada kuadran III.

Dalam kehidupan sehari-hari dijumpai gambar denah atau peta. Peta memudahkan kita untuk dalam mencari tempat atau wilayah. Maka kartesius dalam pemetaan gambar lapangan robot sangat diperlukan agar robot mudah mencapai koordinat yang dituju.

## 2.5 Visual Studio

Microsoft Visual Studio adalah pengembangan terintegrasi (IDE) dari Microsoft. Ini digunakan untuk mengembangkan program komputer, serta situs web, aplikasi web, layanan web, dan aplikasi seluler. Dimana IDE ini bisa digunakan untuk membuat visualisasi dengan menampilkan koordinat pergerakan robot seperti sebenarnya. Pada visual studio dapat diperoleh banyak data yang mendukung dalam pembuatan visualisasi robot ini seperti melakukan pergerakan objek, nilai koordinat objek yang dapat diolah untuk mendapatkan kordinat robot yang sebenarnya. Untuk aplikasi visual studio bisa dilihat pada gambar 2.6.





Gambar 2. 6 Aplikasi Visual Studio

## 2.6 Base Station

*Base station* adalah komputer, notebook, atau laptop yang menjalankan software yang digunakan untuk memantau dan mengatur kerja robot secara otomatis berdasarkan instruksi yang diterima dari *Referee Box* (Ardhiansyah & Syarifuddin, 2017). *Base station* melakukan koordinasi aktivitas setiap robotnya dalam pertandingan. Selama pertandingan sedang berjalan, anggota tim peserta tidak diizinkan untuk mengoperasikan *Base station*-nya. *Base station* menggunakan kabel Ethernet Switch untuk melakukan koneksi dengan Referee Box. *Base station* memanggil alamat IP (IP Address) *Referee Box* untuk melakukan koneksi melalui Port 28097. Alamat IP Komputer yang menjalankan *Referee Box* akan diumumkan pada saat pertandingan. Cara koneksi antara *Base station* dengan robot yakni setiap

robot memiliki alamat IP (IP address). *Base station* dapat terhubung dengan robot secara wireless melalui access point yang khusus disediakan panitia dalam pertandingan (KRSBI-B, 2017). Untuk *base station* tim robotika SYNA-SWC8 dapat dilihat pada gambar 2.7.

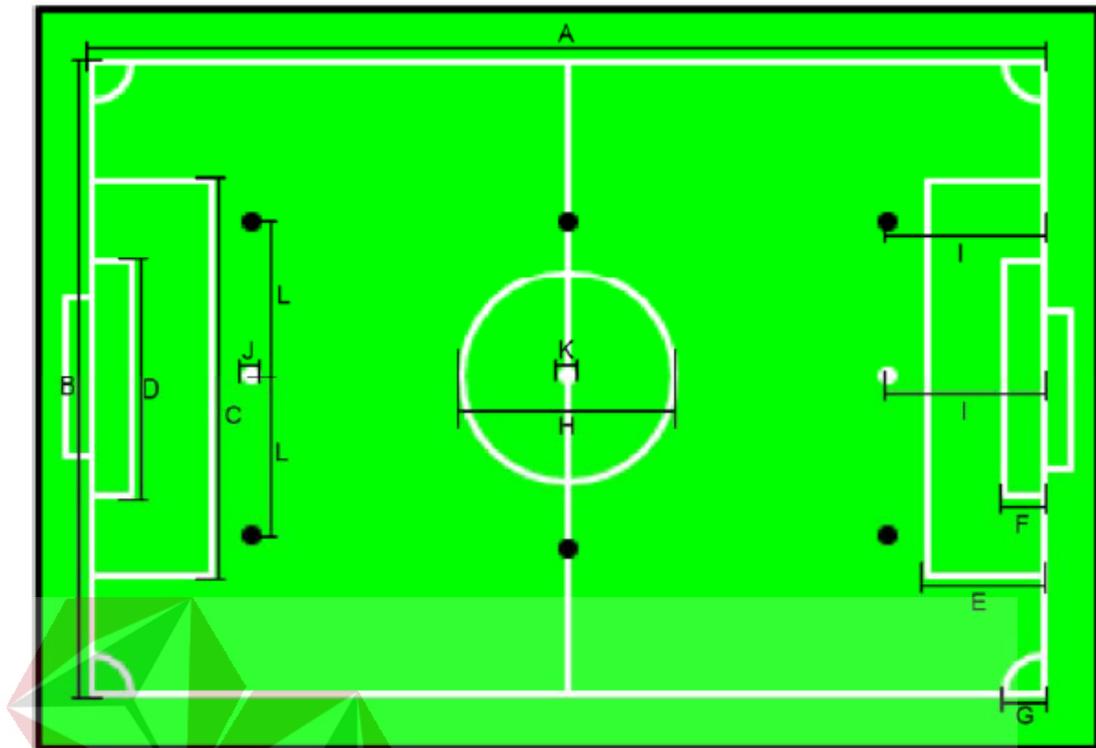


Gambar 2. 7 Aplikasi *Base Station*

## 2.7 Perhitungan Skala Perbandingan Lapangan

Skala pada gambar 2.8 adalah perbandingan paling sederhana antara jarak pada gambar dengan jarak pada objek sebenarnya yang menggunakan satuan jarak pada gambar. Contohnya jika diketahui jarak antara dua titik pada gambar 9 cm sedangkan jarak sebenarnya adalah 9 m. Jika kedua jarak tersebut menggunakan satuan yang jarak pada gambar yaitu cm, maka jarak pada gambar adalah 9 cm dan jarak sebenarnya adalah 900 cm. Perbandingan jarak pada gambar dengan jarak sebenarnya adalah 9:900, dan perbandingan paling sederhana adalah 1:100.

Agar pembuatan jarak pada gambar dapat sesuai dengan jarak sebenarnya terdapat acuan dari aturan kompetisi yang ada pada tabel 2.1.



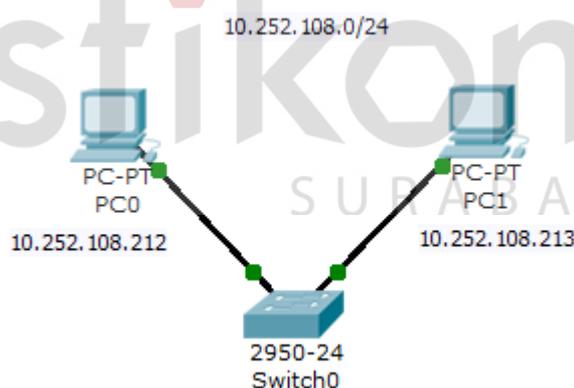
Gambar 2. 8 Lapangan KRSBI Beroda

Tabel 2. 1 Ukuran Lapangan KRSBI Beroda Dalam Satuan Meter

A	9	G	0,375
B	6	H	2
C	Lebar gawang 2,25	I	1,5
D	Lebar gawang 0,75	J	0,05
E	1,125	K	0,075
F	0,375	L	1,5

## 2.8 TCP/IP

TCP/IP merupakan singkatan dari *Transmission Control Protocol/Internet Protocol*. TCP/IP adalah salah satu perangkat lunak jaringan komputer (networking software) yang terdapat dalam sistem, dan dipergunakan dalam komunikasi data dalam local area network (LAN) maupun internet. TCP/IP merupakan protokol yg telah diterapkan pada hampir semua perangkat keras dan sistem operasi. Protokol ini menggunakan skema pengalamatan yang sederhana biasa disebut alamat IP yang mengizinkan banyak komputer agar bisa saling berhubungan antara satu dengan lainnya. IP adalah deretan angka biner antar 32 bit sampai 128 bit yang dipakai untuk mengidentifikasi tiap komputer (Wikipedia, 2018). TCP/IP sendiri memiliki topologi yang digunakan untuk mengetahui tiap-tiap komputer agar bisa berkomunikasi satu sama lain. Untuk topologi bisa dilihat pada gambar 2.10.



Gambar 2. 9 *Topology Jaringan TCP/IP*

Gambar 2.10 menjelaskan bahwa dengan adanya topologi jaringan akan memudahkan user mengetahui komputer mana saja yang saling berkomunikasi.

## 2.9 Arduino

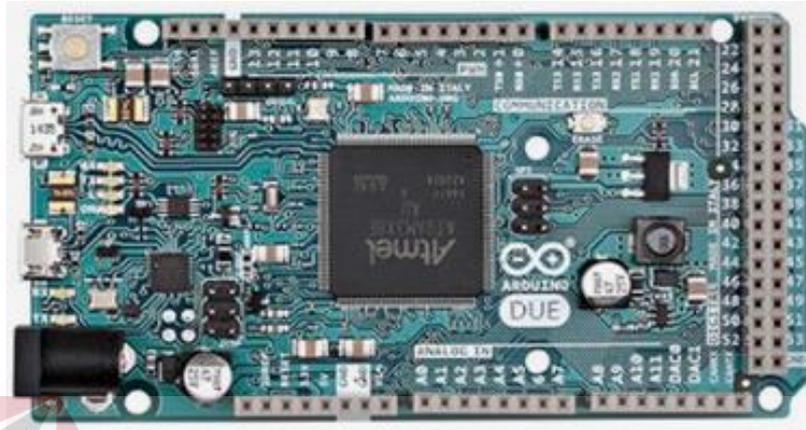
Arduino merupakan papan rangkaian elektronik dengan lisensi yang bersifat *open source*, dirancang dengan tujuan untuk memudahkan pengguna dalam penggunaan elektronika dalam berbagai bidang. Didalamnya terdapat komponen utama yaitu chip bernama AVR dari atmel, bahkan *software* arduino menggunakan bahasa sendiri yaitu bahasa C yang disederhanakan dan telah dibantu dengan *libraries* arduino yang telah ada. Saat ini arduino telah populer diseluruh dunia karena banyak yang menggunakan. Banyak pemula merasa senang belajar dengan arduino karena akses yang sangat mudah dipahami. Perlu diketahui bahwa bahasa pemrograman arduino sangat fleksibel dan tidak sulit maka dari itu banyak orang di dunia sangat menggemari arduino untuk merancang elektronika.

### 2.9.1 Arduino Due

Arduino Due yang dirilis tahun 2012 dengan CPU 32-bit ARM Cortex-M3 sama sekali berbeda dari Arduino Duemilanove dengan MCU 8-bit ATmega168 yang dirilis tahun 2009. Miskonsepsi yang sering terjadi karena "Due" disalah-artikan sebagai singkatan dari *Duemilanove*. Due adalah *advanced board* baru dengan kemampuan jauh di atas varian Arduino lainnya, Duemilanove adalah *entry-level board* lama yang merupakan "pendahulu" Arduino Uno.

Papan pengembang ini memiliki pin masukan/keluaran digital sebanyak 54 pin (12 di antaranya berkemampuan PWM), 12 pin masukan analog, 4 UART / hardware serial port, pencacah-waktu / clock berfrekuensi 84 MHz, koneksi dengan kemampuan USB OTG, 2 DAC (digital-to-analog converter), 2 TWI (Two Wire Interface, kompatibel dengan protokol I2C dari Phillips), soket jack catu daya

standar (5,5/2,1mm), konektor SPI header, konektor JTAG header, tombol reset, dan sebuah tombol hapus (erase button) (Indoware, 2019). Arduino due dapat dilihat pada gambar 2.10 berikut.



Gambar 2. 10 Arduino Due

Kelebihan utama Arduino Due sebagai berikut.

1. Core processor 32-bit yang memungkinkan operasi data sebanyak 4 byte sekaligus (tipe data DWORD) pada satu siklus waktu.
2. RAM statis / SRAM sebesar 96 KiloByte (48 kali lipat lebih besar dibanding Arduino Uno, 12 kali lipat lebih besar dibanding Arduino Mega 2560).
3. Ruang untuk kode program / Flash Memory sebesar 512 KB (16 kali lipat lebih besar dibanding Arduino Uno, 2 kali lipat lebih besar dibanding Arduino Mega 2560).
4. Terdapat pengendali akses memori langsung (Direct Memory Access / DMA controller) yang dapat membebaskan CPU dari operasi memori yang intensif.
5. Terdapat dua kanal DAC (Digital-Analog-Converter) terpadu (Arduino lainnya memiliki ADC tapi tidak memiliki DAC yang merupakan komplemen fungsi

dari ADC -- ADC mengubah sinyal analog menjadi digital, DAC mengubah sinyal digital menjadi sinyal analog yang sesungguhnya / true analog.

## 2.10 Wemos

Wemos bisa disebut dengan mikrokontroler yang dikembangkan berbasis ESP 8266 dimana wemos sendiri dapat diprogram layaknya arduino. Wemos diciptakan sebagai solusi dari sistem mikrokontroler yang berbasis *wireless tetapi* mampu didapatkan dengan harga terjangkau. Dengan menggunakan mikrokontroler wemos dapat meminimalisir biaya yang digunakan untuk membangun sistem mikrokontroler berbasis *wifi*. Wemos memiliki dua versi yaitu wemos D1 mini dan wemos D1. Pada Tugas Akhir ini penulis akan menggunakan wemos dengan versi D1 mini. Bentuk fisik dari wemos sendiri dapat dilihat pada gambar 2.11.



Gambar 2. 11 Wemos D1 mini

### 2.10.1 Pin I/O Wemos

Wemos sendiri memiliki spesifikasi teknis dan fungsi pin. Untuk spesifikasi teknis meliputi ukuran, berat, kapasitas, dan tegangan yang dibutuhkan. Untuk fungsi pin wemos memiliki 9 pin input / output digital, 2 pin serial Rx, Tx, 1 pin digunakan untuk analog input tegangan maksimal 3,2 volt, 2 pin input tegangan 3,3 dan 5 volt, 1 pin ground, 1 pin reset. Untuk lebih jelasnya dapat dilihat pada tabel 2.2 dan 2.3.

Tabel 2. 2 Spesifikasi Teknik

<b>Mikrokontroler</b>	<b>ESP 8266EX</b>
Operating Voltage	3.3V
Digital I/O Pins	11
Analog Input Pins	1(Max input: 3.2V)
Clock Speed	80MHz/160MHz
Flash	4M bytes
Length	34.2mm
Width	25.6mm
Weight	3g

Pada tabel 2.2 menjelaskan bahwa mengoperasikan wemos perlu daya 3,3 atau 5 volt agar wemos dapat beroperasi. Jika wemos mendapatkan tegangan kurang dari 3,3 volt maka wemos tidak akan bisa dioperasikan, dan sebaliknya jika wemos mendapat tegangan diatas 5 volt maka wemos akan terbakar atau rusak karena kelebihan tegangan.

Tabel 2. 3 Fungsi Pin

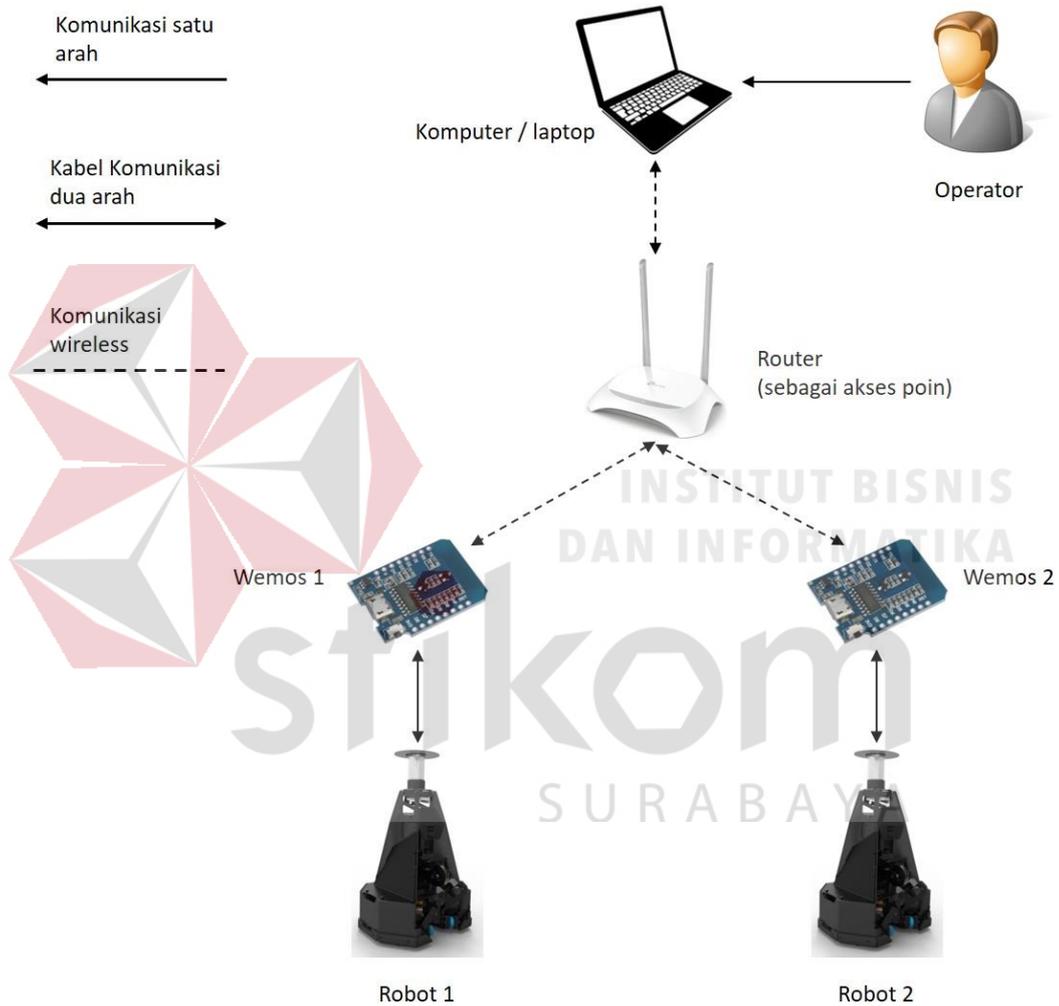
<b>Pin</b>	<b>Fungsi</b>	<b>ESP-8266 Pin</b>
Tx	TxD	TxD
Rx	RxD	RxD
A0	Analog input, max 3,3V Input	A0
D0	IO	GPIO16
D1	IO, SCL	GPIO5
D2	IO, SDA	GPIO4
D3	IO, 10k Pull-up	GPIO0
D4	IO, 10k Pull-up, BUILTIN_LED	GPIO2
D5	IO, SCK	GPIO14
D6	IO, MISO	GPIO12
D7	IO, MOSI	GPIO13
D8	IO, 10k Pull-down, SS	GPIO15
G	Ground	Gnd
5V	5V	-
3V3	3,3V	3,3V
RST	Reset	RST

Pada tabel 2.3 telah dijelaskan fungsi masing-masing pin pada wemos D1 mini. Telah diketahui juga bahwa wemos hampir mirip dengan arduino.

