

PENGEMBANGAN KOMUNIKASI DAN VISUALISASI *BASE STATION* PADA ROBOT SEPAK BOLA BERODA

Moch. Angga Putra Wicaksono ¹⁾ Susijanto Tri Rasmana ²⁾ Ira Puspasari ³⁾

Program Studi/Jurusan Teknik Komputer

Universitas Dinamika.

Jl.Raya Kedung Baruk 98 Surabaya, 60298

Email: 1)anggaputra182@gmail.com, 2) susyanto@dinamika.ac.id, 3)ira@dinamika.ac.id

Abstrak: Pengendalian robot secara manual membuat pergerakan robot KRSBI menjadi tidak fleksibel dan tidak relevan seiring berjalannya waktu. Dengan penelitian ini bisa menjawab dari permasalahan tersebut. Robot dapat langsung mengetahui perintah dari *RefereeBox*, aplikasi juga mampu memberikan informasi status robot membawa bola yang nantinya dapat digunakan sebagai strategi bagi robot. Selain itu visualisasi dibuat sangat *smooth* sejalan dengan jalan robot yang sebenarnya. Pada penelitian ini penulis mendapatkan hasil yang diperoleh dari proses uji validasi komunikasi antara robot dengan aplikasi *base station*, terbukti data yang dikirimkan dan diterima memiliki kesamaan dan tingkat uji coba tersebut adalah 100%, hasil yang diperoleh dari hasil uji kesamaan pergerakan robot sebenarnya dengan pergerakan robot pada aplikasi *base station* dengan tingkat rata-rata *error* yang tidak besar dan dapat ditoleransi dengan ketentuan 1 pixel = 2 cm, dan hasil uji kecepatan pengiriman data robot ke aplikasi *base station* juga menyimpulkan bahwa *range* periode pengiriman terbaik adalah dari 33 milidetik sampai 200 milidetik. Dapat disimpulkan aplikasi *base station* yang dibuat ini dapat digunakan untuk perlombaan di ajang KRSBI.

Kata kunci: *basestation*, *refereebox*, robot, KRI, KRSBI.

PENDAHULUAN

Kontes Robot Indonesia (KRI) adalah suatu event kontes robot bagi perguruan tinggi di Indonesia yang dapat diikuti oleh mahasiswa yang rutin dilaksanakan setiap tahun sekali. KRI diselenggarakan oleh Kementerian Riset, Teknologi dan Pendidikan Tinggi untuk tujuan meningkatkan bakat dan kemampuan mahasiswa dalam bidang robotika. Salah satu divisi dari Kontes Robot Indonesia (KRI) adalah Kontes Robot Sepak Bola Indonesia Beroda (KRSBI), mahasiswa dituntut untuk mengembangkan kemampuan pada bidang elektronika, *image processing*, komunikasi, strategi, mekanik, artificial intelligence (AI), manufaktur, pemrograman, dan sekaligus pengembangan kedisiplinan, sportifitas, toleransi, kerjasama dalam tim, saling menghargai, kontrol emosi dan kemampuan *softskill* lainnya (Kementerian Riset, 2019).

Pada saat pertandingan, robot dikontrol dan diperintah berdasarkan instruksi wasit melalui perangkat lunak yang disebut *Refbox* (*Referee Box*). *Refbox* merupakan game *controller* salah satu

teknologi yang digunakan untuk mempermudah wasit dalam mengatur jalannya pertandingan KRSBI. Fungsi dari *Refbox* adalah untuk menyampaikan keputusan wasit kepada robot yang sedang bermain, mencatat skor, dan sebagai penampilan waktu pertandingan. Protokol komunikasi *Refbox* mengirimkan beberapa pesan ke komputer kedua tim (magenta dan cyan) (Ardhiansyah & Syarifuddin, 2017).

Universitas Dinamika yang dahulu masih memakai nama Institut Bisnis Dan Informatika Stikom Surabaya telah mengikuti tiga kali kontes tersebut yaitu pada tahun 2017, 2018, dan 2019. Pada tahun pertama robot sepak bola tim hanya mampu mendeteksi warna bola dan juga menendang bola. Hal ini masih jauh belum memenuhi syarat kriteria dari *rule competition* dan tim belum mampu membuat aplikasi *Base Station* untuk dapat berkomunikasi dengan baik antar robot maupun menerima perintah dari *Refree Box* (kontrol wasit).

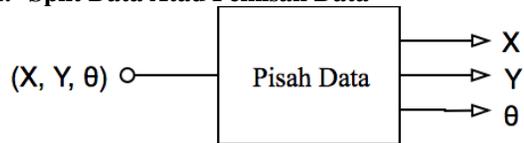
Pada tahun kedua robot sepak bola Universitas Dinamika mengalami kemajuan dalam hal mendeteksi warna bola lebih akurat,

menentukan arah serta dapat membuat aplikasi *Base Station* untuk komunikasi antar robot dan menerima perintah dari *Refree Box* ke *Base Station*, adanya aplikasi *Base Station* ini mempermudah tim dalam pengontrolan robot dari laptop pada saat itu hanya dapat maju, mundur, rotasi kanan dan rotasi kiri.

