

## DESAIN DAN IMPLEMENTASI KOMUNIKASI CONTROL ROBOT SOCCER BERODA MENGGUNAKAN USER DATAGRAM PROTOCOL (UDP)

Azizul<sup>1</sup>, Syaiful Amri<sup>2</sup>, Nirwan Budiyanto<sup>3</sup>, Muhammad Zamhuri<sup>4</sup>  
Politeknik Negeri Bengkalis  
azizul@polbeng.ac.id<sup>1</sup>, syaifulamri@polbeng.ac.id<sup>2</sup>, budi@polbeng.ac.id<sup>3</sup>,  
zamhuri@polbeng.ac.id<sup>4</sup>

### Abstract

*A wheel soccer robot is a robot which is programmed to play a soccer game in specific way such as a soccer game system with movement system by using wheels. In commonly, the wheel soccer robot is an automatically controlled by using camera and searching for an orange ball object. In this final project, the wheel soccer robot was designed to be able to control by using application of PC/Laptop GUI (graphical user interface) which is used NetBeans IDE (integrated development environment). The communication used in this system is UDP (user datagram protocol). GUI application controller will be operated on PC / laptop and connect it to mini PC robot with IP address connection system. The PC/Laptop controller will be given movement instructions data to the robot than, the data will be executed and forwarded to arduino board via serial USB port communication. In controlling, the robot just can be used wifi network. The average distance that can be reached in this controlling is range of 125 meters.*

**Keyword :** *Wheel soccer robot, GUI (graphical user interface), UDP (user datagram protocol), NetBeans, serial communication.*

### 1. PENDAHULUAN

Robotika kini telah menjadi bagian dari kehidupan dan menjadi salah satu perkembangan teknologi suatu negara. Beberapa negara maju terus berupaya untuk terus mengembangkan penelitian di dunia robotika. Dari sekian banyak macam robot yang terus dikembangkan, salah satunya adalah robot pemain sepakbola (Robot *Soccer*). Sebuah robot yang beraksi di dalam lapangan sepakbola *mini* dengan peraturan yang lebih sederhana jika dibandingkan dengan sepakbola yang sebenarnya.

Para peneliti robot *soccer* ini terus melakukan berbagai riset. Pembagian divisi robot pun dibagi menjadi dua yaitu divisi robot *soccer humanoid* dan divisi robot *soccer* beroda. Divisi robot *soccer humanoid* berkonsentrasi dalam membangun sebuah mekanik robot pemain sepakbola layaknya seperti manusia. Divisi robot *soccer* beroda berkonsentrasi pada *Artificial Intelligent* (AI) untuk mengenali pergerakan bola, lawan dan kawan dengan sistem gerak menggunakan roda.

Pada alat ini robot *soccer* beroda yang akan dibuat tidak sama halnya dengan robot *soccer* beroda yang ada dipertandingan kontes robot yang ada di Indonesia. Robot yang akan dibuat adalah robot yang bisa dikendalikan dari jarak jauh secara manual menggunakan PC/laptop *controller* melalui jaringan *wifi*. Jadi, robot bisa bermanuver hanya dengan mengontrol melalui PC/laptop tanpa harus menggunakan sensor kamera.

### 2. TINJAUAN PUSTAKA

Robot *soccer* beroda adalah robot yang dirancang untuk melakukan permainan sepakbola dengan aturan-aturan tertentu seperti permainan sepakbola dengan sistem pergerakan menggunakan roda. Robot ditugaskan untuk mencari, mengiring, serta memasukkan bola ke gawang lawan. Maka dari itu untuk itu robot harus bisa melakukan

*manuver*. Dengan mengubah arah putaran keempat buah *motor* maka robot dapat bergerak maju, mundur, geser, rotasi dan diagonal.