# BAB III

## METODE PENELITIAN

### 3.1 Metode Penelitian



Gambar 3. 1 Rancangan Penelitian

Metode Penelitian yang digunakan dalam pengerjaan Tugas Akhir ini adalah studi literatur, dan pembuatan program. Pada penelitian Tugas Akhir ini, peneliti telah membuat aplikasi visual dua dimensi yang memvisualisasikan

tampilan lapangan beserta gerak robot sepak bola beroda. Agar robot dapat berjalan secara *real-time* maka diperlukan koneksi menggunakan jaringan yang akan langsung terkoneksi ke robot sepak bola beroda. Blok diagram penelitian ditunjukkan pada gambar 3.1.

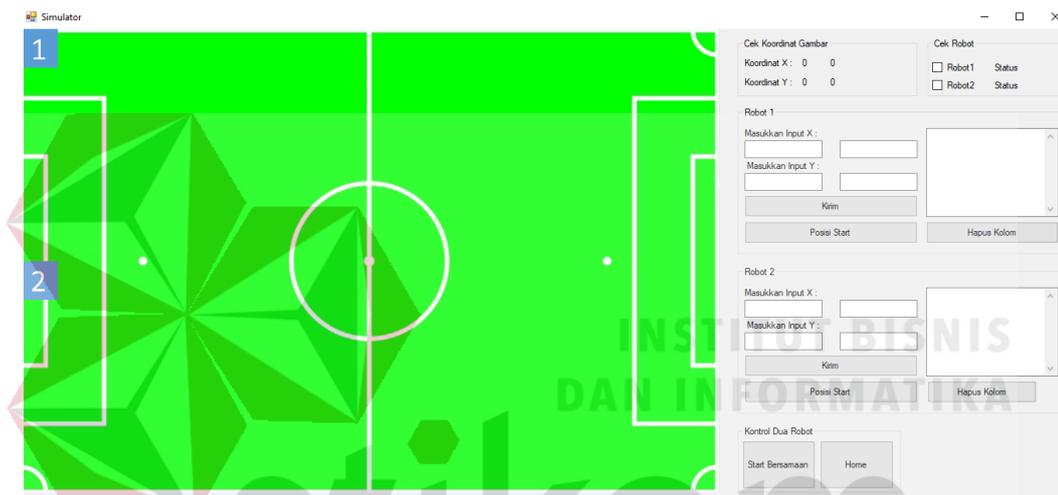
Penjelasan dari perancangan penelitian pada gambar 3.1 terdapat operator, komputer, router, wemos, dan robot. Dua robot yang digunakan dalam Tugas Akhir ini statusnya sebagai robot penyerang, karena yang aktif dalam pertandingan adalah robot penyerang maka dari itu perlu mengendalikannya. Karena terdapat dua robot maka diperlukan juga dua wemos dengan alamat *IP* yang berbeda supaya bisa digunakan untuk mengontrol kedua robot. Kedua robot memiliki posisi awal sendiri – sendiri yaitu untuk robot 1 pada koordinat x adalah 0, dan y adalah 0 sedangkan robot 2 berada pada koordinat x adalah 0, dan y adalah 300. Proses pengendalian robot 1 dan robot 2 sama, berikut akan dijelaskan tentang proses pengiriman data dari visualisasi menuju robot sampai bisa untuk monitoring gerak robot.

1. Pada proses awal operator akan melakukan koneksi ke *IP* wemos pada masing - masing robot. Pengkoneksian ini dilakukan agar data yang dikirim dapat sampai ke tujuan.
2. Setelah terkoneksi antara komputer dengan robot lalu robot akan mengirimkan nilai umpan balik yaitu dengan koordinat x bernilai 0 dan koordinat y juga bernilai 0.
3. Perlu peletakan robot secara bergantian pada sudut kiri atas lapangan yaitu robot 2 terlebih dahulu lalu robot 1. Peletakan robot 2 didahulukan karena posisi awal robot 2 berada pada koordinat  $x = 0$  dan  $y = 300$ .

4. Setelah berada pada posisi start awal masing – masing maka aplikasi siap dimasukkan input koordinat x dan y oleh operator.
5. Visualisasi akan mengolah data yang telah diinputkan oleh operator kemudian data tersebut akan dikirim ke wemos lalu dikirim ke operator. Data yang dimasukkan oleh operator berisi koordinat sumbu x dan sumbu y yang masih dalam bentuk pixel pada komputer.
6. Paket data yang dikirimkan oleh komputer pada router akan dilihat IP nya. Kemudian router akan mulai melakukan broadcast kepada IP yang terhubung. Router berfungsi membuat jalan paket data yang diterima dan mengirimkan ke IP tujuan.
7. Paket data yang dikirimkan akan diterima oleh wemos yang memiliki target IP sesuai. Wemos akan membaca isi dari paket data tersebut yang berisi koordinat sumbu x dan sumbu y. Wemos akan meneruskan kembali pengiriman paket data melalui serial Rx Tx pada mikrokontroler pada robot.
8. Paket data yang diterima melalui serial Rx Tx mikrokontroler pada robot akan dibaca dan diproses oleh mikrokontroler agar robot dapat bergerak sesuai dengan koordinat yang telah diinputkan pada visualisasi oleh operator.
9. Saat robot bergerak akan menghasilkan data dari perputaran motor dan data tersebut akan diproses oleh mikrokontroler kemudian dikirimkan lagi ke wemos melalui serial Rx Tx. Fungsi pengembalian data dari mikrokontroler ke visualisasi adalah agar dapat memonitoring atau membuat gambar visualisasi robot dua dimensi pada visualisasi dapat bergerak sesuai koordinat.

### 3.2 Perancangan Visualisasi

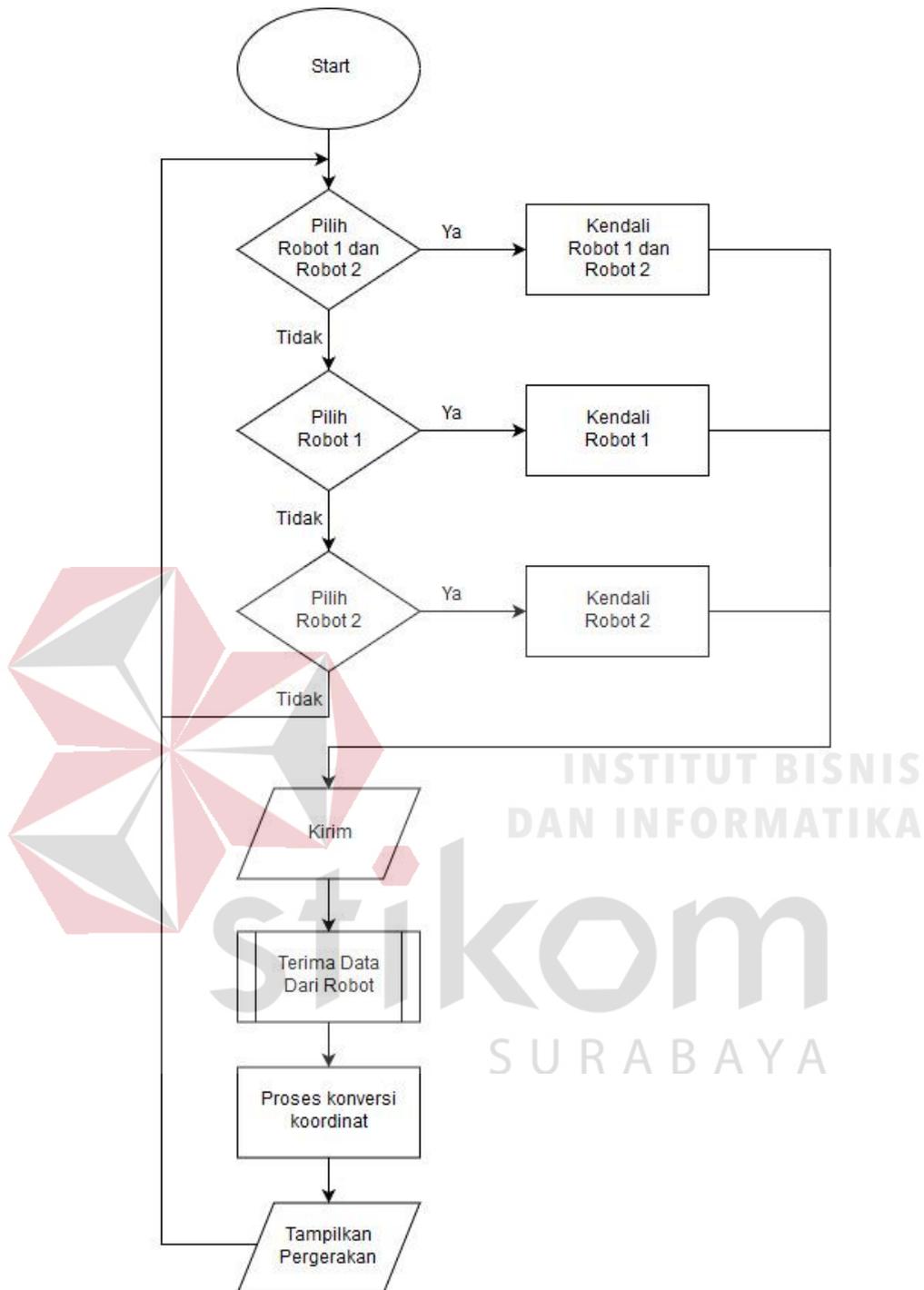
Perancangan visualisasi pada Tugas Akhir ini menggunakan aplikasi visual studio. Dalam perancangan visualisasi terdapat desain visualisasi 2 dimensi lapangan dan robot yang masing-masing memiliki ukuran atau dimensinya sendiri-sendiri. Dapat dilihat desain visualisasi secara keseluruhan pada gambar 3.2.



Gambar 3. 2 Visualisasi

#### 3.2.1 Algoritma Visualisasi

Untuk memudahkan user dapat menggunakan visualisasi ini alangkah lebih baiknya memahami *flowchart* dari perancangan visualisasi lebih dulu. Berikut *flowchart* yang digunakan untuk merancang visualisasi dapat dilihat pada gambar 3.3.



Gambar 3. 3 Flowchart Visualisasi

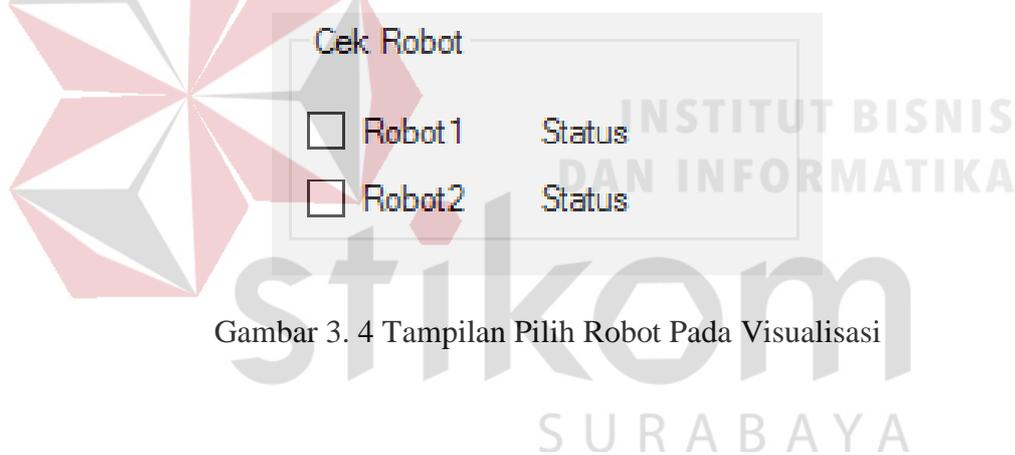
Pada gambar 3.3 menjelaskan bahwa langkah awal untuk mengoperasikan visualisasi yaitu user memilih mengkoneksikan pada robot 1, robot 2 atau keduanya juga bisa. Setelah terkoneksi antara visualisasi dengan robot lalu user akan menginputkan data koordinat x dan y dan mengirimkan data tersebut ke robot yang

telah teroneksi. Robot akan merespon dan melaksanakan perintah yang dikirim dari user tersebut dan mengirimkan kembali data koordinatnya berada ke visualisasi. Setelah terkirim maka visualisasi akan memproses data dari robot agar objek gambar robot pada visualisasi bisa bergerak sesuai kiriman koordinat robot saat bergerak.

Akan dijelaskan lebih detail lagi dengan informasi berikut ini.

#### 1. Pilih robot

Agar bisa terkoneksi antara visualisasi dengan robot maka perlu diberi IP *address* sendiri-sendiri dan mengkonfigurasi *gateway*. Berikut tampilan pada visualisasi. Untuk lebih jelasnya dapat melihat tampilannya pada gambar 3.4.



Gambar 3. 4 Tampilan Pilih Robot Pada Visualisasi

Berikut adalah potongan program koneksi robot 1.

```
private void checkBox1_CheckedChanged(object sender, EventArgs e)
{
    //Connect Robot 1
    try
    {
        if (checkBox1.Checked == true)
        {
            //Set IP dan Port Wemos
            sck1 = new Socket(AddressFamily.InterNetwork,
                SocketType.Stream, ProtocolType.Tcp);
            IP_sck1 = new
                IPEndPoint(IPAddress.Parse("192.168.0.150"), 10);
            //Connect Wemos

            sck1.Connect(IP_sck1);
            backgroundWorker1.RunWorkerAsync();
            backgroundWorker1.WorkerSupportsCancellation = false;
            timer1.Enabled = true;

            label17.Text = "Connected";
        }
        else
        {
            sck1.Close();
            //sck.Disconnect(true);
            label17.Text = "Disconnected";
            backgroundWorker1.WorkerSupportsCancellation = true;
            timer1.Enabled = false;
        }
    }
    catch
    {
        MessageBox.Show("Tidak Ada Koneksi Robot Penyerang 1");
    }
}
```

Berikut adalah potongan program koneksi robot 2.

```
private void checkBox2_CheckedChanged(object sender, EventArgs e)
{
    //Connect Robot 2
    try
    {
        if (checkBox2.Checked == true)
        {
            //Set IP dan Port Wemos
            sck2 = new Socket(AddressFamily.InterNetwork,
SocketType.Stream, ProtocolType.Tcp);
            IP_sck2 = new
IPEndPoint(IPAddress.Parse("192.168.0.150"), 10);
            //Connect Wemos

            sck2.Connect(IP_sck2);
            backgroundWorker2.RunWorkerAsync();
            backgroundWorker2.WorkerSupportsCancellation = false;
            timer2.Enabled = true;

            label15.Text = "Connected";
        }
        else
        {
            sck2.Close();
            //sck.Disconnect(true);
            label15.Text = "Disconnected";
            backgroundWorker2.WorkerSupportsCancellation = true;
            timer2.Enabled = false;
        }
    }
    catch
    {
        MessageBox.Show("Tidak Ada Koneksi Robot Penyerang 2");
    }
}
```

## 2. Koordinat Dari User

Untuk pengiriman data dibutuhkan koordinat x dan y agar robot dapat menyesuaikan input dari user. Untuk lebih jelasnya dapat melihat tampilannya pada gambar 3.5.