Pada tahun ketiga tim terdapat kemajuan dalam hal menentukan arah lawan lebih akurat, base robot yang lebih presisi, penandang menggunakan motor dan dapat mengontrol robot sesuai dengan koordinat dalam skala lapangan yang sebenarnya. Tetapi terdapat beberapa kelemahan dari sisi komunikasi. Sampai saat ini, untuk melakukan perintah *start*, *stop*, dan lain-lain yang seharusnya diberikan oleh *Refree Box* masih dilakukan secara manual menggunakan tombol yang terdapat pada *Base Station*, sebagai contoh ketika wasit memberi perintah start seketika itu *Base Station* menerima perintah dan melanjutkan perintah ke robot dengan menekan tombol start secara manual. Seharusnya tidak boleh terjadi karena melanggar aturan dari Dikti sebagaimana yang telah diatur di rule tersebut seharusnya *Base Station* dapat meneruskan perintah dari *Refree Box* milik wasit ke robot secara langsung tanpa harus menekan tombol secara manual. Kelemahan yang lain adalah belum mampu mengetahui pergerakan robot secara lebih detail di aplikasi *Base Station*.

Dari penelitian yang ada yang berjudul “Perancangan Visualisasi Posisi Robot Pada Pengendali Robot Sepak Bola Beroda” yang dikerjakan oleh Vicky Indrarta. Penelitian ini membuat visualisasi pergerakan robot bertujuan untuk memonitoring pergerakan robot sesuai dengan sebenarnya (Indrarta, 2019). Tetapi masih ada kekurangan dari penelitian ini adalah pergerakan robot di visualisasi terputah-putah atau tidak smooth. Penelitian tersebut hanya terfokus pada visualisasi robot dan belum terdapat *Base Station* yang menjadi syarat perlombaan tersebut. Untuk menangani macam-macam persoalan yang ada, maka penulis ingin mengembangkan aplikasi *Base Station* yang sudah ada. Tujuan dari penelitian yaitu dapat mencatat dan menampilkan pergerakan robot secara lebih detail. Pengembangan ini dapat digunakan untuk meningkatkan strategi kedepannya. Pengembangan selanjutnya yaitu dapat meneruskan perintah dari *Refree Box* ke *Base Station* lalu di *broadcast* ke robot secara otomatis yang menjadi rule yang sebenarnya.

1. Split Data Atau Pemisah Data

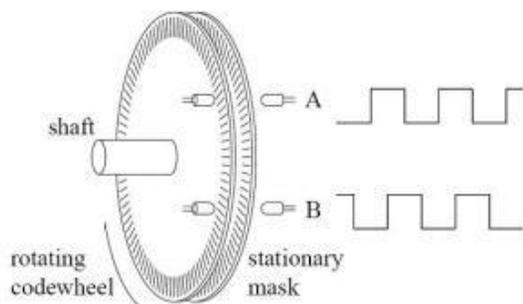


Gambar 1. Pemisah Data X, Y, θ

Dengan proses ini data yang semula bertipe data string dan belum dikelompokkan menjadi data individu yang diproses menjadi beberapa data yang siap dilakukan perhitungan (Abseno, 2019).

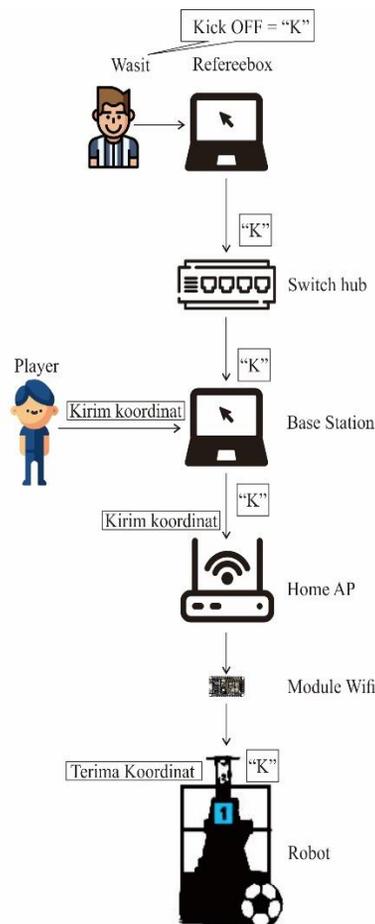
2. External Rotary Encoder

Rotary Encoder (ES403-100-3-T-24) merupakan sensor yang digunakan untuk menghitung putaran poros, sehingga digunakan untuk mendeteksi pergerakan posisi. Terdiri dari single track dan double track dan dua channel A dan B (gambar 2). Ketika poros berputar, deretan pulsa muncul di masing-masing channel pada frekuensi yang proporsional dengan kecepatan putar sedangkan hubungan fasa antara channel A dan B menghasilkan arah putaran. Dengan menghitung jumlah pulsa yang terjadi terhadap resolusi piringan, maka putaran dapat diukur. Untuk mengetahui arah putaran, dengan mengetahui channel mana yang leading terhadap channel satunya dapat kita tentukan arah putaran yang terjadi karena kedua channel tersebut selalu berbeda fasa seperempat putaran (Karnadi, 2019).