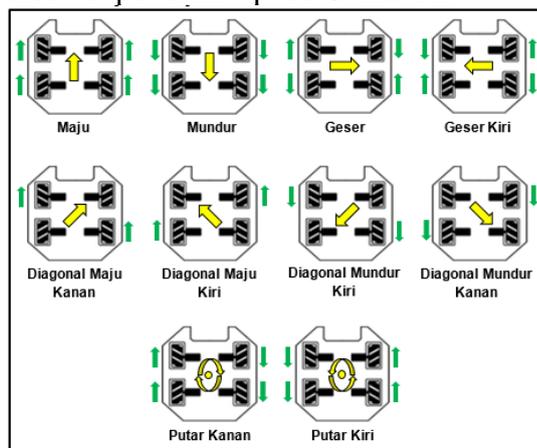
Mannes, Poel (2003), mengatakan robot sepak bola adalah masalah standar dimana teknologi baru dapat diintegrasikan dan diuji. Contohnya adalah pengolahan gambar, teori kontrol, kecerdasan buatan, sistem *multi-agen*, perencanaan gerak dan sistem yang tertanam. Salah satu inisiatif sepak bola adalah *Federasi Internasional Robot Soccer Association* (FIRA). Sebuah tim yang telah dikembangkan yang berkompetisi di FIRA Kejuaraan Eropa 2003 mendapat tempat keempat dikategori *middle size MiroSot League*. Sistem ini dikembangkan menggunakan kamera di atas lapangan yang akan menyediakan semua informasi yang diperlukan. Informasi ini akan digunakan oleh satu atau lebih komputer yang menghitung perintah robot yang sesuai. Perintah ini akan dikirim ke robot dengan komunikasi *frekuensi radio*

F. B. Samudra, dkk. (2011), pada jurnalnya yang berjudul “Pengendalian robot *soccer* dalam mengambil bola dengan metode *evolutionary artificial potential field*” ini membahas suatu program yang berguna untuk simulasi robot *soccer* dalam mencari jalur aman untuk mengambil bola dengan mempertimbangkan posisi robot terhadap gawang lawan. Metode yang digunakan adalah *Evolutionary Artificial Potential Field* (EAPF) yaitu metode pengembangan dari *Potential Field* (PF) yang memperhatikan gaya dari halangan serta bola. *Metode Evolutionary Artificial Potential Field* (EAPF) dipilih karena prosesnya sederhana dan diharapkan dapat memposisikan robot menghadap kearah gawang lawan.

### Sistem Gerak Pada Robot Soccer Beroda

Alat gerak yang digunakan pada robot adalah menggunakan empat buah roda kenis *mecanum wheels*, tujuan penggunaan roda jenis ini adalah karena memiliki kelebihan yang tidak dimiliki oleh roda biasa. Dengan menggunakan roda jenis *mechanum wheels* ini tidak hanya dapat melakukan pergerakan maju dan mundur namun juga dapat melakukan pergerakan geser kiri, geser kanan serta pergerakan diagonal kiri dan diagonal kanan.

Melakukan gerakan *manuver* adalah hal yang paling penting pada robot. Yaitu dengan merubah arah perputaran keempat buah *motor* maka robot sudah bisa melakukan berbagai gerakan *manuver*. Contohnya, untuk bergerak putar ke kanan arah putaran keempat buah *motor* digerakkan dengan searah jarum jam, sedangkan untuk bergerak putar ke kiri arah perputaran keempat buah *motor* digerakkan dengan berlawanan arah jarum jam. Skema pergerakan robot *soccer* beroda dapat dilihat pada Gambar 1.



Gambar 1. Skema pergerakan yang dapat dilakukan oleh robot

### Komunikasi UDP

UDP singkatan dari *user datagram protocol* adalah salah satu protokol lapisan transpor *TCP/IP* yang mendukung komunikasi *unreliable* yang merupakan bagian dari *internet protocol*. Dengan UDP, aplikasi komputer dapat mengirimkan pesan kepada

komputer lain dalam jaringan lain tanpa melakukan komunikasi awal. UDP melakukan komunikasi secara sederhana dengan mekanisme yang sangat minimal. Ada proses *checksum* untuk menjaga integritas data.

UDP mempunyai karakteristik *connectionless* (tidak berbasis koneksi). Data yang dikirimkan dalam bentuk *packet* tidak harus melakukan *call setup*. Data dalam protokol UDP akan dikirimkan sebagai datagram tanpa adanya nomor *identifier*. Sehingga sangat besar sekali kemungkinan data sampai tidak berurutan dan sangat mungkin hilang/rusak dalam perjalanan dari *host* asal ke *host* tujuan.