Gambar 3. 5 Tampilan Tombol Kirim Koordinat Pada Visualisasi

Berikut potongan program kirim robot 1.

```
private void button4_Click(object sender, EventArgs e)
{
    double angka1, angka2;
    angka1 = Convert.ToDouble(textBox1.Text);
    angka2 = Convert.ToDouble(textBox2.Text);
    //=====Kirim Split Data textBox=====
    string data = "<" + "-" + angka2 + ":" + angka1 + ":" + "0" + ">";
    byte[] kirim_data = Encoding.ASCII.GetBytes(data);
    sck1.Send(kirim_data);
}
```

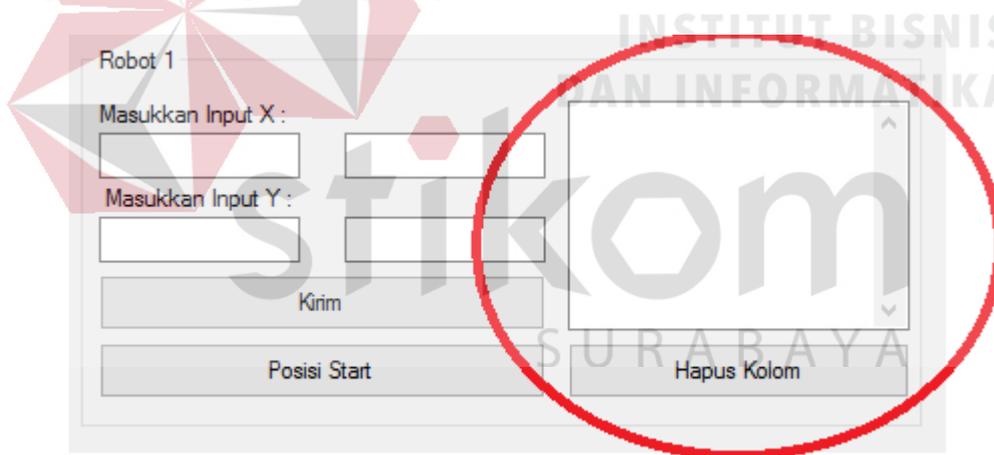
Berikut potongan program kirim robot 2.

```
private void button6_Click(object sender, EventArgs e)
{
    double angka1, angka2;
    angka1 = Convert.ToDouble(textBox9.Text);
    angka2 = Convert.ToDouble(textBox10.Text);

    //=====Kirim Split Data textBox=====
    string data2 = "<" + "-" + angka2 + ":" + angka1 + ":" + "0" + ">";
    byte[] kirim_data = Encoding.ASCII.GetBytes(data2);
    sck2.Send(kirim_data);
}
```

### 3. Terima Data Dari Robot

Robot yang terkoneksi akan terus mengirimkan data koordinatnya berada maka dari itu harus ada tempat untuk menampungnya sebelum diproses. Untuk lebih jelasnya dapat melihat tampilannya pada gambar 3.6.



Gambar 3. 6 Tampilan Penampung Data Yang Dikirim Dari Robot

Berikut potongan program terima data robot 1.

```
private void backgroundWorker1_DoWork(object sender, DoWorkEventArgs e)
{
    while (sck1.Connected)
    {
        try
        {
            byte[] terima_data = new byte[1024];
            int data_int = sck1.Receive(terima_data);
            String data = Encoding.ASCII.GetString(terima_data,0,
            data_int);});
```

Berikut potongan program terima data robot 2.

```
private void backgroundWorker2_DoWork(object sender, DoWorkEventArgs e)
{
    while (sck2.Connected)
    {
        try
        {
            byte[] terima_data = new byte[1024];
            int data_int = sck2.Receive(terima_data);
            String data = Encoding.ASCII.GetString(terima_data, 0,
            data_int);
```

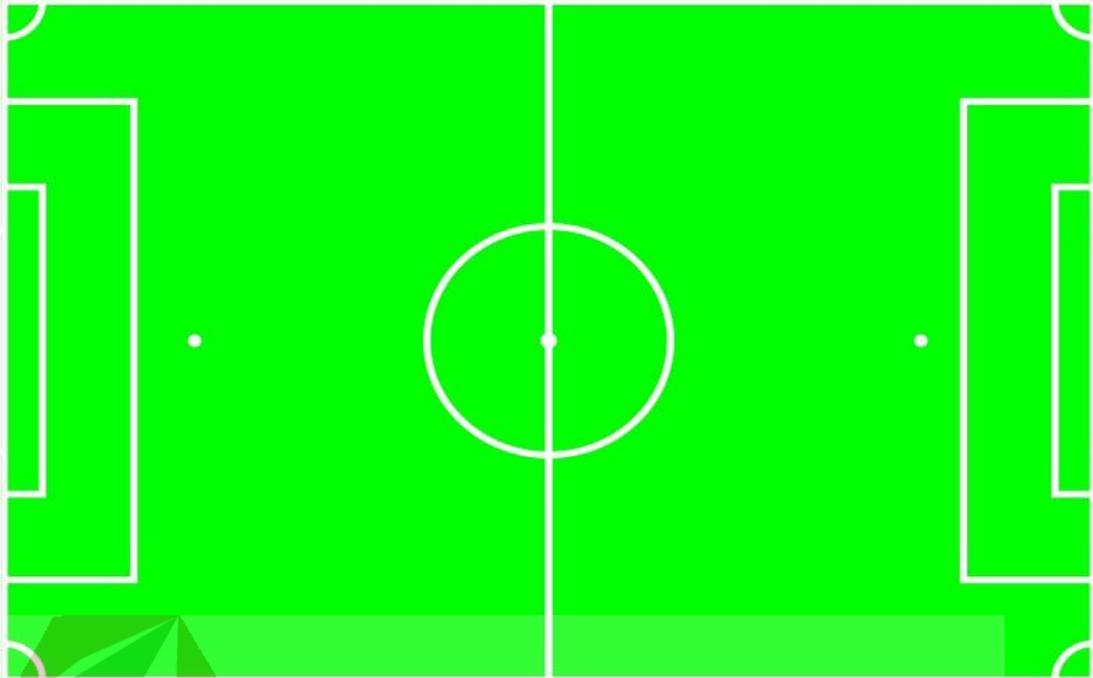
### 3.2.2 Rancang Lapangan Pada Visualisasi

Adapun bentuk lapangan yang sebenarnya memiliki panjang 9 meter dan lebar 6 meter. Pada perancangan visualisasi posisi robot menggunakan ketentuan 1 cm = 1 pixel yaitu dengan merubah panjang 9 m menjadi 900 cm dan lebar 6 m menjadi 600 cm. Karena untuk mendesain dalam aplikasi visual studio menggunakan satuan pixel maka dari cm akan dikonversikan menjadi *pixel* dengan berpedoman pada ketentuan 1 cm = 1 pixel. Untuk hal ini peneliti akan menentukannya mulai dari desain aplikasi terlebih dahulu, lalu menentukan skala dari lapangan sebenarnya yang satuannya meter (m) dijadikan senti meter (cm) agar

mendapatkan hasil yang pas untuk melihat di monitor PC nantinya. Karena telah ditentukan maka dalam merancang lapangan peneliti memiliki patokan yaitu 1 *pixel* sama dengan 1 *cm*, dengan begitu panjang lapangan menjadi 900 *pixel* dan lebarnya 600 *pixel*. Dapat dilihat lapangan robot sebenarnya pada gambar 3.7 dan Perancangan lapangan pada visualisasi posisi robot dapat dilihat pada gambar 3.8.



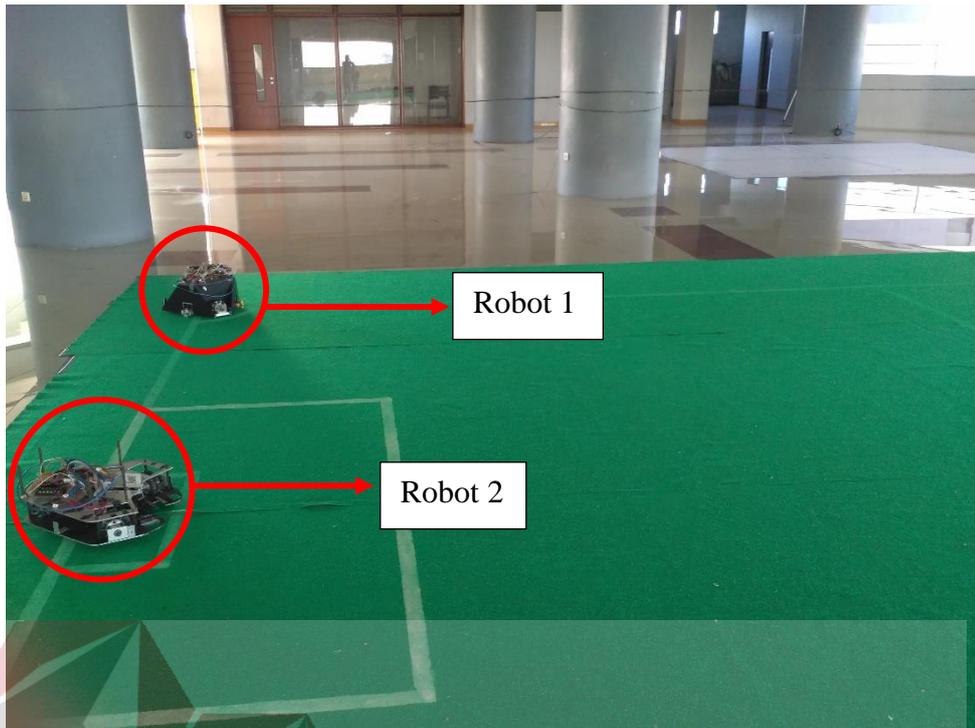
Gambar 3. 7 Lapangan Sebenarnya Robot Sepak Bola Beroda



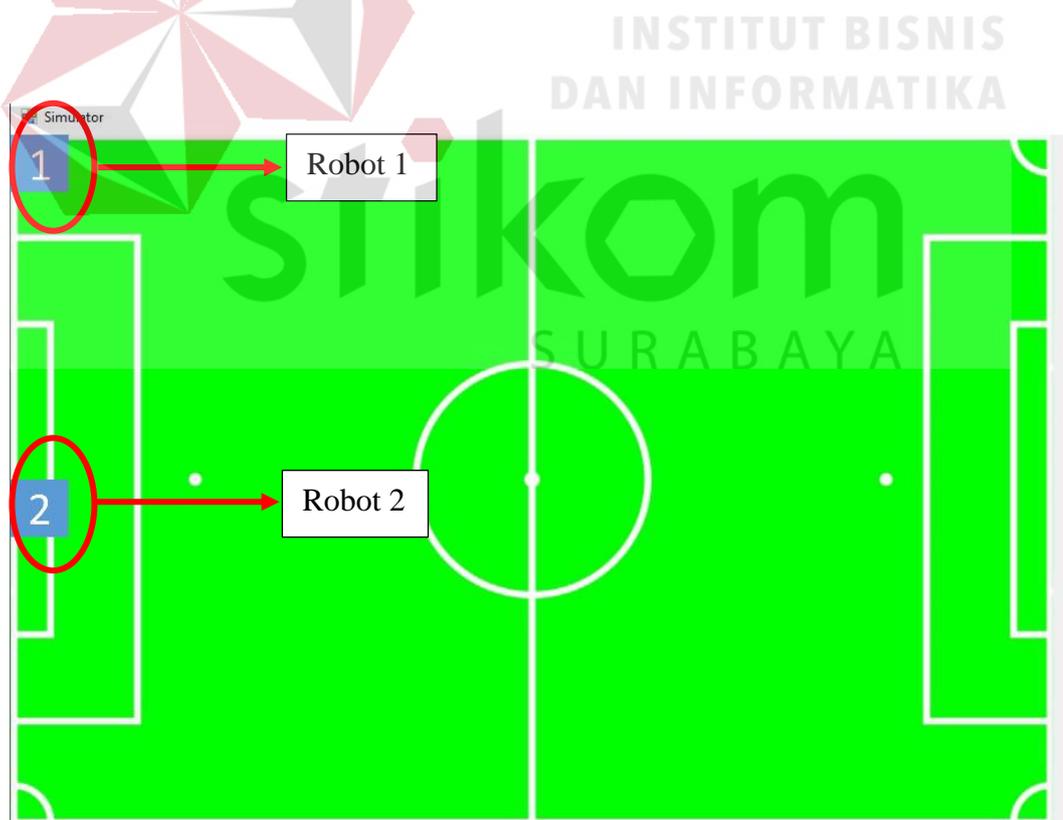
Gambar 3. 8 Lapangan Robot Sepak Bola Beroda Pada Visualisasi

### 3.2.3 Rancang Robot Pada Visualisasi

Robot yang digunakan untuk Tugas Akhir ini memiliki dimensi panjang dan lebar 50 *cm*. Maka tidak perlu diskala dan langsung diimplementasikan ke visualisasi dengan panjang dan lebar 50 *pixel*. Karena pada Tugas Akhir ini membutuhkan dua robot untuk diambil datanya maka peneliti akan membedakan nomor pada desain robot satu dengan dua. Untuk lebih jelasnya robot sebenarnya bisa dilihat pada gambar 3.9 dan untuk robot pada visualisasi posisi robot bisa dilihat pada gambar 3.10.



Gambar 3. 9 Robot Sebenarnya



Gambar 3. 10 Robot Pada Visualisasi

### 3.2.4 Rancang Tools Pada Visualisasi

Penulis membuat macam-macam tools dalam visualisasi ini agar memudahkan user dalam mengaksesnya. Agar terlihat sedikit rapi macam-macam tools dibuatkan grup sendiri sesuai dengan fungsinya. Adapun *tools* dalam visualisasi posisi robot dapat dilihat pada gambar 3.11.

The image shows a web application interface for robot control. It is divided into several sections:

- Cek Koordinat Gambar:** A section for checking image coordinates, showing "Koordinat X : 0 0" and "Koordinat Y : 0 0".
- Cek Robot:** A section for checking robot status, with checkboxes for "Robot1" and "Robot2", each followed by the word "Status".
- Robot 1:** A section for Robot 1, containing:
  - Input fields for "Masukkan Input X :" and "Masukkan Input Y :".
  - A "Kirim" button.
  - A "Posisi Start" button.
  - A "Hapus Kolom" button.
  - A large empty rectangular area for visualization.
- Robot 2:** A section for Robot 2, containing:
  - Input fields for "Masukkan Input X :" and "Masukkan Input Y :".
  - A "Kirim" button.
  - A "Posisi Start" button.
  - A "Hapus Kolom" button.
  - A large empty rectangular area for visualization.
- Kontrol Dua Robot:** A section at the bottom with two buttons: "Start Bersamaan" and "Home".

A large watermark "stikom SURABAYA" is overlaid on the interface.

Gambar 3. 11 Tampilan *Tools* Pada Visualisasi

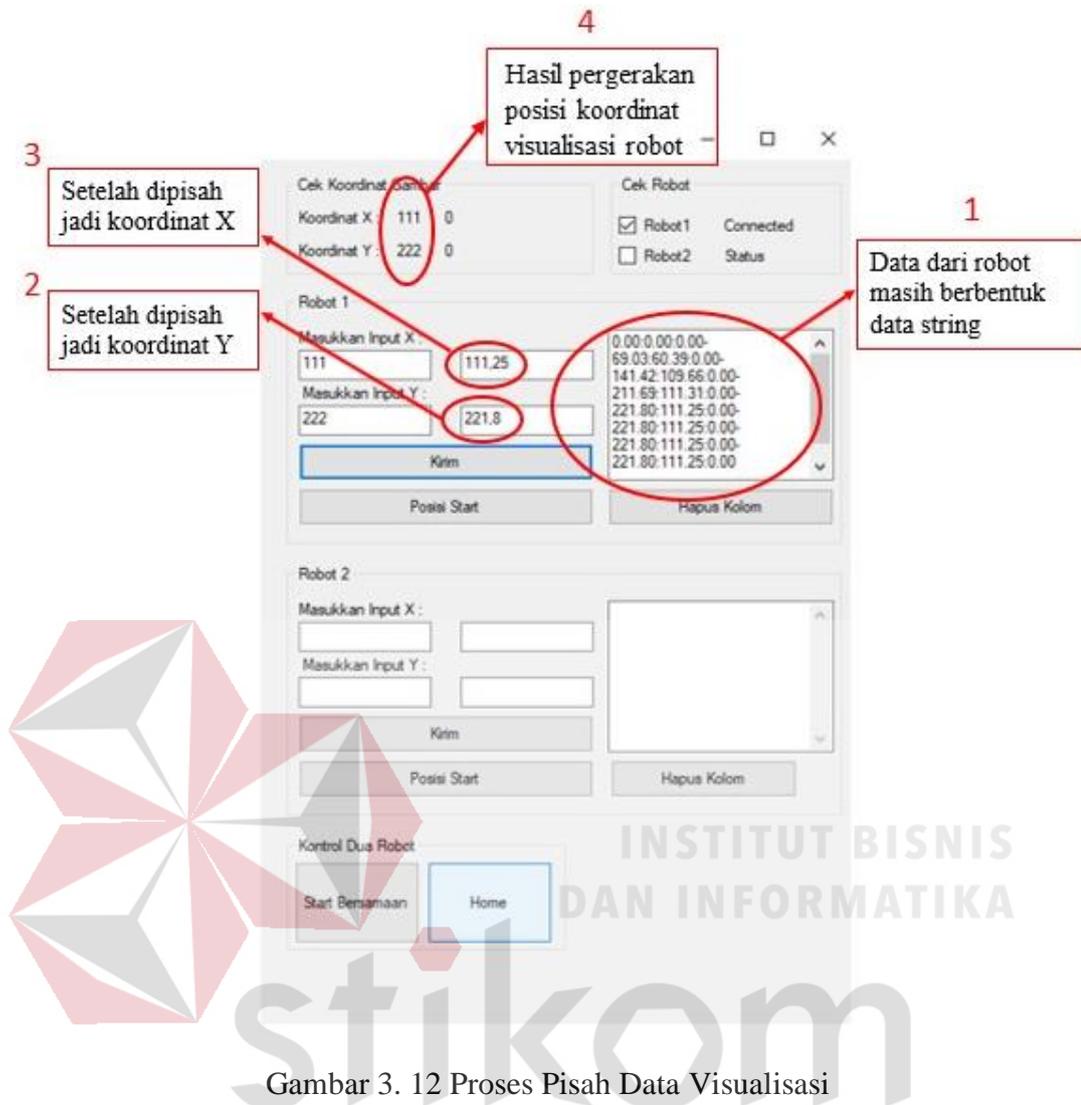
Terdapat macam-macam *tools* dalam pembuatan visualisasi posisi robot pada gambar 3.11 dan penulis akan menjelaskan *tools* pada tiap grup.