Gambar 2. External Rotary Encoder

METODE PENELITIAN



Gambar 3. Blok Diagram

Pada penelitian ini, penulis merancang sebuah aplikasi *Base Station* yang berfungsi untuk mengontrol gerak robot lebih *smooth* dan memvisualisasikan pada tampilan lapangan, memonitoring status dimana robot membawa bola atau tidak dan meneruskan perintah dari *RefereeBox* secara otomatis dan robot memberikan feedback nilai yang meliputi nilai koordinat, status, karakter dari *RefereeBox*. Agar dapat berkomunikasi antara aplikasi dan robot diperlukan koneksi jaringan yang langsung terkoneksi ke robot sepak bola beroda. Blok diagram ditunjukkan pada gambar 3.

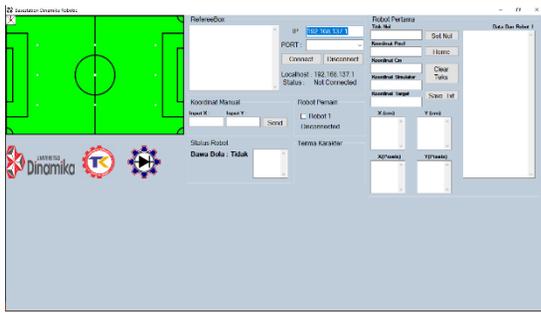
Rancangan penelitian tersebut menjelaskan tentang proses pengiriman data dari aplikasi basestation menuju robot sampai bisa untuk monitoring gerak robot.

1. Proses awal operator (TIM) melakukan koneksi ke IP nodemcu 8266 yang telah disetting

sebelumnya, yang berfungsi untuk berkomunikasi dan menukar data.

2. Setelah terkoneksi dengan robot. Robot mengirimkan nilai umpan balik yang meliputi sudut horizontal koordinat X, sudut vertikal koordinat Y, statusbola, dan karakter *Referee Box*. Data tersebut di split dan diambil data X dan Y khusus untuk pergerakannya robot, posisi awal robot mengirimkan nilai X adalah 0, dan y adalah 0. Posisi robot sebenarnya berada pada pojok kanan atas lapangan.
3. Setelah itu operator (Tim) juga harus melakukan koneksi ke aplikasi *Referee Box* milik wasit guna untuk mengatur sebuah pertandingan. Dengan menginputkan IP laptop milik wasit dan Port 28097. Setelah terkoneksi informasi yang dikirimkan oleh wasit aplikasi dapat mengetahuinya.
4. Setelah sudah terkoneksi dengan robot dan aplikasi *Referee Box*, aplikasi *Base Station* ini sudah dapat digunakan untuk memposisikan robot sesuai dengan titik x dan y yang telah di setting oleh tim.
5. Aplikasi *Base Station* mengolah data inputan dari operator kemudian dikirimkan ke nodemcu 8266 lalu dikirim ke robot. Data yang dikirimkan masih berupa data *pixel* butuh konversi sebelum dikirimkan ke robot.
6. Data dikirim dari *basestation* lalu diterima oleh router dan di sebarakan kepada kedua tim atau kepada IP yang sudah terhubung dengan router melalui kabel UTP.
7. Data diterima dengan nodemcu 8266 yang memiliki IP yang sudah terhubung oleh aplikasi *Base Station*. Nodemcu 8266 meneruskan pada mikrokontroler pada robot.
8. Data diterima oleh Arduino due melalui komunikasi serial dengan nodemcu8266, data tersebut diproses menjadi pergerakan sesuai target oleh arduino due.
9. Robot berjalan dan menghasilkan *output* putaran *external rotary* yang nantinya di proses oleh Arduino due lalu dikirim ke aplikasi melalui nodemcu8266 sebagai nilai untuk pergerakan robot di aplikasi. Pengiriman tersebut juga ada status bola, dan data karakter dari *Referee Box*.