### **Komunikasi Serial**

Komunikasi *serial* adalah komunikasi yang pengiriman datanya per-bit secara berurutan dan bergantian. Komunikasi ini mempunyai suatu kelebihan yaitu hanya membutuhkan satu jalur dan kabel yang sedikit dibandingkan dengan komunikasi *parallel*. Pada prinsipnya komunikasi *serial* merupakan komunikasi dimana pengiriman data dilakukan per-bit sehingga lebih lambat dibandingkan komunikasi *parallel* atau dengan kata lain komunikasi *serial* merupakan salah satu metode komunikasi data di mana hanya satu *bit* data yang dikirimkan melalui seuntai kabel pada suatu waktu tertentu.

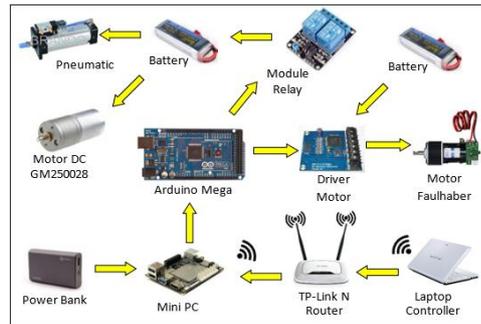
Pada dasarnya komunikasi *serial* adalah kasus khusus komunikasi *parallel* dengan nilai  $n = 1$  atau dengan kata lain adalah suatu bentuk komunikasi *parallel* dengan jumlah kabel hanya satu dan hanya mengirimkan satu bit data secara simultan. Hal ini dapat disandingkan dengan komunikasi *parallel* yang sesungguhnya di mana  $n$ -bit data dikirimkan bersamaan dengan nilai umumnya  $8 \leq n \leq 128$ . Hal yang paling penting dalam menghubungkan dua perangkat melalui komunikasi *serial* adalah memastikan bahwa kedua perangkat berkomunikasi dengan konfigurasi yang sama. Terdapat beberapa *parameter* yang digunakan untuk membangun komunikasi secara *serial*, diantaranya adalah *baud rate*, *paket data*, *parity bit* dan *synchronization bit*.

### **3. METODE PENELITIAN**

Pada tahap ini akan memaparkan tentang metodologi penelitian yang digunakan, yaitu terdiri dari perancangan *system hardware* dan perancangan *system software*. Adapun tahapannya adalah sebagai berikut.

#### **Perancangan Hardware**

Perancangan blok diagram *system hardware* keseluruhan pada penelitian ini ditunjukkan pada Gambar 2. Dari gambar blok diagram dan hasil perancangan tersebut dapat dijelaskan *Arduino MEGA 2560* berfungsi sebagai mikrokontroler yang nantinya akan mengontrol keempat buah *driver motor* dan *module relay*. *Mini PC* ini berfungsi sebagai *receiver* data yang akan dikirim oleh PC/laptop *controller* dan meneruskan data yang diterima ke *board Arduino*. *TP-Link N Router* Berfungsi sebagai perantara penghubung antara PC/laptop *controller* dan *mini PC* robot. Laptop *controller* digunakan sebagai pengendali robot dan laptop juga yang akan mengirim data instruksi ke robot. *Driver motor* berfungsi sebagai mengontrol arah perputaran dan kecepatan *motor DC faulhaber*. *Motor faulhaber* berfungsi sebagai aktuator utama yang menggerakkan roda *mechanum wheels*. *Module relay* digunakan sebagai *switch* yang bekerja untuk memutuskan dan menghubungkan tegangan dari *battery* ke *valve pneumatic*. *Pneumatic* Digunakan sebagai aktuator penendang objek yaitu berupa bola. *Battery* digunakan untuk men-supply *valve pneumatic*, *board Arduino* dan keempat *driver motor* yang digunakan. *Motor DC GM250028* Digunakan sebagai menggring bola. *Power bank* digunakan untuk men-supply *mini PC*.