1. Cek Robot untuk mengkoneksikan robot yang mau dipilih. Setelah memilih robot untuk dikoneksikan maka tulisan status akan berubah menjadi terkoneksi, jika setelah megkoneksikan namun tidak dapat terkoneksi maka tulisan status akan berubah menjadi tidak terkoneksi.
2. Cek Koordinat Gambar untuk melihat status gambar robot berada pada koordinat mana pada saat terkoneksi.
3. Robot 1 untuk mengontrol pergerakan robot 1.
  - a. Button kirim yaitu untuk mengirim data ke robot koordinat x dan y yang telah diisi.
  - b. Button Posisi Start untuk pergerakan awal posisi start robot berada pada koordinat  $x = 0, y = 0$ .
  - c. Kolom untuk melihat data pergerakan rotary yang dikirim dari robot.
  - d. Button Hapus Kolom untuk menghapus tulisan yang ada pada kolom.
4. Robot 2 untuk mengontrol pergerakan robot 2.
  - a. Button kirim yaitu untuk mengirim data ke robot koordinat x dan y yang telah diisi.
  - b. Button Posisi Start untuk pergerakan awal posisi start robot berada pada koordinat  $x = 0, y = 300$ .
  - c. Kolom untuk melihat data pergerakan rotary yang dikirim dari robot.

- d. Button Hapus Kolom untuk menghapus tulisan yang ada pada kolom
5. Kontrol Dua Robot untuk mengontrol langsung pergerakan robot 1 dan robot 2
    - a. Start Bersamaan untuk memposisikan start awal kedua robot pada koordinat tertentu.
    - b. Home untuk memposisikan robot 1 pada posisi  $x=0, y=0$  dan robot 2  $x=0, y=300$ . Tombol home juga berfungsi untuk mengembalikan kedua robot yang tidak berada pada posisi start awal menjadi ke start awal.

### 3.2.5 Rancang Pisah Data Visualisasi

Data dari robot yang diterima dan ditampung oleh visualisasi masih berbentuk data string, maka dari itu perlu proses pemisahan data agar lebih mudah dalam mengolah data. Data yang ditampung tadi akan dipisahkan oleh karakter “:” supaya lebih mudah untuk mengolah data yang nantinya akan membedakan nilai koordinat X, Y dan Z. Perlu diketahui bahwa pada tugas akhir ini yang digunakan hanya koordinat X dan Y maka untuk koordinat Z tidak akan diproses. Data yang diterima dari robot bisa dilihat pada gambar 3.12 berikut.



Gambar 3. 12 Proses Pisah Data Visualisasi

Gambar 3.12 terlihat pada proses pertama visualisasi menerima data dari robot dan data masih berbentuk string belum melalui proses pemisah data. Pada proses kedua dan ketiga pemisahan data telah dilakukan dan akan langsung menjadi input pergerakan visualisasi robot. Proses keempat adalah proses pengecekan koordinat posisi visualisasi robot yaitu dengan cara membulatkan data inputan dari proses kedua yakni jika kurang dari 0,50 maka akan ikut pembulatan bawah, jika lebih dari 0,49 maka akan ikut pembulatan atas. Berikut ini penjelasan pengolahan data konversi koordinat dan tampilkan pergerakan.

1. Proses Konversi koordinat

Data dari robot yang telah diterima oleh akan di konversi dan dipisah agar bisa dijadikan sebagai input koordinat yang nantinya posisi visualisasi robot tersebut bisa bergerak sesuai nilai yang telah diproses tersebut.



Berikut potongan program konversi koordinat robot 1.

```

if (!data.Contains(">"))
{
    data1 += data;
}
else if (data.Contains(">"))
{
    data1 += data;
    for (int aa = 0; aa < data1.Length; aa++)
    {
        if (data1[aa] != '<' && data1[aa] != '>')
        {
            bantu1 += data1[aa];
        }
        else if (data1[aa] == '>')
        {
            bungkus1 = bantu1;
            s1 = true;
            break;
        }
    }
}

this.textBox3.Invoke(new MethodInvoker(delegate() {
textBox3.AppendText(bungkus1 + "\n"); }));

if (s1)
{
    xyz1 = bungkus1.Split(':');

    this.Invoke(new EventHandler(displaytext1));

    x1 = Convert.ToDouble(xyz1[0]);
    y1 = Convert.ToDouble(xyz1[1]);
    z1 = Convert.ToDouble(xyz1[2]);
    xx1 = (-1*x1)/100;
    yy1 = y1/100;

    data1 = "";
    bungkus1 = "";
    bantu1 = "";
    textBox5.Clear();
    textBox6.Clear();

    s1 = false;
}

```

Berikut potongan program konversi koordinat robot 2.

```

if (!data.Contains(">"))
{
    data2 += data;
}
else if (data.Contains(">"))
{
    data2 += data;
    for (int aa = 0; aa < data2.Length; aa++)
    {
        if (data2[aa] != '<' && data2[aa] != '>')
        {
            bantu2 += data2[aa];
        }
        else if (data2[aa] == '>')
        {
            bungkus2 = bantu2;
            s2 = true;
            break;
        }
    }
}

this.textBox14.Invoke(new MethodInvoker(delegate() {
textBox14.AppendText(bungkus2 + "\n"); }));

if (s2)
{
    xyz2 = bungkus2.Split(':');

    this.Invoke(new EventHandler(displaytext2));

    x2 = Convert.ToDouble(xyz2[0]);
    y2 = Convert.ToDouble(xyz2[1]);
    z2 = Convert.ToDouble(xyz2[2]);
    xx2 = (-1 * x2) / 100;
    yy2 = y2 / 100;

    data2 = "";
    bungkus2 = "";
    bantu2 = "";
    textBox11.Clear();
    textBox12.Clear();

    s2 = false;
}

```

## 2. Tampilkan Pergerakan

Setelah data diproses maka data akan dijadikan sebagai input koordinat supaya visualisasi robot bisa bergerak sesuai perubahan input koordinat.

Berikut potongan program pergerakan gambar robot 1.

```
private void timer1_Tick(object sender, EventArgs e)
{
    double angka1, angka2;
    angka1 = Convert.ToDouble(xx1);
    angka2 = Convert.ToDouble(yy1);

    pictureBox1.Location = new Point((pictureBox1.Location.X -
pictureBox1.Location.X) + (Convert.ToInt32(angka2)),
pictureBox1.Location.Y);
    pictureBox1.Location = new Point(pictureBox1.Location.X,
(pictureBox1.Location.Y - pictureBox1.Location.Y) +
(Convert.ToInt32(angka1)));
    //----- Nilai Output Koordinat
    label1.Text = pictureBox1.Location.X.ToString();
    label2.Text = pictureBox1.Location.Y.ToString();
}
```

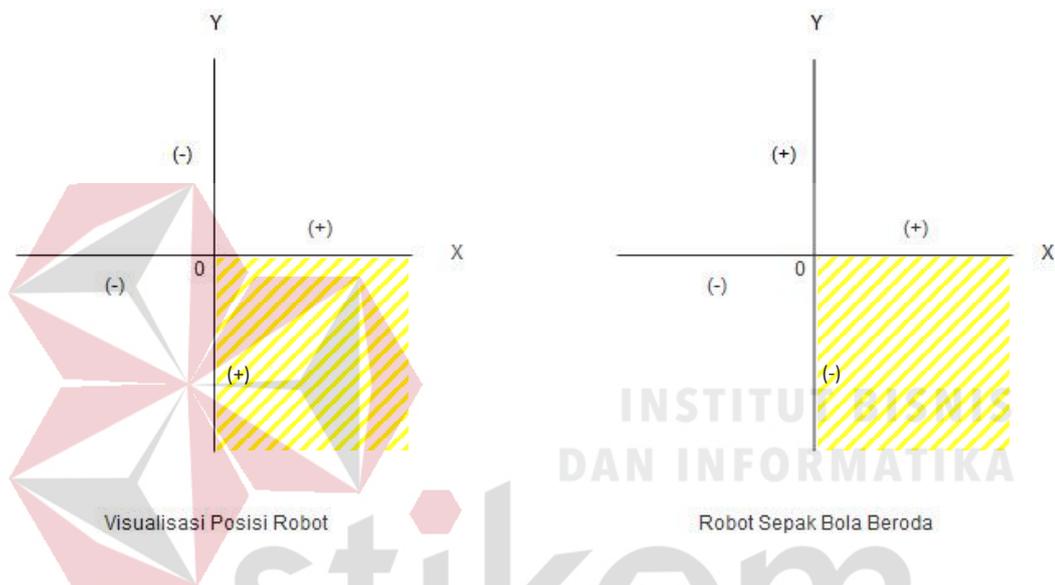
Berikut potongan program pergerakan gambar robot 2.

```
private void timer2_Tick(object sender, EventArgs e)
{
    double angka1, angka2;
    angka1 = Convert.ToDouble(xx2);
    angka2 = Convert.ToDouble(yy2);

    pictureBox3.Location = new Point((pictureBox3.Location.X -
pictureBox3.Location.X) + (Convert.ToInt32(angka2)),
pictureBox3.Location.Y);
    pictureBox3.Location = new Point(pictureBox3.Location.X,
(pictureBox3.Location.Y - pictureBox3.Location.Y) +
(Convert.ToInt32(angka1)));
    //----- Nilai Output Koordinat
    label16.Text = pictureBox3.Location.X.ToString();
    label17.Text = pictureBox3.Location.Y.ToString();
}
```

### 3.2.6 Perancangan Kartesius Pada Visualisasi dan Robot Sepak Bola Beroda

Koordinat objek pada visualisasi dan koordinat robot sepak bola beroda pada lapangan yang sebenarnya masih perlu dilakukan konversi, karena perhitungan koordinat objek pada robot dengan koordinat pada visualisasi sangat berbeda. Konversi yang dilakukan adalah dengan melakukan persamaan dari keduanya.



Gambar 3.13 Kartesius Visualisasi Dan Robot Sepak Bola Beroda

Gambar 3.13 menunjukkan bahwa nilai dari visualisasi sebelum dikirim ke robot perlu dilakukan konversi seperti pada persamaan 3.1. Sedangkan untuk persamaan 3.2 merupakan konversi nilai dari robot yang akan dilakukan dalam visualisasi agar dapat menampilkan pergerakan robot.

$$x = x\_akhir - x\_awal \quad (3.1)$$

$$y = - (y\_akhir - y\_awal)$$

$$x = x\_akhir + x\_awal \quad (3.2)$$

$$y = y\_akhir + y\_awal$$

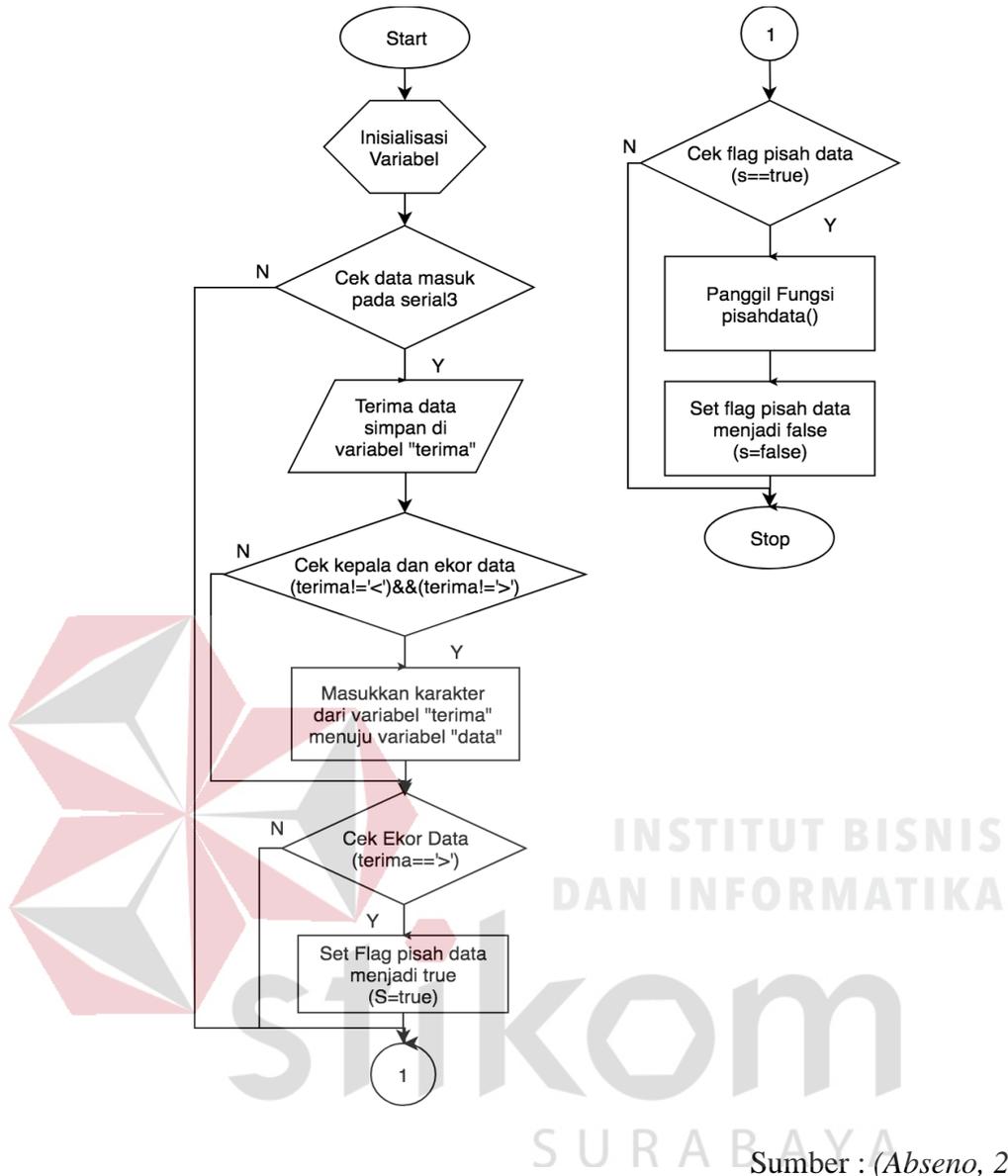
### 3.3 Perancangan Komunikasi Robot

Bagian yang menjelaskan mengenai visualisasi dan robot yang saling terkoneksi agar dapat bertukar data guna untuk mengontrol dan memonitoring pergerakan robot.

#### 3.3.1 Terima Data Koordinat

Setelah visualisasi mengirim data berupa koordinat tujuan pada robot, menurut (Abseno, 2019) perlu adanya proses penerimaan data dengan menggunakan komunikasi serial yang diterima dari wemos ke mikrontroler. Berikut adalah flowchart untuk melakukan penerimaan data koordinat pada gambar 3.13.