1. Perancangan Aplikasi *Basestation*

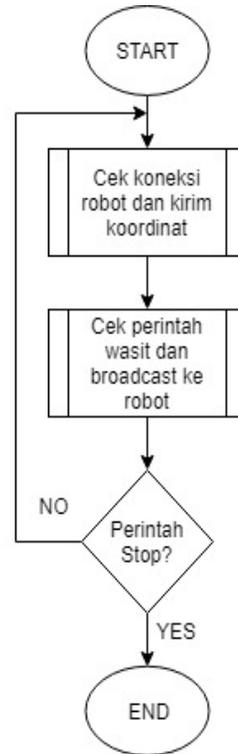


Gambar 4. Aplikasi *Basestation*

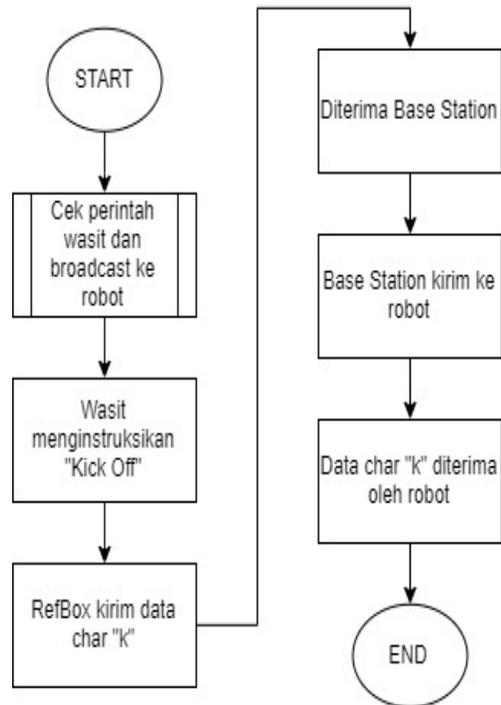
Base Station sebuah aplikasi untuk memantau gerak robot atau sebagai control robot secara otomatis berpacu dari instruksi dari *Referee Box* (Ardhiansyah & Syarifuddin, 2017). Pada gambar 4 tersebut adalah graphical user interface (GUI). Terdapat gambar lapangan, gambar robot, grupbox *referee box*, grup koordinat manual, grup robot pemain, grup status bola, grup terima karakter, grup robot pertama, dimana masing-masing dari grup didalamnya terdapat toolbox-toolbox.

A. Algoritma Visualisasi

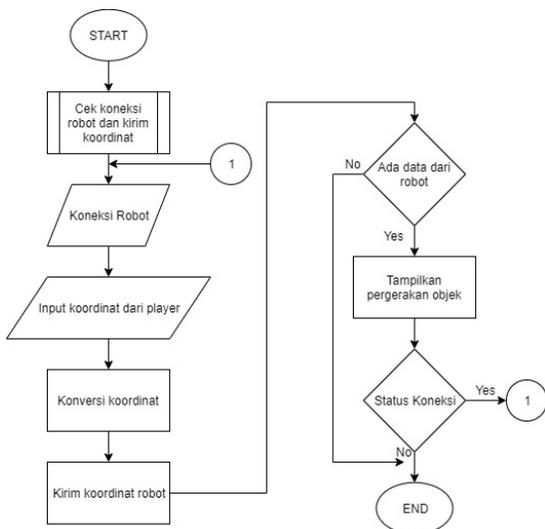
Berikut penjelasan *flowchart* gambar 5 dan gambar 6, Setelah aplikasi terkoneksi dengan komputer wasit, maka wasit dapat mengirimkan sebuah instruksi kepada tim yang bertanding melalui masing-masing aplikasi *Base Station* dari kedua tim, dari *flowchart* di atas wasit mengirimkan perintah *Kick Off* yang diwakili dengan karakter 'K' melalui *Referee Box*, lalu diterima oleh *Base Station* kemudian di *broadcast* ke robot dan diterjemahkan oleh robot dan dikirim balik ke aplikasi *Base Station*.



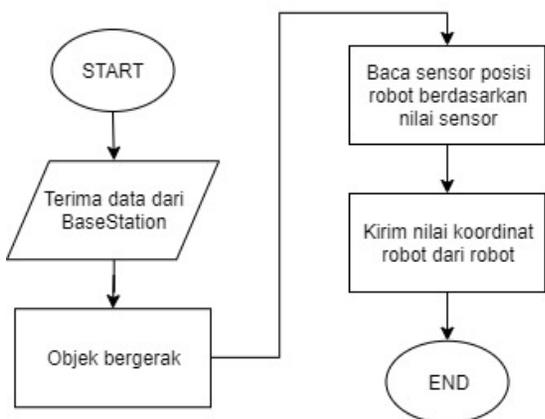
Gambar 5. Flowchart Alur Komunikasi



Gambar 6. Flowchart penerimaan data dari *Referee Box*



Gambar 7. Flowchart kontrol simulator



Gambar 8. Flowchart kirim data dari robot

Gambar *flowchart* 7 dan 8 adalah menjelaskan mengenai proses pada yang berjalan sekuensial. Langkah awal untuk mengoperasikan aplikasi ini adalah user harus koneksikan pada Robot 1. Setelah terhubung antara aplikasi *Base Station* dan robot lalu tim dapat mengirimkan target koordinat x dan y dan diteruskan ke robot. Robot menerjemahkan inputan tersebut menjadi pergerakan robot sesuai dengan input user dan mengirimkan nilai koordinat robot kepada aplikasi *Base Station*. Aplikasi *Base Station* memroses data yang diterima dari robot lalu menerjemahkan data tersebut menjadi pergerakan simulator robot sesuai dengan gerak robot yang sebenarnya.