Sumber : (Abseno, 2019)

Gambar 3. 14 Flowchart Terima Data Koordinat Robot

Berikut adalah potongan program dari proses pemisahan data:

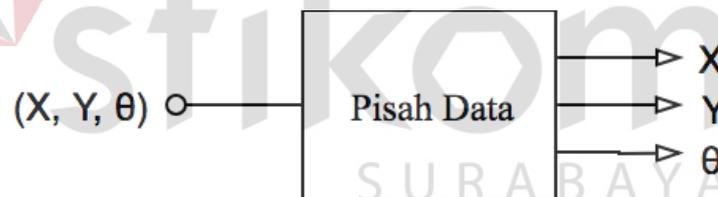
```

if (Serial3.available())
{
    terima = (char)Serial3.read();
    if((terima!='<') && (terima!='>')) data+=terima;
    if(terima=='>') s=true;
}
if(s)
{
    pisahdata();
    s=false;
}

```

### 3.3.2 Pemisah Data

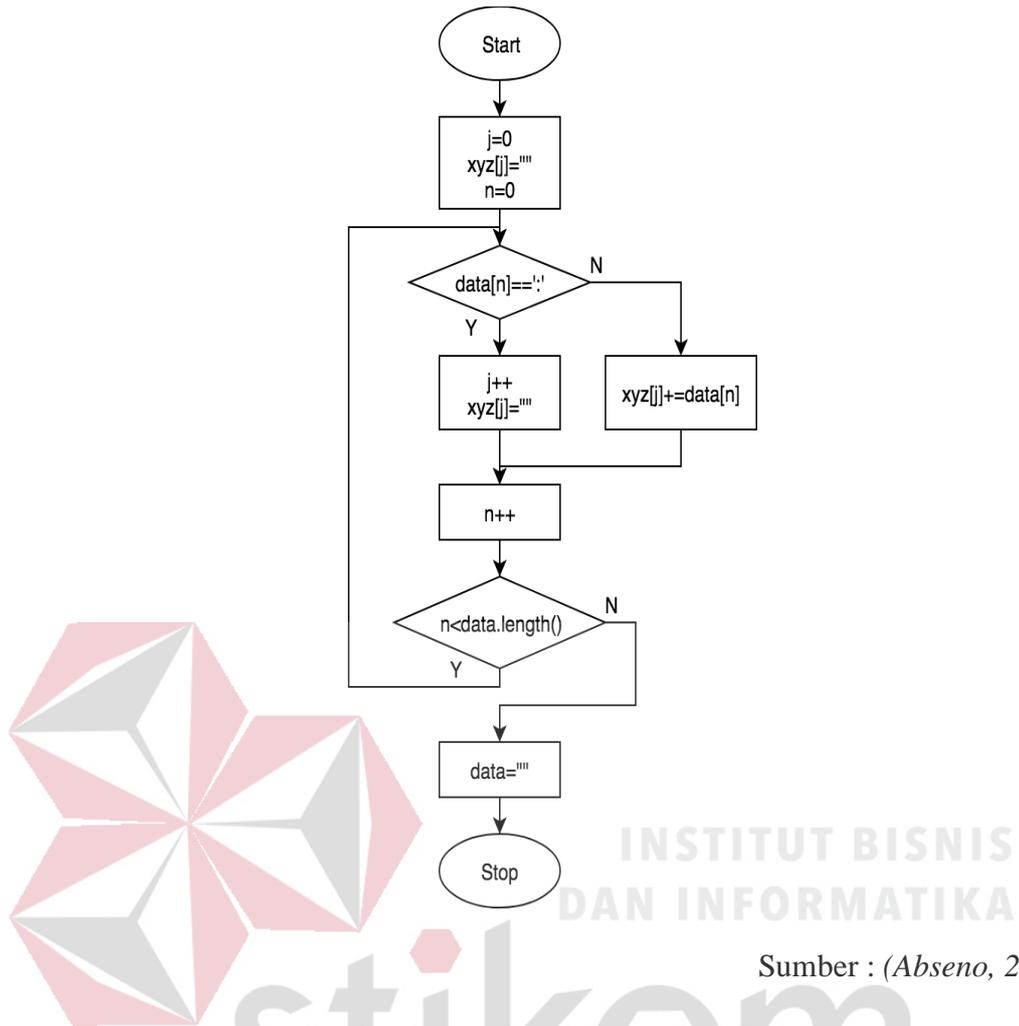
Setelah melakukan penerimaan data maka proses selanjutnya yang perlu dilakukan adalah pemisahan data.



Sumber : (Abseno, 2019)

Gambar 3. 15 Pemisah Data X, Y,  $\Theta$

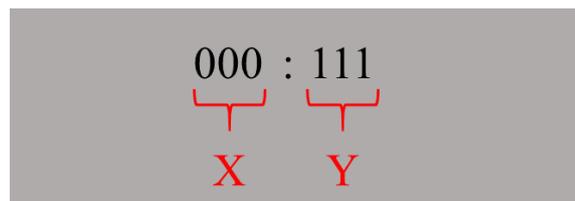
Dengan proses ini data yang semula masih berbentuk data string dan belum dikelompokkan menjadi satuan individu akan diproses menjadi suatu data yang siap dilakukan perhitungan seperti pada gambar 3.14 sehingga mempermudah proses perhitungan pada sistem (Abseno, 2019). Berikut adalah *flowchart* dari proses pemisahan data pada gambar 3.15 berikut ini :



Sumber : (Abseno, 2019)

Gambar 3. 16 Flowchart Pemisah Data

Data masuk yang masih berupa data string akan diseleksi sebagai pembatas menggunakan karakter “:”. Karakter tersebut berfungsi untuk memudahkan proses memisah data pada mikrokontroler arduino. Penjelasan fungsi dari setiap data dan karakter “:” akan dijelaskan pada format gambar 3.16 berikut.



Gambar 3. 17 format pemisah data

Pada gambar 3.16 menjelaskan bahwa karakter “:” merupakan pemisah data antara koordinat X dan koordinat Y. Data yang bernilai 000 akan ditampung dalam variabel dan diproses yang nantinya akan diketahui sebagai nilai koordinat X yang harus dicapai oleh robot, begitu juga data bernilai 111 akan ditampung dalam variabel dan diproses yang nantinya akan diketahui sebagai nilai koordinat Y yang harus dicapai oleh robot.

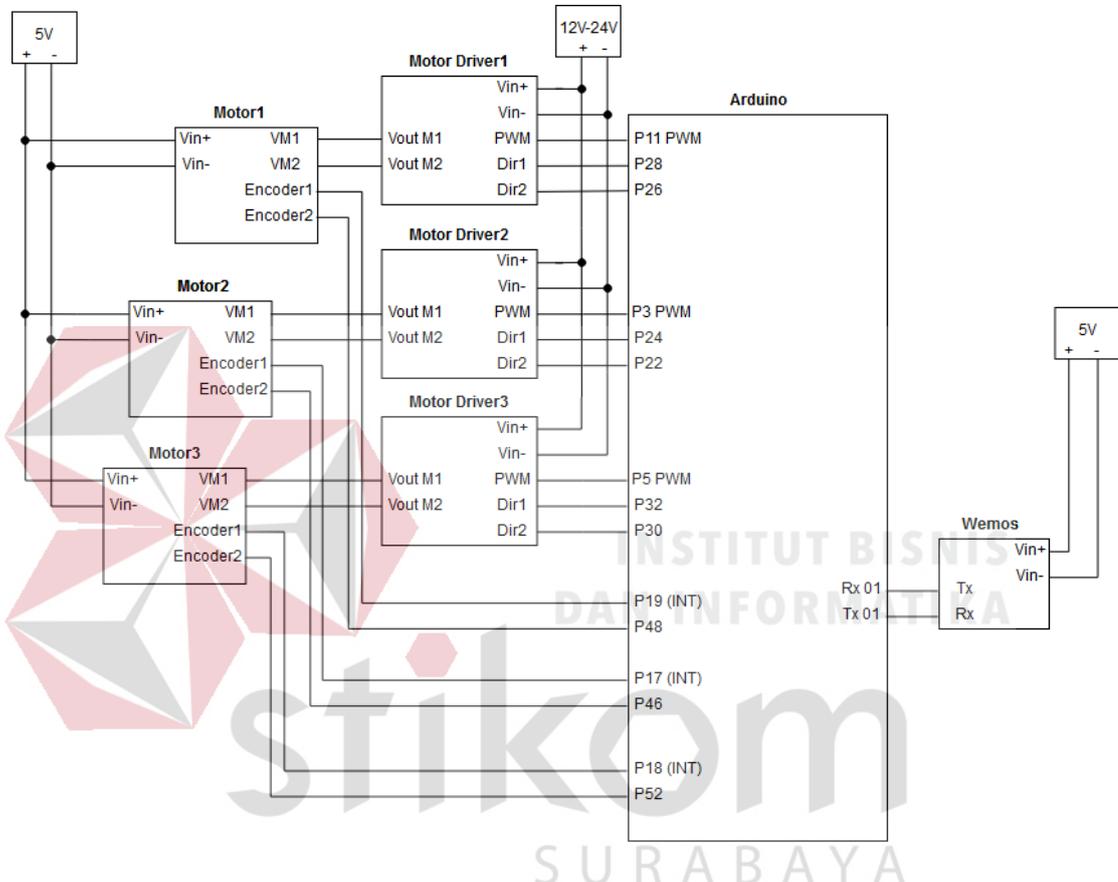
Berikut adalah potongan program dari proses pemisahan data :

```
void pisahdata ()
{
    j=0;
    xyz[j]="";
    for(int n=0;n<data.length();n++)
    {
        if(data[n]==':')
        {
            j++;
            xyz[j]="";
        }
        else
        {
            xyz[j]+=data[n];
        }
    }
    data="";
}
```



### 3.4 Perancangan Komunikasi Data

Pada robot sepak bola beroda ini menggunakan mikrokontroler arduino untuk mengontrol robot, dan wemos sebagai pengiriman data dari PC ke mikrokontroler. Untuk lebih jelasnya dapat dilihat pada gambar 3.17.



Gambar 3. 18 Blok Diagram Perancangan Komunikasi Data

Proses pada gambar 3.17 terlihat terdapat dua daya yang diperlukan yaitu 5 volt dan 12-24 volt. Daya 5 volt digunakan sebagai sumber daya meliputi mikrokontroler, wemos, motor driver, rotary encoder. Sedangkan daya 12-24 volt digunakan sebagai peggerak aktuator yaitu motor.

Motor driver yang digunakan adalah motor driver dengan kapasitas ampere yang cukup besar yaitu sebesar 30 ampere sesuai dengan tipenya EMS-30A, pada

Tugas Akhir ini motor driver mengendalikan motor DC dengan sumber tegangan 12V yang berasal dari baterai lithium polymer 3 sel. Untuk mengendalikan motor driver ini diperlukan sebanyak 3 masukan yaitu direksi 1, direksi 2, dan kontrol PWM. Direksi 1 dan direksi 2 digunakan sebagai menentukan arah putaran motor, saat diberi dengan kondisi (high) untuk direksi 1 dan (low) untuk direksi 2 maka motor akan berputar searah jarum jam dan berlaku juga sebaliknya, namun arah putaran motor juga dapat dipengaruhi dari input sumber tegangan motor yang dibalik untuk port VMotor pada motor driver. Untuk kontrol PWM dilakukan dengan memberikan input berupa pulsa dengan perbandingan duty cycle sesuai kebutuhan guna menentukan kecepatan putar motor DC.

Rotary encoder menggunakan rotary encoder internal yang ada pada motor DC. Dalam membaca arah putaran motor maka digunakanlah kedua output dari rotary encoder yaitu output A dan output B, dikarenakan kedua output tersebut mempunyai output pulse yang berdampingan. Salah satu output dari rotary encoder dihubungkan kepada pin kontroler yang telah diset menjadi pin interrupt untuk melakukan pembacaan putaran.

## **BAB IV**

### **PENGUJIAN**

Dalam bab ini penulis tidak akan membahas tentang ketepatan gerak robot sebenarnya tetapi akan menjelaskan tentang sistem yang telah dirancang pada aplikasi visualisasi dan begitu juga beberapa hasil pengujian. Tujuan pembahasan dalam bab ini untuk mengetahui tingkat keberhasilan sistem yang telah dirancang dan dikerjakan oleh penulis. Terdapat dua pengujian dalam bab ini yaitu meliputi uji komunikasi antara robot dengan visualisasi, dan uji ketepatan koordinat robot sepak bola beroda dengan visualisasi. Pada setiap pengujian menggunakan sampel pergerakan robot ada tiga yaitu dengan gerak ke samping, gerak maju, dan gerak serong. Dari setiap pergerakan akan diambil sampel sebanyak 5 kali percobaan.

#### **4.1 Uji Komunikasi Antara Robot Dengan Visualisasi**

Pengujian ini berfungsi untuk mengetahui data dari koordinat robot sepak bola beroda yang terkirim ke visualisasi telah sesuai atau tidak. Data yang dikirimkan dari robot sepak bola beroda ke visualisasi merupakan hasil dari jarak tempuh perjalanan menuju koordinat yang telah di inputkan melalui visualisasi.

##### **4.1.1 Alat Yang Digunakan Pada Uji Komunikasi Antara Robot Dengan Visualisasi**

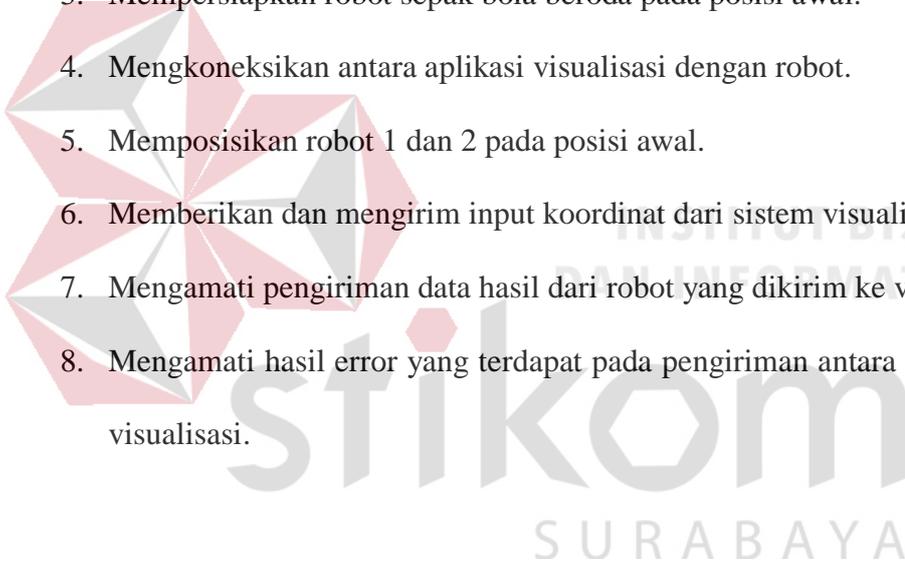
Peralatan yang diperlukan untuk pengujian ini adalah sebagai berikut :

1. Robot Sepak Bola Beroda
2. Komputer/laptop
3. Aplikasi visualisasi

#### **4.1.2 Prosedur Pengujian Pada Uji Komunikasi Antara Robot Dengan Visualisasi**

Langkah-langkah yang dilakukan dalam melakukan pengujian ini adalah sebagai berikut :

1. Mempersiapkan komputer untuk melihat pengiriman data dari serial monitor pada arduino.
2. Mempersiapkan aplikasi visualisasi pada komputer untuk kontrol robot sepak bola beroda.
3. Mempersiapkan robot sepak bola beroda pada posisi awal.
4. Mengkoneksikan antara aplikasi visualisasi dengan robot.
5. Memposisikan robot 1 dan 2 pada posisi awal.
6. Memberikan dan mengirim input koordinat dari sistem visualisasi robot.
7. Mengamati pengiriman data hasil dari robot yang dikirim ke visualisasi.
8. Mengamati hasil error yang terdapat pada pengiriman antara robot dengan visualisasi.



### 4.1.3 Hasil Uji Komunikasi Antara Robot Dengan Visualisasi

Pada tabel 4.1 berikut adalah hasil uji komunikasi antara robot 1 ke visualisasi dengan gerak robot ke samping.

Tabel 4. 1 Uji Komunikasi Antara Robot 1 Ke Visualisasi Dengan Gerak Robot Ke Samping

No.	Input Data Koordinat (cm)		Data Yang Dikirim Arduino (cm)		Data Yang Diterima Aplikasi (cm)		Error (cm)	
	X	Y	X	Y	X	Y	X	Y
1	0	100	0,00	0,00	0,00	0,00	0	0
			0,00	10,35	0,00	10,35		
			0,00	76,91	0,00	76,91		
			0,00	100,02	0,00	100,02		
2	0	100	0,00	0,00	0,00	0,00	0	0
			0,18	60,10	0,18	60,10		
			0,06	99,73	0,06	99,73		
3	0	100	0,00	0,00	0,00	0,00	0	0
			0,06	10,35	0,06	10,35		
			0,18	76,09	0,18	76,09		
			-0,29	100,26	-0,29	100,26		
4	0	100	0,00	0,00	0,00	0,00	0	0
			0,00	10,06	0,00	10,06		
			0,12	74,09	0,12	74,09		
			-0,12	100,02	-0,12	100,02		
5	0	100	0,00	0,00	0,00	0,00	0	0
			0,00	36,16	0,00	36,16		
			0,24	101,90	0,24	101,90		
			-0,18	99,96	-0,18	99,96		
6	0	200	0,00	0,00	0,00	0,00	0	0

			0,18	10,29	0,18	10,29		
			0,18	52,27	0,18	52,27		
			0,29	94,73	0,29	94,73		
			0,24	137,60	0,24	137,60		
			0,18	180,76	0,18	180,76		
			0,06	200,45	0,06	200,45		
7	0	200	0,00	0,00	0,00	0,00	0	0
			0,29	17,52	0,29	17,52		
			0,24	58,86	0,24	58,86		
			0,24	100,96	0,24	100,96		
			0,12	142,53	0,12	142,53		
			0,29	186,40	0,29	186,40		
			0,12	200,57	0,12	200,57		
8	0	200	0,00	0,00	0,00	0,00	0	0
			0,24	28,70	0,24	28,70		
			0,18	71,09	0,18	71,09		
			0,18	113,25	0,18	113,25		
			0,24	156,53	0,24	156,53		
			0,12	198,34	0,12	198,34		
			0,06	200,34	0,06	200,34		
9	0	200	0,00	0,00	0,00	0,00	0	0
			0,24	35,57	0,24	35,57		
			0,24	77,97	0,24	77,97		
			0,29	119,54	0,29	119,54		
			0,24	162,00	0,24	162,00		
			0,18	200,45	0,18	200,45		
10	0	200	0,00	0,00	0,00	0,00	0	0
			0,18	8,35	0,18	8,35		
			0,12	51,22	0,12	51,22		
			0,24	93,79	0,24	93,79		
			0,24	135,54	0,24	135,54		
			0,24	177,52	0,24	177,52		

			0,12	200,16	0,12	200,16		
11	0	300	0,00	0,00	0,00	0,00	0	0
			0,12	21,46	0,12	21,46		
			0,12	60,51	0,12	60,51		
			0,24	99,37	0,24	99,37		
			0,06	180,93	0,06	180,93		
			0,24	222,15	0,24	222,15		
			0,06	263,67	0,06	263,67		
			0,12	300,48	0,12	300,48		
			0,12	300,42	0,12	300,42		
12	0	300	0,00	0,00	0,00	0,00	0	0
			0,12	21,58	0,12	21,58		
			0,24	61,57	0,24	61,57		
			0,18	101,43	0,18	101,43		
			0,18	141,77	0,18	141,77		
			0,24	182,70	0,24	182,70		
			0,12	224,50	0,12	224,50		
			0,18	267,25	0,18	267,25		
			0,00	300,36	0,00	300,36		
13	0	300	0,00	0,00	0,00	0,00	0	0
			0,12	5,70	0,12	5,70		
			0,12	45,10	0,12	45,10		
			0,12	84,44	0,12	84,44		
			0,18	124,13	0,18	124,13		
			0,18	164,53	0,18	164,53		
			0,18	205,39	0,18	205,39		
			0,24	246,26	0,24	246,26		
			0,06	287,54	0,06	287,54		
			0,12	300,42	0,12	300,42		
14	0	300	0,00	0,00	0,00	0,00	0	0
			0,06	10,53	0,06	10,53		
			0,00	77,50	0,00	77,50		

			0,18	145,12	0,18	145,12		
			0,00	212,69	0,00	212,69		
			0,12	280,31	0,12	280,31		
			-0,18	299,95	-0,18	299,95		
15	0	300	0,00	0,00	0,00	0,00	0	0
			0,00	64,56	0,00	64,56		
			0,24	131,77	0,24	131,77		
			-0,06	199,51	-0,06	199,51		
			0,00	267,49	0,00	267,49		
			0,00	300,12	0,00	300,12		
Rata-rata							0	0

Berdasarkan tabel 4.1 ditunjukkan bahwa nilai input koordinat (0,100), (0,200), (0,300) setelah di uji menghasilkan nilai error 0% . Sebagai contoh pada pengujian input (0,100) data komunikasi yang diperoleh antara robot dengan visualisasi yakni sama. Koordinat x menempuh 0.00, 0.00, 0.00, 0.00 sedangkan y menempuh 0.00, 10.35, 76.91, 100.02.

Pada tabel 4.2 berikut adalah hasil uji komunikasi antara robot 2 ke visualisasi dengan gerak robot ke samping.

Tabel 4. 2 Uji Komunikasi Antara Robot 2 Ke Visualisasi Dengan Gerak Robot Ke Samping

No.	Input Data Koordinat (cm)		Data Yang Dikirim Arduino (cm)		Data Yang Diterima Aplikasi (cm)		Error (cm)	
	X	Y	X	Y	X	Y	X	Y
1	0	400	0,00	300,00	0,00	300,00	0	0
			0,00	376,02	0,00	376,02		
			0,00	402,26	0,00	402,26		
2	0	400	0,00	300,00	0,00	300,00	0	0
			0,00	322,28	0,00	322,28		
			0,00	402,58	0,00	402,58		
3	0	400	0,00	300,00	0,00	300,00	0	0
			0,16	402,73	0,16	402,73		
			0,16	402,26	0,16	402,26		
4	0	400	0,00	300,00	0,00	300,00	0	0
			0,00	314,47	0,00	314,47		
			0,00	402,58	0,00	402,58		
5	0	400	0,00	300,00	0,00	300,00	0	0
			0,00	314,62	0,00	314,62		
			0,00	402,26	0,00	402,26		
6	0	500	0,00	300,00	0,00	300,00	0	0
			0,00	325,25	0,00	325,25		
			0,00	466,00	0,00	466,00		
			0,00	502,56	0,00	502,56		
7	0	500	0,00	300,00	0,00	300,00	0	0
			0,00	324,62	0,00	324,62		
			0,00	466,63	0,00	466,63		

			0,00	502,09	0,00	502,09		
8	0	500	0,00	300,00	0,00	300,00	0	0
			0,00	325,09	0,00	325,09		
			0,00	466,94	0,00	466,94		
			0,00	502,24	0,00	502,24		
9	0	500	0,00	300,00	0,00	300,00	0	0
			0,00	320,87	0,00	320,87		
			0,16	462,56	0,16	462,56		
			0,00	502,71	0,00	502,71		
10	0	500	0,00	300,00	0,00	300,00	0	0
			0,16	318,37	0,16	318,37		
			0,00	461,16	0,00	461,16		
			0,00	502,56	0,00	502,56		
11	0	600	0,00	300,00	0,00	300,00	0	0
			0,00	381,80	0,00	381,80		
			0,00	524,11	0,00	524,11		
			0,00	602,69	0,00	602,69		
12	0	600	0,00	300,00	0,00	300,00	0	0
			0,00	318,22	0,00	318,22		
			0,00	461,31	0,00	461,31		
			0,16	602,54	0,16	602,54		
13	0	600	0,00	300,00	0,00	300,00	0	0
			0,00	305,56	0,00	305,56		
			0,00	448,04	0,00	448,04		
			0,00	592,38	0,00	592,38		
			0,00	602,07	0,00	602,07		
14	0	600	0,00	300,00	0,00	300,00	0	0
			0,16	328,84	0,16	328,84		
			0,00	472,72	0,00	472,72		
			0,31	602,54	0,31	602,54		
15	0	600	0,00	300,00	0,00	300,00	0	0
			0,00	314,94	0,00	314,94		

			0,00	458,35	0,00	458,35		
			0,16	601,60	0,16	601,60		
Rata-rata							0	0

Berdasarkan tabel 4.2 ditunjukkan bahwa nilai input koordinat (0,400), (0,500), (0,600) setelah di uji menghasilkan nilai error 0% . Sebagai contoh pada pengujian input (0,400) data komunikasi yang diperoleh antara robot dengan visualisasi yakni sama. Koordinat x menempuh 0.00, 0.00, 0.00 sedangkan y menempuh 300.00, 376.02, 402.26.



Pada tabel 4.3 berikut adalah hasil uji komunikasi antara robot 1 ke visualisasi dengan gerak robot maju.

Tabel 4. 3 Uji Komunikasi Antara Robot 1 Ke Visualisasi Dengan Gerak Robot  
Maju

No.	Input Data Koordinat (cm)		Data Yang Dikirim Arduino (cm)		Data Yang Diterima Aplikasi (cm)		Error (cm)	
	X	Y	X	Y	X	Y	X	Y
1	150	0	0,00	0,00	0,00	0,00	0	0
			25,99	0,12	25,99	0,12		
			60,92	0,12	60,92	0,12		
			94,79	0,24	94,79	0,24		
			128,48	0,18	128,48	0,18		
			149,65	0,18	149,65	0,18		
2	150	0	0,00	0,00	0,00	0,00	0	0
			17,82	0,12	17,82	0,12		
			52,63	0,12	52,63	0,12		
			86,97	0,18	86,97	0,18		
			121,90	0,12	121,90	0,12		
			150,06	0,12	150,06	0,12		
3	150	0	0,00	0,00	0,00	0,00	0	0
			0,76	0,12	0,76	0,12		
			31,34	0,18	31,34	0,18		
			62,92	0,24	62,92	0,24		
			95,02	0,24	95,02	0,24		
			127,66	0,18	127,66	0,18		
149,83	0,00	149,83	0,00					
4	150	0	0,00	0,00	0,00	0,00	0	0
			8,17	0,18	8,17	0,18		
			41,57	0,18	41,57	0,18		

			74,74	0,12	74,74	0,12		
			108,31	0,06	108,31	0,06		
			143,59	0,06	143,59	0,06		
			150,12	0,06	150,12	0,06		
5	150	0	0,00	0,00	0,00	0,00	0	0
			8,64	0,18	8,64	0,18		
			41,98	0,24	41,98	0,24		
			74,62	0,24	74,62	0,24		
			106,61	0,18	106,61	0,18		
			139,77	0,18	139,77	0,18		
			149,71	0,24	149,71	0,24		
6	200	0	0,00	0,00	0,00	0,00	0	0
			35,57	0,12	35,57	0,12		
			73,68	0,00	73,68	0,00		
			111,66	0,00	111,66	0,00		
			150,59	0,06	150,59	0,06		
			188,64	0,06	188,64	0,06		
			200,28	0,00	200,28	0,00		
7	200	0	0,00	0,00	0,00	0,00	0	0
			45,60	0,06	45,60	0,06		
			77,68	0,12	77,68	0,12		
			120,50	0,06	120,50	0,06		
			150,59	0,06	150,59	0,06		
			188,01	0,06	188,01	0,06		
			200,50	0,00	200,50	0,00		
8	200	0	0,00	0,00	0,00	0,00	0	0
			13,17	0,12	13,17	0,12		
			53,27	0,12	53,27	0,12		
			92,67	0,12	92,67	0,12		
			132,89	0,18	132,89	0,18		
			172,29	0,18	172,29	0,18		
			200,28	0,06	200,28	0,06		

9	200	0	0,00	0,00	0,00	0,00	0	0
			11,05	0,12	11,05	0,12		
			82,56	0,06	82,56	0,06		
			154,24	0,06	154,24	0,06		
			199,69	0,00	199,69	0,00		
10	200	0	0,00	0,00	0,00	0,00	0	0
			4,41	0,12	4,41	0,12		
			75,15	0,06	75,15	0,06		
			146,77	0,06	146,77	0,06		
			199,81	0,06	199,81	0,06		
11	450	0	0,00	0,00	0,00	0,00	0	0
			38,87	0,12	38,87	0,12		
			79,73	0,06	79,73	0,06		
			119,66	0,00	119,66	0,00		
			159,59	0,06	159,59	0,06		
			198,22	0,00	198,22	0,00		
			236,56	0,12	236,56	0,12		
			275,60	0,12	275,60	0,12		
			314,76	0,06	314,76	0,06		
			353,81	0,12	353,81	0,12		
			393,15	0,18	393,15	0,18		
			431,96	0,06	431,96	0,06		
			450,30	0,00	450,30	0,00		
	450	0	0,00	0,00	0,00	0,00	0	0
			28,99	0,12	28,99	0,12		
			64,09	0,12	64,09	0,12		
			99,55	0,12	99,55	0,12		
			135,36	0,12	135,36	0,12		
			171,70	0,12	171,70	0,12		
			207,86	0,18	207,86	0,18		
			244,50	0,18	244,50	0,18		
			281,01	0,18	281,01	0,18		

			318,00	0,06	318,00	0,06		
			355,22	0,18	355,22	0,18		
			392,38	0,12	392,38	0,12		
			429,49	0,12	429,49	0,12		
			450,30	0,00	450,30	0,00		
	450	0	0,00	0,00	0,00	0,00	0	0
			18,93	0,00	18,93	0,00		
			56,45	0,06	56,45	0,06		
			93,38	0,06	93,38	0,06		
			130,66	0,06	130,66	0,06		
			167,23	0,12	167,23	0,12		
			203,81	0,12	203,81	0,12		
			240,15	0,06	240,15	0,06		
			277,78	0,12	277,78	0,12		
			314,29	0,18	314,29	0,18		
			350,81	0,12	350,81	0,12		
			386,38	0,06	386,38	0,06		
			422,43	0,12	422,43	0,12		
			450,36	0,06	450,36	0,06		
14	450	0	0,00	0,00	0,00	0,00	0	0
			11,00	0,12	11,00	0,12		
			82,62	0,06	82,62	0,06		
			154,24	0,06	154,24	0,06		
			225,86	0,06	225,86	0,06		
			297,48	0,06	297,48	0,06		
			369,10	0,06	369,10	0,06		
			440,72	0,06	440,72	0,06		
			450,18	0,12	450,18	0,12		
15	450	0	0,00	0,00	0,00	0,00	0	0
			7,76	0,12	7,76	0,12		
			79,21	0,06	79,21	0,06		
			150,83	0,06	150,83	0,06		

			222,45	0,06	222,45	0,06		
			294,07	0,06	294,07	0,06		
			365,69	0,06	365,69	0,06		
			437,31	0,06	437,31	0,06		
			450,36	0,06	450,36	0,06		
Rata-rata							0	0

Berdasarkan tabel 4.3 ditunjukkan bahwa nilai input koordinat (150,0), (200,0), (450,0) setelah di uji menghasilkan nilai error 0% . Sebagai contoh pada pengujian input (150,0) data komunikasi yang diperoleh antara robot dengan visualisasi yakni sama. Koordinat x menempuh 0.00, 25.99, 60.92, 94.79, 128.48, 149.65 sedangkan y menempuh 0.00, 0.12, 0.12, 0.24, 0.18, 0.18.



Pada tabel 4.4 berikut adalah hasil uji komunikasi antara robot ke visualisasi dengan gerak robot secara maju.

Tabel 4. 4 Uji Komunikasi Antara Robot 2 Ke Visualisasi Dengan Gerak Robot  
Maju

No.	Input Data Koordinat (cm)		Data Yang Dikirim Arduino (cm)		Data Yang Diterima Aplikasi (cm)		Error (cm)	
	X	Y	X	Y	X	Y	X	Y
1	100	300	0,00	300,00	0,00	300,00	0	0
			87,80	300,47	87,80	300,47		
			99,36	300,16	99,36	300,16		
2	100	300	0,00	300,00	0,00	300,00	0	0
			29,37	300,00	29,37	300,00		
			99,20	300,00	99,20	300,00		
3	100	300	0,00	300,00	0,00	300,00	0	0
			78,42	300,31	78,42	300,31		
			99,51	300,00	99,51	300,00		
4	100	300	0,00	300,00	0,00	300,00	0	0
			4,06	300,00	4,06	300,00		
			99,51	300,00	99,51	300,00		
5	100	300	0,00	300,00	0,00	300,00	0	0
			4,06	300,00	4,06	300,00		
			100,14	300,00	100,14	300,00		
6	200	300	0,00	300,00	0,00	300,00	0	0
			147,16	300,16	147,16	300,16		
			198,87	300,00	198,87	300,00		
7	200	300	0,00	300,00	0,00	300,00	0	0
			20,62	300,00	20,62	300,00		
			204,33	300,31	204,33	300,31		
			199,02	300,00	199,02	300,00		

8	200	300	0,00 58,27 199,34	300,00 300,31 300,00	0,00 58,27 199,34	300,00 300,31 300,00	0	0
9	200	300	0,00 20,31 204,18 199,34	300,00 300,00 300,16 300,00	0,00 20,31 204,18 199,34	300,00 300,00 300,16 300,00	0	0
10	200	300	0,00 11,72 205,12 198,87	300,00 300,16 300,16 300,16	0,00 11,72 205,12 198,87	300,00 300,16 300,16 300,16	0	0
11	300	300	0,00 97,95 295,72 299,32	300,00 300,16 300,16 300,00	0,00 97,95 295,72 299,32	300,00 300,16 300,16 300,00	0	0
12	300	300	0,00 78,27 276,20 299,32	300,00 300,31 300,16 300,00	0,00 78,27 276,20 299,32	300,00 300,31 300,16 300,00	0	0
13	300	300	0,00 20,31 217,61 298,69 298,85 299,00	300,00 300,16 300,31 300,16 300,16 300,31	0,00 20,31 217,61 298,69 298,85 299,00	300,00 300,16 300,31 300,16 300,16 300,31	0	0
14	300	300	0,00 78,27 276,82 299,00	300,00 300,16 300,16 300,16	0,00 78,27 276,82 299,00	300,00 300,16 300,16 300,16	0	0
15	300	300	0,00 137,94 297,91	300,00 300,00 300,00	0,00 137,94 297,91	300,00 300,00 300,00	0	0

			299,63	300,00	299,63	300,00		
			299,78	300,00	299,78	300,00		
Rata-Rata							0	0

Berdasarkan tabel 4.4 ditunjukkan bahwa nilai input koordinat (100,300), (200,300), (300,300) setelah di uji menghasilkan nilai error 0% . Sebagai contoh pada pengujian input (100,300) data komunikasi yang diperoleh antara robot dengan visualisasi yakni sama. Koordinat x menempuh 0.00, 87.80, 99.36 sedangkan y menempuh 300.00, 300.47, 300.16.