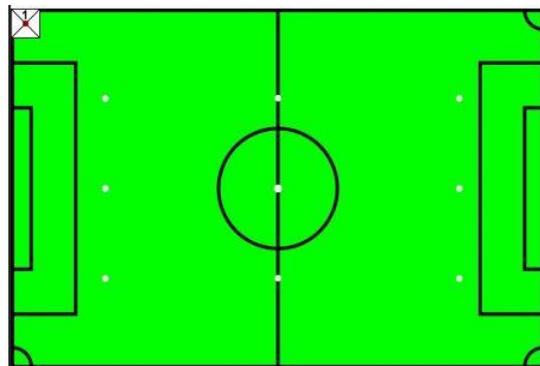
B. Rancang Lapangan Pada Visualisasi

Adapun rancangan pada sebuah lapangan yang memiliki ukuran yang sebenarnya adalah panjang 9 meter dan lebar 6 meter. Pada

perancangan lapangan pada aplikasi *Base Station* menggunakan ketentuan 1 *pixel* = 2 *cm* yaitu dengan merubah panjang 900 *cm* lalu dikali dengan 0,5 dan menghasilkan nilai 450 *cm*, dan sisi lebar 600 *cm* juga dikali dengan 0,5 dan menghasilkan nilai 300 *cm*. Karena untuk memperkecil ukuran lapangan pada aplikasi *Base Station* agar untuk menghemat *space* (ruang) untuk keperluan lainnya. Dan juga untuk merancang dalam sebuah aplikasi visual studio menggunakan satuan *pixel*, maka dari *cm* perlu di konversikan menjadi *pixel* dengan pedoman 2 *cm* = 1 *pixel*, pedoman tersebut berasal dari 900 *cm* dibagi dengan 450 *pixel* dan menghasilkan nilai 2, maka itu yang penulis menjadikan acuan dari penelitian ini.

C. Rancang Robot Pada Visualisasi

Robot yang rancang ini memiliki dimensi lebar 50 *cm* dan panjang juga 50 *cm*, maka perlu penyesuaian seperti rancangan lapangan perlu dikali dengan nilai 0,5 sehingga mendapat panjang 25 *cm* dan lebar 25 *cm*, maka langsung diimplementasikan ke aplikasi *Base Station* dengan panjang dan lebar 25 *pixel*. Untuk lebih jelasnya bisa dilihat pada gambar 9.

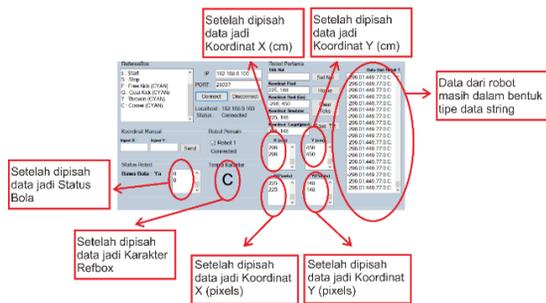


Gambar 9. Robot pada aplikasi

D. Rancang Pisah Data Visualisasi

Setelah menerima data robot dan ditampilkan di variable di aplikasi *Base Station* masih dalam bentuk tipe data *string*, maka diperlukan sebuah proses pemisahan data konversi tipe data, agar memudahkan untuk pengolahan data. Data diterima oleh aplikasi *Base Station* dari robot adalah dalam bentuk koordinat X: koordinat Y: StatusBola: Karakter *Referee Box*. Data yang ditampilkan dipisah atau displit oleh karakter ":" yang nantinya disimpan dalam bentuk array untuk mengolah data yang nantinya membedakan nilai, Status Bola, Karakter *Referee Box*, koordinat X, koordinat Y. Gambar 10 menunjukkan data yang terima dari robot.

Data yang diterima melalui proses pisah data agar bisa dijadikan sebagai inputan koordinat X dan Y, Status bola, dan Karakter *Referee Box*. Yang nantinya ada proses konversi buat koordinat X dan Y untuk kebutuhan visualisasi gambar robot pada aplikasi.



Gambar 10. Proses pisah data

2. Perancangan Komunikasi Robot

Pada perancangan KRSBI ini memakai arduino due untuk sebagai proses pergerakan robot atau kontrol robot, sensor proximity untuk sebagai sensor bahwa robot dalam kondisi membawa bola atau tidak dan Nodemcu ESP8266 sebagai pengirim data. Dua catu daya yang digunakan yaitu 5 volt untuk nodemcu esp8266, motor driver, external rotary. Sedangkan 12-24 volt digunakan untuk daya aktuator yaitu motor.