Pada tabel 4.5 berikut adalah hasil uji komunikasi antara robot 1 ke visualisasi dengan gerak robot serong.

Tabel 4. 5 Uji Komunikasi Antara Robot 1 Ke Visualisasi Dengan Gerak Robot Serong

No.	Input Data Koordinat (cm)		Data Yang Dikirim Arduino (cm)		Data Yang Diterima Aplikasi (cm)		Error (cm)	
	X	Y	X	Y	X	Y	X	Y
1	150	300	0,00	0,00	0,00	0,00	0	0
			26,28	27,64	26,28	27,64		
			71,86	74,56	71,86	74,56		
			117,66	120,60	117,66	120,60		
			150,12	167,06	150,12	167,06		
			150,18	207,63	150,18	207,63		
			150,06	247,38	150,06	247,38		
			150,12	286,72	150,12	286,72		
			149,94	300,42	149,94	300,42		
2	150	300	0,00	0,00	0,00	0,00	0	0
			4,47	5,06	4,47	5,06		
			49,16	52,39	49,16	52,39		
			94,79	98,32	94,79	98,32		
			141,12	143,77	141,12	143,77		
			150,06	185,28	150,06	185,28		
			150,12	224,50	150,12	224,50		
			150,06	263,43	150,06	263,43		
			150,12	300,71	150,12	300,71		
150,06	300,42	150,06	300,42					
3	150	300	0,00	0,00	0,00	0,00	0	0
			43,10	44,28	43,10	44,28		
			88,97	91,97	88,97	91,97		

			134,83	138,07	134,83	138,07		
			150,12	182,58	150,12	182,58		
			150,12	222,86	150,12	222,86		
			150,12	264,55	150,12	264,55		
			150,00	300,59	150,00	300,59		
			150,00	300,53	150,00	300,53		
4	150	300	0,00	0,00	0,00	0,00	0	0
			16,52	17,99	16,52	17,99		
			78,85	87,09	78,85	87,09		
			141,01	156,18	141,01	156,18		
			149,12	224,03	149,12	224,03		
			149,00	292,01	149,00	292,01		
			149,89	300,24	149,89	300,24		
5	150	300	0,00	0,00	0,00	0,00	0	0
			4,23	4,70	4,23	4,70		
			66,33	73,21	66,33	73,21		
			128,30	142,83	128,30	142,83		
			148,65	211,80	148,65	211,80		
			149,89	279,54	149,89	279,54		
			149,71	299,89	149,71	299,89		
6	450	300	0,00	0,00	0,00	0,00	0	0
			25,34	23,76	25,34	23,76		
			68,74	67,03	68,74	67,03		
			111,43	109,37	111,43	109,37		
			156,18	152,71	156,18	152,71		
			200,34	195,63	200,34	195,63		
			245,50	239,20	245,50	239,20		
			290,42	282,89	290,42	282,89		
			330,93	300,18	330,93	300,18		
			365,69	300,24	365,69	300,24		
			400,20	300,30	400,20	300,30		
			434,48	300,24	434,48	300,24		

			450,30	300,18	450,30	300,18		
7	450	300	0,00	0,00	0,00	0,00	0	0
			14,11	14,82	14,11	14,82		
			56,80	56,68	56,80	56,68		
			99,55	98,61	99,55	98,61		
			143,01	140,83	143,01	140,83		
			185,87	184,64	185,87	184,64		
			230,21	226,80	230,21	226,80		
			275,01	270,19	275,01	270,19		
			317,53	300,24	317,53	300,24		
			352,99	300,36	352,99	300,36		
			387,97	300,36	387,97	300,36		
			422,84	300,36	422,84	300,36		
			450,01	300,18	450,01	300,18		
8	450	300	0,00	0,00	0,00	0,00	0	0
			3,88	4,00	3,88	4,00		
			44,92	45,04	44,92	45,04		
			87,61	86,38	87,61	86,38		
			131,01	127,89	131,01	127,89		
			174,29	169,82	174,29	169,82		
			217,98	212,51	217,98	212,51		
			262,43	254,61	262,43	254,61		
			307,06	297,89	307,06	297,89		
			343,11	300,36	343,11	300,36		
			376,98	300,48	376,98	300,48		
			410,02	300,42	410,02	300,42		
			442,30	300,36	442,30	300,36		
			449,89	300,42	449,89	300,42		
9	450	300	0,00	0,00	0,00	0,00	0	0
			0,18	0,06	0,18	0,06		
			30,69	30,34	30,69	30,34		
			72,38	71,21	72,38	71,21		

			115,72	112,60	115,72	112,60		
			158,53	154,06	158,53	154,06		
			202,51	196,22	202,51	196,22		
			246,32	237,85	246,32	237,85		
			290,60	279,90	290,60	279,90		
			330,93	300,36	330,93	300,36		
			363,86	300,36	363,86	300,36		
			396,50	300,30	396,50	300,30		
			428,96	300,36	428,96	300,36		
			449,71	300,12	449,71	300,12		
10	450	300	0,00	0,00	0,00	0,00	0	0
			23,17	21,93	23,17	21,93		
			65,68	63,45	65,68	63,45		
			108,19	104,55	108,19	104,55		
			152,24	146,77	152,24	146,77		
			195,40	188,58	195,40	188,58		
			238,97	230,50	238,97	230,50		
			283,01	272,96	283,01	272,96		
			324,70	300,24	324,70	300,24		
			358,10	300,24	358,10	300,24		
			391,50	300,24	391,50	300,24		
			424,43	300,36	424,43	300,36		
			449,95	300,42	449,95	300,42		
11	750	300	0,00	0,00	0,00	0,00	0	0
			7,17	7,88	7,17	7,88		
			69,62	76,44	69,62	76,44		
			131,66	145,89	131,66	145,89		
			193,69	215,21	193,69	215,21		
			255,96	284,42	255,96	284,42		
			326,41	300,12	326,41	300,12		
			398,38	300,12	398,38	300,12		
			470,00	300,12	470,00	300,12		

			541,62	300,12	541,62	300,12		
			613,24	300,12	613,24	300,12		
			684,86	300,12	684,86	300,12		
			751,19	299,95	751,19	299,95		
			750,01	299,95	750,01	299,95		
12	750	300	0,00	0,00	0,00	0,00	0	0
			4,29	4,65	4,29	4,65		
			66,68	73,09	66,68	73,09		
			128,95	142,12	128,95	142,12		
			191,05	211,45	191,05	211,45		
			253,14	280,72	253,14	280,72		
			323,47	300,18	323,47	300,18		
			396,09	300,12	396,09	300,12		
			468,71	300,12	468,71	300,12		
			541,39	300,18	541,39	300,18		
			614,06	300,18	614,06	300,18		
			685,86	300,12	685,86	300,12		
			750,31	300,01	750,31	300,01		
			749,90	300,06	749,90	300,06		
13	750	300	0,00	0,00	0,00	0,00	0	0
			33,81	35,16	33,81	35,16		
			78,62	82,32	78,62	82,32		
			124,31	130,54	124,31	130,54		
			170,41	179,11	170,41	179,11		
			217,15	226,97	217,15	226,97		
			263,37	275,66	263,37	275,66		
			308,00	300,06	308,00	300,06		
			348,46	300,12	348,46	300,12		
			388,80	300,12	388,80	300,12		
			427,84	300,12	427,84	300,12		
			467,47	300,18	467,47	300,18		
			507,34	300,12	507,34	300,12		

			546,74	300,12	546,74	300,12		
			586,66	300,12	586,66	300,12		
			625,94	300,18	625,94	300,18		
			664,87	300,12	664,87	300,12		
			704,03	300,12	704,03	300,12		
			745,07	300,06	745,07	300,06		
			750,42	300,01	750,42	300,01		
14	750	300	0,00	0,00	0,00	0,00	0	0
			32,63	35,52	32,63	35,52		
			77,50	83,79	77,50	83,79		
			122,48	131,60	122,48	131,60		
			168,23	179,99	168,23	179,99		
			213,86	228,09	213,86	228,09		
			260,49	276,84	260,49	276,84		
			306,65	299,95	306,65	299,95		
			347,75	299,95	347,75	299,95		
			389,03	299,95	389,03	299,95		
			430,37	299,89	430,37	299,89		
			471,41	299,95	471,41	299,95		
			513,16	299,95	513,16	299,95		
			554,44	299,95	554,44	299,95		
			595,54	300,01	595,54	300,01		
			636,47	299,95	636,47	299,95		
			677,75	299,89	677,75	299,89		
			719,20	299,89	719,20	299,89		
			750,37	299,83	750,37	299,83		
			750,42	300,01	750,42	300,01		
15	750	300	0,00	0,00	0,00	0,00	0	0
			12,76	13,47	12,76	13,47		
			58,21	60,62	58,21	60,62		
			104,67	108,14	104,67	108,14		
			151,18	156,29	151,18	156,29		

			198,40	204,34	198,40	204,34		
			245,44	252,20	245,44	252,20		
			293,54	299,65	293,54	299,65		
			336,99	299,83	336,99	299,83		
			378,33	299,83	378,33	299,83		
			417,43	299,83	417,43	299,83		
			455,65	299,83	455,65	299,83		
			494,23	299,83	494,23	299,83		
			533,39	299,77	533,39	299,77		
			572,02	299,83	572,02	299,83		
			610,42	299,83	610,42	299,83		
			648,58	299,89	648,58	299,89		
			687,74	299,83	687,74	299,83		
			726,43	299,77	726,43	299,77		
			750,37	299,83	750,37	299,83		
Rata-rata							0	0

Berdasarkan tabel 4.5 ditunjukkan bahwa nilai input koordinat (150,300), (450,300), (750,300) setelah di uji menghasilkan nilai error 0% . Sebagai contoh pada pengujian input (150,300) data komunikasi yang diperoleh antara robot dengan visualisasi yakni sama. Koordinat x menempuh 0.00, 26.28, 71.86, 117.66, 150.12, 150.18, 150.06, 150.12, 149.94 sedangkan y menempuh 0.00, 27.64, 74.56, 120.60, 167.06, 207.63, 247.38, 286.72, 300.42.

Pada tabel 4.6 berikut adalah hasil uji komunikasi antara robot 2 ke visualisasi dengan gerak robot secara serong.

Tabel 4. 6 Uji Komunikasi Antara Robot 2 Ke Visualisasi Dengan Gerak Robot Serong

No.	Input Data Koordinat (cm)		Data Yang Dikirim Arduino (cm)		Data Yang Diterima Aplikasi (cm)		Error (cm)	
	X	Y	X	Y	X	Y	X	Y
1	100	400	0,00	300,00	0,00	300,00	0	0
			97,64	392,73	97,64	392,73		
			99,67	400,86	99,67	400,86		
2	100	400	0,00	300,00	0,00	300,00	0	0
			98,57	401,64	98,57	401,64		
3	100	400	0,00	300,00	0,00	300,00	0	0
			99,98	401,80	99,98	401,80		
4	100	400	0,00	300,00	0,00	300,00	0	0
			99,98	401,64	99,98	401,64		
5	100	400	0,00	300,00	0,00	300,00	0	0
			25,00	322,75	25,00	322,75		
			99,20	401,48	99,20	401,48		
6	200	500	0,00	300,00	0,00	300,00	0	0
			132,94	419,45	132,94	419,45		
			201,05	501,78	201,05	501,78		
7	200	500	0,00	300,00	0,00	300,00	0	0
			157,47	442,26	157,47	442,26		
			200,59	501,31	200,59	501,31		
8	200	500	0,00	300,00	0,00	300,00	0	0
			58,74	356,02	58,74	356,02		
			202,77	503,81	202,77	503,81		
			199,02	501,78	199,02	501,78		

9	200	500	0,00 10,62 198,87	300,00 312,28 501,78	0,00 10,62 198,87	300,00 312,28 501,78	0	0
10	200	500	0,00 91,08 198,87	300,00 386,33 501,93	0,00 91,08 198,87	300,00 386,33 501,93	0	0
11	450	600	0,00 17,97 182,15 348,84 451,32	300,00 319,47 468,03 603,63 601,44	0,00 17,97 182,15 348,84 451,32	300,00 319,47 468,03 603,63 601,44	0	0
12	450	600	0,00 10,47 173,72 399,15 451,63	300,00 312,59 461,47 604,10 601,44	0,00 10,47 173,72 399,15 451,63	300,00 312,59 461,47 604,10 601,44	0	0
13	450	600	0,00 10,78 174,65 340,24 451,32	300,00 309,94 458,66 605,66 600,66	0,00 10,78 174,65 340,24 451,32	300,00 309,94 458,66 605,66 600,66	0	0
14	450	600	0,00 42,49 205,90 376,18 451,94	300,00 342,28 492,09 600,51 601,44	0,00 42,49 205,90 376,18 451,94	300,00 342,28 492,09 600,51 601,44	0	0
15	450	600	0,00 33,90 197,77 366,65 451,63	300,00 334,31 481,94 601,44 600,97	0,00 33,90 197,77 366,65 451,63	300,00 334,31 481,94 601,44 600,97	0	0
Rata-rata							0	0

Berdasarkan tabel 4.6 ditunjukkan bahwa nilai input koordinat (100,400), (200,500), (450,600) setelah di uji menghasilkan nilai error 0% . Sebagai contoh pada pengujian input (100,400) data komunikasi yang diperoleh antara robot dengan visualisasi yakni sama. Koordinat x menempuh 0.00, 97.64, 99.67 sedangkan y menempuh 300.00, 392.73, 400.86.



#### 4.1.4 Hasil Akhir Uji Komunikasi Antara Robot Dengan Visualisasi

Pada tabel 4.7 berikut adalah hasil akhir uji komunikasi antara robot dengan visualisasi.

Tabel 4. 7 Hasil Akhir Uji Komunikasi Antara Robot Dengan Visualisasi

Robot	Error Ke samping		Error Maju		Error Serong	
	X	Y	X	Y	X	Y
1	0	0	0	0	0	0
2	0	0	0	0	0	0

#### 4.1.5 Kesimpulan Dari Semua Hasil Uji Komunikasi Antara Robot Dengan Visualisasi

Berdasarkan dari semua hasil uji coba yang dilakukan pada hasil uji komunikasi antara robot 1 dan 2 dengan visualisasi terbukti bahwa data yang dikirim dari robot lalu diterima oleh visualisasi tidak ada error sama sekali (error = 0%).

## **4.2 Uji Ketepatan Koordinat Robot Sepak Bola Beroda Dengan Visualisasi Posisi Robot**

Pengujian ini berfungsi untuk mengetahui ketepatan koordinat yang diterima oleh visualisasi terhadap gambar visual robot. Nilai umpan balik dari proses odometry yang dikirim ke visualisasi akan diseleksi jika kurang dari 0,50 maka akan mengikuti pembulatan bawah, jika lebih dari 0,49 maka akan ikut pembulatan atas.

### **4.2.1 Alat Yang Digunakan Pada Uji Ketepatan Koordinat Robot Sepak Bola Beroda Dengan Visualisasi Posisi Robot**

Peralatan yang diperlukan untuk pengujian ini adalah sebagai berikut :

1. Robot Sepak Bola
2. Laptop
3. Aplikasi Visualisasi

### **4.2.2 Prosedur Pengujian Pada Uji Ketepatan Koordinat Robot Sepak Bola Beroda Dengan Visualisasi Posisi Robot**

Langkah-langkah yang dilakukan dalam melakukan pengujian ini adalah sebagai berikut :

1. Mempersiapkan pc untuk melihat pengiriman data dari serial monitor pada arduino.
2. Mempersiapkan aplikasi visualisasi pada pc untuk kontrol robot.
3. Mempersiapkan robot pada posisi awal.
4. Mengkoneksikan antara aplikasi visualisasi dengan robot.

5. Memposisikan robot 1 dan 2 pada posisi awal.
6. Memberikan dan mengirim input koordinat dari aplikasi visualisasi.
7. Mengamati pengiriman data hasil dari koordinat robot yang dikirim ke aplikasi visualisasi.
8. Megamati hasil koordinat robot dan menyesuaikan dengan gambar visual robot. Jika terdapat nilai kurang dari 0,50 maka gambar akan mengikuti pembulatan bawahnya yaitu 0, jika terdapat nilai lebih dari 0,49 maka gambar akan mengikuti pembulatan atasnya yaitu 1, jika tidak terjadi pembulatan pada visualisasi maka terdapat error.



### 4.2.3 Hasil Uji Ketepatan Koordinat Robot Sepak Bola Beroda Dengan Visualisasi Posisi Robot

Pada tabel 4.8 berikut adalah hasil ketepatan gerak robot 1 ke samping kanan pada visualisasi.

Tabel 4. 8 Uji Ketepatan Gerak Robot 1 Ke Samping Kanan Pada Visualisasi

No	Input Dari Visualisasi		Input Data Dari Robot		Output Data Dari Robot		Output Dari Visualisasi		Error	
	X	Y	X	Y	X	Y	X	Y	X	Y
1	0	100	0	100	0,06	100,32	0	100	0	100
2	0	100	0	100	0,06	99,84	0	100	0	100
3	0	100	0	100	0,00	99,67	0	100	0	100
4	0	100	0	100	0,00	99,90	0	100	0	100
5	0	100	0	100	0,00	100,02	0	100	0	100
6	0	200	0	200	0,00	199,69	0	200	0	200
7	0	200	0	200	0,00	199,69	0	200	0	200
8	0	200	0	200	0,00	199,75	0	200	0	200
9	0	200	0	200	0,06	200,10	0	200	0	200
10	0	200	0	200	0,00	199,81	0	200	0	200
11	0	300	0	300	0,00	299,83	0	300	0	300
12	0	300	0	300	0,00	300,24	0	300	0	300
13	0	300	0	300	0,06	299,65	0	300	0	300
14	0	300	0	300	0,00	299,89	0	300	0	300
15	0	300	0	300	0,00	299,95	0	300	0	300
Rata-rata									0	0

Berdasarkan tabel 4.8 ditunjukkan bahwa nilai input koordinat (0,100), (0,200), (0,300) setelah di uji menghasilkan nilai error 0% . Sebagai contoh pada pengujian input (0,100) setelah pergerakan dari posisi awalnya robot sepak bola beroda mengirimkan posisinya berada pada koordinat x (0,06) dan y (100,32). Setelah

melalui proses pembulatan pada visualisasi maka gambar robot berada pada posisi koordinat x (0) dan y (100).

Pada tabel 4.9 berikut adalah hasil ketepatan gerak robot 2 ke samping kanan pada visualisasi.

Tabel 4. 9 Uji Ketepatan Gerak Robot 2 Ke Samping Kanan Pada Visualisasi

No	Input Dari Visualisasi		Input Data Dari Robot		Output Data Dari Robot		Output Dari Visualisasi		Error	
	X	Y	X	Y	X	Y	X	Y	X	Y
1	0	400	0	400	0,00	402,26	0	402	0	0
2	0	400	0	400	0,00	402,58	0	403	0	0
3	0	400	0	400	0,16	402,26	0	402	0	0
4	0	400	0	400	0,00	402,58	0	403	0	0
5	0	400	0	400	0,00	402,26	0	402	0	0
6	0	500	0	500	0,00	502,56	0	503	0	0
7	0	500	0	500	0,00	502,09	0	502	0	0
8	0	500	0	500	0,00	502,24	0	502	0	0
9	0	500	0	500	0,00	502,71	0	503	0	0
10	0	500	0	500	0,00	502,56	0	503	0	0
11	0	600	0	600	0,00	602,69	0	603	0	0
12	0	600	0	600	0,00	602,54	0	603	0	0
13	0	600	0	600	0,31	602,07	0	602	0	0
14	0	600	0	600	0,00	602,54	0	603	0	0
15	0	600	0	600	0,00	601,60	0	602	0	0
Rata-rata									0	0

Berdasarkan tabel 4.9 ditunjukkan bahwa nilai input koordinat (0,400), (0,500), (0,600) setelah di uji menghasilkan nilai error 0% . Sebagai contoh pada pengujian input (0,400) setelah pergerakan dari posisi awalnya robot sepak bola beroda

mengirimkan posisinya berada pada koordinat x (0,0) dan y (402,26). Setelah melalui proses pembulatan pada visualisasi maka gambar robot berada pada posisi koordinat x (0) dan y (400).

Pada tabel 4.10 berikut adalah hasil ketepatan gerak robot 1 maju pada visualisasi.

Tabel 4. 10 Uji Ketepatan Gerak Robot 1 Maju Pada Visualisasi Posisi Robot

No	Input Dari Visualisasi		Input Data Dari Robot		Output Data Dari Robot		Output Dari Visualisasi		Error	
	X	Y	X	Y	X	Y	X	Y	X	Y
1	100	0	100	0	100,08	0,00	100	0	0	0
2	100	0	100	0	99,84	0,06	100	0	0	0
3	100	0	100	0	99,67	0,00	100	0	0	0
4	100	0	100	0	99,73	0,06	100	0	0	0
5	100	0	100	0	99,79	0,00	100	0	0	0
6	200	0	200	0	199,69	0,00	200	0	0	0
7	200	0	200	0	199,69	0,00	200	0	0	0
8	200	0	200	0	199,75	0,00	200	0	0	0
9	200	0	200	0	199,63	0,00	200	0	0	0
10	200	0	200	0	199,98	0,00	200	0	0	0
11	300	0	300	0	299,83	0,00	300	0	0	0
12	300	0	300	0	300,24	0,00	300	0	0	0
13	300	0	300	0	299,65	-0,06	300	0	0	0
14	300	0	300	0	299,65	0,00	300	0	0	0
15	300	0	300	0	299,71	0,00	300	0	0	0
Rata-rata									0	0

Berdasarkan tabel 4.10 ditunjukkan bahwa nilai input koordinat (100,0), (200,0), (300,0) setelah di uji menghasilkan nilai error 0% . Sebagai contoh pada pengujian input (100,0) setelah pergerakan dari posisi awalnya robot sepak bola berada

mengirimkan posisinya berada pada koordinat x (100,08) dan y (0,00). Setelah melalui proses pembulatan pada visualisasi maka gambar robot berada pada posisi koordinat x (100) dan y (0).

Pada tabel 4.11 berikut adalah hasil ketepatan gerak robot 2 maju pada visualisasi.

Tabel 4. 11 Uji Ketepatan Gerak Robot 2 Maju Pada Visualisasi

No	Input Dari Visualisasi		Input Data Dari Robot		Output Data Dari Robot		Output Dari Visualisasi		Error	
	X	Y	X	Y	X	Y	X	Y	X	Y
1	100	300	100	300	99,36	300,16	99	300	0	0
2	100	300	100	300	99,20	300,00	99	300	0	0
3	100	300	100	300	99,51	300,00	100	300	0	0
4	100	300	100	300	99,51	300,00	100	300	0	0
5	100	300	100	300	100,14	300,00	100	300	0	0
6	200	300	200	300	198,87	300,00	199	300	0	0
7	200	300	200	300	199,02	300,00	199	300	0	0
8	200	300	200	300	199,34	300,00	199	300	0	0
9	200	300	200	300	199,34	300,00	199	300	0	0
10	200	300	200	300	198,87	300,00	199	300	0	0
11	300	300	300	300	299,32	300,00	299	300	0	0
12	300	300	300	300	299,32	300,00	299	300	0	0
13	300	300	300	300	299,00	300,31	299	300	0	0
14	300	300	300	300	299,00	300,16	299	300	0	0
15	300	300	300	300	299,78	300,00	300	300	0	0
Rata-rata									0	0

Berdasarkan tabel 4.11 ditunjukkan bahwa nilai input koordinat (100,300), (200,300), (300,300) setelah di uji menghasilkan nilai error 0% . Sebagai contoh pada pengujian input (100,300) setelah pergerakan dari posisi awalnya robot sepak bola beroda mengirimkan posisinya berada pada koordinat x (99,36) dan y (300,16).

Setelah melalui proses pembulatan pada visualisasi maka gambar robot berada pada posisi koordinat x (99) dan y (300).

Pada tabel 4.12 berikut adalah hasil ketepatan gerak robot 1 serong pada visualisasi.

Tabel 4. 12 Uji Ketepatan Gerak Robot 1 Serong Pada Visualisasi

No	Input Dari Visualisasi		Input Data Dari Robot		Output Data Dari Robot		Output Dari Visualisasi		Error	
	X	Y	X	Y	X	Y	X	Y	X	Y
1	100	200	100	200	99,90	200,28	100	200	0	0
2	100	200	100	200	99,90	200,69	100	201	0	0
3	100	200	100	200	99,79	199,81	100	200	0	0
4	100	200	100	200	99,90	199,75	100	200	0	0
5	100	200	100	200	100,08	199,75	100	200	0	0
6	200	300	200	300	200,10	300,30	200	300	0	0
7	200	300	200	300	200,10	275,96	200	276	0	0
8	200	300	200	300	200,04	299,83	200	300	0	0
9	200	300	200	300	200,04	299,89	200	300	0	0
10	200	300	200	300	199,81	300,12	200	300	0	0
11	450	300	450	300	450,13	299,89	450	300	0	0
12	450	300	450	300	449,66	299,89	450	300	0	0
13	450	300	450	300	449,95	299,89	450	300	0	0
14	450	300	450	300	449,77	299,89	450	300	0	0
15	450	300	450	300	450,13	299,89	450	300	0	0
Rata-rata									0	0

Berdasarkan tabel 4.12 ditunjukkan bahwa nilai input koordinat (100,200), (200,300), (450,300) setelah di uji menghasilkan nilai error 0% . Sebagai contoh pada pengujian input (100,200) setelah pergerakan dari posisi awalnya robot sepak bola beroda mengirimkan posisinya berada pada koordinat x (99,90) dan y (200,28).

Setelah melalui proses pembulatan pada visualisasi maka gambar robot berada pada posisi koordinat x (100) dan y (200).

Pada tabel 4.13 berikut adalah hasil ketepatan gerak robot 2 serong pada visualisasi.

Tabel 4. 13 Uji Ketepatan Gerak Robot 2 Serong Pada Visualisasi

No	Input Dari Visualisasi		Input Data Dari Robot		Output Data Dari Robot		Output Dari Visualisasi		Error	
	X	Y	X	Y	X	Y	X	Y	X	Y
1	100	400	100	400	99,67	400,86	100	401	0	0
2	100	400	100	400	98,57	401,64	99	402	0	0
3	100	400	100	400	99,98	401,80	100	402	0	0
4	100	400	100	400	99,98	401,64	100	402	0	0
5	100	400	100	400	99,20	401,48	99	401	0	0
6	200	500	200	500	201,05	501,78	201	502	0	0
7	200	500	200	500	200,59	501,31	201	501	0	0
8	200	500	200	500	199,02	501,78	199	502	0	0
9	200	500	200	500	198,87	501,78	199	502	0	0
10	200	500	200	500	198,87	501,93	199	502	0	0
11	450	600	450	600	451,32	601,44	451	601	0	0
12	450	600	450	600	451,63	601,44	452	601	0	0
13	450	600	450	600	451,32	600,66	451	601	0	0
14	450	600	450	600	451,94	601,44	452	601	0	0
15	450	600	450	600	451,63	600,97	452	601	0	0
Rata-rata									0	0

Berdasarkan tabel 4.13 ditunjukkan bahwa nilai input koordinat (100,400), (200,500), (450,600) setelah di uji menghasilkan nilai error 0% . Sebagai contoh pada pengujian input (100,400) setelah pergerakan dari posisi awalnya robot sepak bola beroda mengirimkan posisinya berada pada koordinat x (99,67) dan y (400,86).

Setelah melalui proses pembulatan pada visualisasi maka gambar robot berada pada posisi koordinat x (100) dan y (400).

#### 4.2.4 Hasil Akhir Uji Ketepatan Koordinat Robot Sepak Bola Beroda Dengan Visualisasi Posisi Robot

Pada tabel 4.14 berikut adalah hasil akhir ketepatan gerak robot 2 serong pada visualisasi.

Tabel 4. 14 Hasil Uji Ketepatan Koordinat Robot Sepak Bola Beroda Dengan Visualisasi

Robot	Error Ke samping		Error Maju		Error Serong	
	X	Y	X	Y	X	Y
1	0	0	0	0	0	0
2	0	0	0	0	0	0

#### 4.2.5 Kesimpulan Dari Semua Hasil Uji Ketepatan Koordinat Robot Dengan Visualisasi

Dari semua hasil uji ketepatan koordinat gambar robot 1 dan 2 pada visualisasi tidak terdapat error dalam pembulatan koordinat dari robot yang terkirim ke visualisasi (error = 0%).

## BAB V

### PENUTUP

Dalam bab ini berisi tentang kesimpulan dan saran berdasarkan dari hasil pengujian yang telah dilakukan pada Tugas Akhir ini. Dalam pengerjaan Tugas Akhir ini diperoleh kesimpulan dan saran agar bisa mengembangkan sistem lebih baik lagi kedepannya.

#### 5.1 Kesimpulan

Berdasarkan pengujian pada Tugas Akhir diperoleh kesimpulan dari beberapa poin sebagai berikut :

1. Dalam merancang visualisasi posisi robot terdapat robot dan lapangan robot sepak bola beroda. Agar sesuai keadaan di lapangan robot sepak bola berada maka diperlukan skala untuk menunjukkan perbandingan jarak pada perancangan visualisasi posisi robot dengan keadaan sebenarnya. Disini penulis menggunakan ketentuan  $1 \text{ cm} = 1 \text{ pixel}$  dan hasilnya baik.
2. Untuk merekam gerak robot sesuai lokasi yang sebenarnya diperlukan menangkap data hasil rotary pergerakan dari robot lalu mengolah data tersebut supaya bisa membuat gambar visual robot di visualisasi bisa bergerak mengikuti pergerakan robot sebenarnya.
3. Hasil yang diperoleh dari proses uji komunikasi antara robot dengan visualisasi terbukti bahwa data yang dikirim robot ke aplikasi visualisasi selalu sesuai atau tidak ada (error = 0%).

4. Hasil koordinat yang dikirim dari robot tidak selalu bernilai angka bulat, maka dari itu harus dibulatkan oleh sistem agar dapat diterima oleh visualisasi posisi robot. Pembulatan dilakukan setelah proses penerimaan data oleh aplikasi visualisasi, dimana jika kurang dari 0,50 maka akan ikut pembulatan bawah, jika lebih dari 0,49 akan ikut pembulatan atas.

## 5.2 Saran

Saran dari penulis pada pengembangan Tugas Akhir ini agar kedepannya dapat dikembangkan lebih baik lagi adalah sebagai berikut :

1. Pemetaan robot terhadap garis tepi dari agar robot bisa dikendalikan tidak keluar dari garis tepi lapangan.
2. Robot dapat mengetahui daerah letak gawang lawan melalui visualisasi posisi robot.
3. Robot dapat mengetahui daerah sendiri dan daerah lawan melalui visualisasi posisi robot.
4. Terdapat indikator di aplikasi visualisasi bahwa robot dapat mengetahui jika mendapat bola.
5. Saat sistem perancangan ini digabungkan dapat meringankan beban *base station* agar tidak terjadi tabrakan antar pengiriman data saat komunikasi dua arah berlangsung.

## DAFTAR PUSTAKA

- Abseno, A. P. (2019). *Penerapan Kinematika Untuk Lokalisasi Pada Robot Sepak Bola Beroda*. Surabaya: Institut Bisnis dan Informatika Stikom Surabaya.
- Amin, A. M. (2019). Perancangan Sistem Komunikasi Data Robot Sepak Bola Dalam Kontes Robot Sepak Bola Indonesia Beroda (KRSBI). *Artikel Skripsi*, 2.
- Ardhiansyah, T., & Syarifuddin, I. (2017). Pergerakan Otomatis Robot Sepak Bola Beroda Melalui Komunikasi dengan Refree Box Menggunakan Base Station. *5th Indonesian Symposium on Robotic System and Control*, 5.
- Due, A. (2017). *ARDUINO DUE*. Retrieved from <https://store.arduino.cc/usa/arduino-due>
- Indoware. (2019, Juni 2019). *indo-ware*. Retrieved from [www.indo-ware.com](http://www.indo-ware.com): <https://www.indo-ware.com/produk-3771-arduino-due-r3-sam3x8e-32bit-arm-cortexm-compatible-board-module.html>
- Kementrian Riset, T. d. (2017). *Buku Panduan KRSBI Beroda 2018*. Retrieved from <http://kontesrobotindonesia.id/tentang-kri.html>
- KRSBI-B, P. (2017). *Buku Panduan Kontes Robot Sepak Bola Indonesia Devisi (KRSBI BERODA) 2018*. Jakarta: Kementerian Riset, Teknologi, dan Pendidikan Tinggi Republik Indonesia.
- Wahyuni, E. D. (2015). *Rancang Bangun Program Visualisasi Pergerakan*. Surabaya: Politeknik Elektronika Negeri Surabaya.
- Widiyaman, T. (2018, July 17). *Macam Macam Jenis Keluarga ESP8266*. Diambil kembali dari warriornux: <https://www.warriornux.com/macam-macam-jenis-keluarga-esp8266/>

Wikipedia. (2018, Desember 5). *Suit protokol internet*. Retrieved from  
id.Wikipedia.org: [https://id.wikipedia.org/wiki/Suit\\_protokol\\_internet](https://id.wikipedia.org/wiki/Suit_protokol_internet)