Memakai motor driver dengan tipe EMS-30A dengan kapasitas arus sebesar 30A yang fungsinya untuk menggerakkan motor DC dengan sumber tegangan yang berasal dari baterai lipo tiga sel. Membutuhkan tiga inputan untuk menggerakkan sebuah motor yaitu direksi 2, direksi 1, dan PWM. Direksi 1 dan 2 digunakan untuk menentukan arah putaran motor, jika direksi 1 kondisi *high* dan direksi 2 kondisi *low*, maka motor bergerak sesuai dengan arah jarum jam begitu juga sebaliknya. Untuk PWM hanya perlu memberi inputan berupa pulsa dengan perbandingan *duty cycle* sesuai dengan kebutuhan guna mengatur kecepatan putaran motor DC.

Sensor proxymity yang digunakan penulis adalah tipe E18-D80NK, sensor ini membutuhkan catu daya sebesar 5V. Sensor ini berfungsi sebagai *switch* otomatis, jika terdapat bola, maka bernilai 0 (*low*) dan jika tidak terdapat bola, maka bernilai 1 (*high*) yang artinya aktif *low*.

A. Terima Data Koordinat

Setelah aplikasi mengirim data berupa koordinat maupun perintah dari wasit pada robot, dari sisi robot dengan nerima data dari nodemcu

esp8266 ke mikrokontroler melalui komunikasi serial. Berikut alur dari penerimaan data koordinat dan karakter *Referebox* yaitu cek data masuk pada serial 3 jika ada data masuk disimpan pada variabel terima, cek kepala dan ekor data yang diwakili oleh karakter '<' dan '>' dan disimpan pada variabel data, cek ulang ekor data diwakili oleh karakter '>' sebagai akhir dari sebuah data dan set *flag s* menjadi *true*, cek kondisi *s* jika *true* memanggil fungsi pisah data dan set *flag s* menjadi *false*, cek data lagi dari beberapa karakter meliputi : s, S, N, L, h, dan disimpan di variabel *feedback*.

HASIL DAN PEMBAHASAN

Terdapat tiga pengujian yaitu, uji validasi dan komunikasi antara robot dan aplikasi *base station*, uji kesamaan pergerakan robot sebenarnya dengan pergerakan robot di aplikasi *base station* dan uji kecepatan pengiriman data robot ke aplikasi *base station*. Dalam pengujian tersebut pengujian mencoba mengirimkan sejumlah perintah dari *Referee Box* ke *Base Station*, pengujian kedua robot diuji coba dengan gerakan secara ke maju, samping, serong dengan cara user memberikan koordinat pada aplikasi *base station* kepada robot dan pengujian ketiga di uji coba beberapa nilai *range* periode untuk pengiriman data.

1. Uji Validasi Dan Komunikasi Antara Robot Dan Aplikasi Base Station

Pengujian ini berfungsi untuk mengetahui data dari perintah dari aplikasi *RefereeBox* yang terima oleh aplikasi *base station* dan dikirimkan ke robot lalu dikembalikan lagi di aplikasi *base station* telah sesuai atau tidak sesuai. Berdasarkan dari semua hasil uji coba yang dilakukan pada hasil uji validasi dan komunikasi antara robot dan aplikasi *base station*. Terbukti bahwa data yang dikirimkan dan diterima memiliki kesamaan data. Tingkat dari uji coba tersebut adalah 100%.

Tabel 1. Uji validasi dan komunikasi menu utama pada aplikasi *Referebox*.

Uji Coba Ke	<i>Referee Box</i> to <i>Base Station</i>	<i>Base Station</i> to Robot	Robot to <i>Base Station</i>	<i>Base Station</i>	Keterangan
1	START ('s')	"s"	"s"	"s"	Berhasil
2	STOP ('S')	"S"	"S"	"S"	Berhasil
3	DROPBALL ('N')	"N"	"N"	"N"	Berhasil
4	PARK ('L')	"L"	"L"	"L"	Berhasil
5	RESET ('Z')	"Z"	"Z"	"Z"	Berhasil
6	ENDPART ('h')	"h"	"h"	"h"	Berhasil

Tabel 2. Uji validasi dan komunikasi menu tim cyan pada aplikasi *Refereebox*.

Uji Coba Ke	Referee Box to Base Station	Base Station to Robot	Robot to Base Station	Base Station	Keterangan
1	Kick Off ("K")	"K"	"K"	"K"	Berhasil
2	Free Kick ("F")	"F"	"F"	"F"	Berhasil
3	Goal Kick ("G")	"G"	"G"	"G"	Berhasil
4	Throw In ("T")	"T"	"T"	"T"	Berhasil
5	Corner ("C")	"C"	"C"	"C"	Berhasil
6	Repair ("O")	"O"	"O"	"O"	Berhasil
7	Penalty ("P")	"P"	"P"	"P"	Berhasil
8	Goal Cyan ("A")	"A"	"A"	"A"	Berhasil
9	Red ("R")	"R"	"R"	"R"	Berhasil
10	Yellow ("Y")	"Y"	"Y"	"Y"	Berhasil

Tabel 3. Uji validasi dan komunikasi menu tim magenta pada aplikasi *refereebox*.

Uji Coba Ke	Referee Box to Base Station	Base Station to Robot	Robot to Base Station	Base Station	Keterangan
1	Kick Off ("k")	"k"	"k"	"k"	Berhasil
2	Free Kick ("f")	"f"	"f"	"f"	Berhasil
3	Goal Kick ("g")	"g"	"g"	"g"	Berhasil
4	Throw In ("t")	"t"	"t"	"t"	Berhasil
5	Corner ("c")	"c"	"c"	"c"	Berhasil
6	Repair ("o")	"o"	"o"	"o"	Berhasil
7	Penalty ("p")	"p"	"p"	"p"	Berhasil
8	Goal Cyan ("a")	"a"	"a"	"a"	Berhasil
9	Red ("r")	"r"	"r"	"r"	Berhasil
10	Yellow ("y")	"y"	"y"	"y"	Berhasil

2. Uji Kesamaan Pergerakan Robot Sebenarnya Dengan Pergerakan Robot Di Aplikasi Base Station.

Pengujian ini berfungsi untuk mengetahui Kesamaan dari nilai koordinat sebenarnya dan nilai koordinat robot pada aplikasi base station yang diharapkan terdapat Kesamaan pergerakan robot sebenarnya dengan robot pada aplikasi base station.

Tabel 4. Uji coba Kesamaan dengan gerak robot maju.

Waktu Detik Ke	Input koordinat X : 0 dan Y : 100						Error
	Gerak Robot (cm)		Gerak Visual (pixels)		Gerak Visual (cm)		
	X	Y	X	Y	X	Y	
1	0.06	0.59	0	0	0	0	0
	0.06	0.59	0	0	0	0	0
	0.06	0.59	0	0	0	0	0
2	0.06	0.59	0	0	0	0	0
	0.06	2.29	0	1	0	2	0
	0.06	12.35	0	6	0	12	0
3	0.06	16.05	0	8	0	16	0
	0.06	27.28	0	14	0	28	0
	0.00	31.11	0	16	0	32	0
4	0.06	42.51	0	21	0	42	0
	0.06	50.22	0	25	0	50	0
	0.06	53.98	0	27	0	54	0
5	0.00	65.50	0	33	0	66	0
	0.00	69.27	0	35	0	70	0
	0.06	80.91	0	40	0	80	0
6	0.06	88.50	0	44	0	88	0
	0.06	92.38	0	46	0	92	0
	0.00	100.73	0	50	0	100	0
	0.00	100.73	0	50	0	100	0

Berdasarkan dari semua hasil uji coba yang dilakukan pada hasil uji Kesamaan pergerakan robot sebenarnya dengan pergerakan robot di aplikasi *base station* yaitu terdapat Kesamaan nilai dari nilai pergerakan robot sebenarnya dengan robot pada aplikasi *base station*. Error pada uji coba di atas adalah 0%.

Tabel 5. Uji coba Kesamaan dengan gerak robot kesamping.

Waktu Detik Ke	Input koordinat X : 100 dan Y : 0						Error
	Gerak Robot (cm)		Gerak Visual (pixels)		Gerak Visual (cm)		
	X	Y	X	Y	X	Y	
1	0.41	0.18	0	0	0	0	0
	0.41	0.18	0	0	0	0	0
	0.41	0.18	0	0	0	0	0
2	0.41	0.18	0	0	0	0	0
	4.06	0	2	0	4	0	0
	10.7	0.06	5	0	10	0	0
3	17.6	0	9	0	18	0	0
	24.6	0	12	0	24	0	0
	31.6	0.06	16	0	38	0	0
4	38.7	0.06	19	0	38	0	0
	45.7	0.06	23	0	46	0	0
	52.7	0.06	26	0	52	0	0
5	59.8	0.06	30	0	60	0	0
	66.9	0.06	33	0	66	0	0
	73.9	0.06	37	0	74	0	0
6	81	0.06	40	0	80	0	0
	88	0.06	44	0	88	0	0
	95	0.06	48	0	96	0	0
	101	0.06	51	0	102	0	0
	100	0.06	50	0	100	0	0
	100	0.06	50	0	100	0	0
	100	0.06	50	0	100	0	0

Tabel 6. Uji coba Kesamaan dengan gerak robot serong

Waktu Detik Ke	Input koordinat X : 100 dan Y : 100						Error
	Gerak Robot (cm)		Gerak Visual (pixels)		Gerak Visual (cm)		
	X	Y	X	Y	X	Y	
1	0.06	0.41	0	0	0	0	0
	0.06	0.41	0	0	0	0	0
	0.06	0.41	0	0	0	0	0
2	0.06	0.41	0	0	0	0	0
	0.06	0.41	0	0	0	0	0
	4.94	4.65	2	2	4	4	0
3	11.76	10.47	6	5	12	10	0
	18.82	16.52	9	8	18	16	0
	29.58	25.70	15	13	30	26	0
4	33.11	28.75	17	14	34	28	0
	40.34	34.87	20	17	40	34	0
	47.51	40.93	24	20	48	40	0
5	54.63	46.98	27	23	54	46	0
	65.39	56.10	33	28	66	56	0
	68.97	59.15	34	30	68	60	0
6	79.73	68.27	40	34	80	68	0
	83.32	71.27	42	36	84	72	0
	90.55	77.32	45	39	90	78	0
	100.32	86.56	50	43	100	86	0
	100.46	90.03	50	45	100	90	0
	99.79	100.67	50	50	100	100	0
	99.73	101.02	50	50	100	100	0

3. Uji Kecepatan Pengiriman Data Robot Ke Aplikasi Base Station.

Pengujian ini berfungsi untuk mengetahui *range* periode pengiriman terbaik untuk pengiriman data dari robot ke aplikasi *base station*. Pemilihan waktu periode pengiriman dapat mempengaruhi cepat lambatnya pengiriman data dan gerak robot pada aplikasi *base station*.

Hasil yang diperoleh dari hasil uji kecepatan pengiriman data robot ke aplikasi *base station* mendapat kesimpulan bahwa nilai aman buat pengiriman penelitian ini adalah *range* periode pengiriman 33 sampai 200 milidetik karena memiliki jumlah data terbanyak. Dan periode pengiriman 17 milidetik sangat tidak cocok untuk diterapkan karena data yang diterima adalah 0 atau tidak ada data yang masuk.

Tabel 7. Uji kecepatan pengiriman data robot ke aplikasi Base Station

Percobaan Ke	Periode Pengiriman (milidetik)	Jumlah Data Yang Terima
1	668	31
2	634	33
3	601	34
4	568	36
5	534	39
6	501	41
7	468	43
8	434	46
9	401	50
10	367	54
11	334	60
12	301	67
13	267	76
14	234	85
15	200	97
16	167	97
17	134	97
18	84	97
19	50	97
20	33	97
21	17	0

KESIMPULAN

Kesimpulan yang didapatkan dari penelitian ini adalah sebagai berikut:

1. Hasil yang diperoleh dari hasil uji kesamaan pergerakan robot sebenarnya dengan pergerakan robot pada aplikasi *base station* dengan tingkat kesamaan 100%. Dengan ketentuan 1 *pixel* = 2 cm dan hasilnya baik.
2. Hasil yang diperoleh dari proses uji validasi komunikasi antara robot dengan aplikasi *base station*, terbukti data yang dikirimkan dan diterima memiliki kesamaan dan tingkat uji coba tersebut adalah 100%.

3. Hasil yang diperoleh dari hasil uji kecepatan pengiriman data robot ke aplikasi *base station* mendapat kesimpulan bahwa nilai aman buat pengiriman penelitian ini adalah *range* periode pengiriman 33 sampai 200 milidetik karena memiliki jumlah data terbanyak. Dan periode pengiriman 17 milidetik sangat tidak cocok untuk diterapkan karena data yang diterima adalah 0 atau tidak ada data yang masuk.

Saran

Ada beberapa saran untuk pengembangan penelitian selanjutnya diharapkan robot dapat mengetahui daerah letak gawang lawan melalui visualisasi posisi robot, pembatas agar robot tidak keluar dari garis tepi lapangan, dan robot dapat mengetahui daerah tim cyan dan daerah tim magenta melalui visualisasi posisi robot.

DAFTAR PUSTAKA

- Abseno, A. P. (2019). *Penerapan Kinematika Untuk Lokalisasi Pada Robot Sepak Bola Beroda*. Surabaya: Institut Bisnis dan Informatika Stikom Surabaya.
- 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.
- Indrarta, V. (2019). PERANCANGAN VISUALISASI POSISI ROBOT PADA PENGENDALI ROBOT SEPAK BOLA BERODA. *Journal of Control and Network Systems*.
- Karnadi, D. A. (2019). RANCANG BANGUN KONTROL KECEPATAN ROBOT THREE OMNI-DIRECTIONAL MENGGUNAKAN METODE PID (Proportional Integral Derivative). *Journal of Control and Network Systems*.
- Kementrian Riset, T. d. (2019). *Buku Panduan KRSBI Beroda 2019*. Diambil kembali dari <http://kontesrobotindonesia.id/tentang-kri.html>