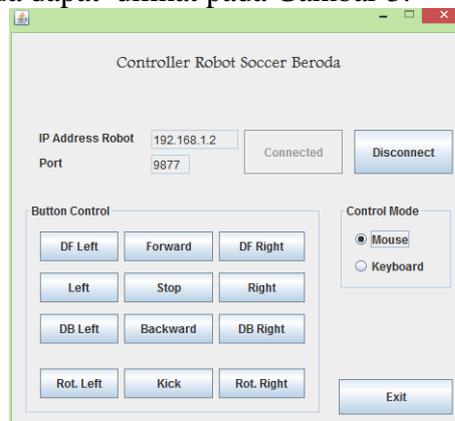


Gambar 2. Blok diagram *system hardware* keseluruhan alat

### Perancangan Software

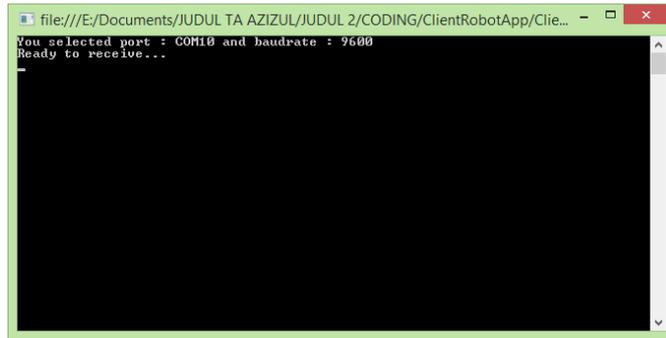
Perancangan perangkat lunak dalam alat ini juga merupakan langkah yang paling penting dalam merealisasikan sistem yang sesungguhnya. Tujuan dari perancangan ini adalah untuk membangun sebuah algoritma-algoritma yang dibuat melalui *coding* program sesuai dengan spesifikasi dan cara kerja dari sistem yang ingin dibuat. Perancangan perangkat lunak ini terdiri dari pembuatan *form application controller*, *console application receiver* dan program *Arduino*.

Tampilan *GUI controller* robot *soccer* beroda ini didesain dengan sangat *simple* agar bisa dan mudah dalam penggunaannya. Maka dibuatlah pengontrolan dengan menggunakan dua sistem *mode*, yaitu *mode controll keyboard* dan *mode controll mouse*. *Mode controll keyboard* adalah dimana *mode* pengontrolan dilakukan dengan menekan tombol *keyboard* pada *PC/laptop controller*. Sedangkan *mode contrtroll mouse* adalah dimana *mode* pengontrolan dengan mengklik *button controller* menggunakan *mouse*. Tampilan *GUI controller* robot *soccer* beroda dapat dilihat pada Gambar 3.



Gambar 3. Tampilan *form application controller*

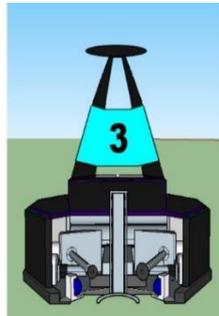
Setelah pembuatan *form application controller*, *console application receiver* juga dibuat dalam perancangan *system software* ini. *Console application* ini dirancang sebagai program *receiver* dan dijalankan di *mini PC* robot. Program *receiver* ini akan terus *standby* menerima data masuk dan ketika program *receiver* ini telah menerima data yang dikirim oleh *PC/laptop controller*, data tersebut akan diteruskan ke *board Arduino MEGA* dengan melalui komunikasi *serial USB* dan program *Arduino* akan menjalankan instruksi pergerakan yang telah dikirim dari *PC/laptop controller* tadi. Tampilan *console application* program *receiver* dapat dilihat pada Gambar 4.



Gambar 4. Tampilan *console application receiver*

### Perancangan Desain Robot

Untuk mempermudah dalam pembuatan alat ini, maka dibuat langkah-langkah perancangan untuk mempermudah dalam merealisasikan alat yang akan dibuat. Desain perancangan dibuat menggunakan *software* aplikasi *sketchup* guna untuk mendapatkan hasil desain yang 3 dimensi. Desain perancangan mekanik *body/chasis* robot dapat dilihat pada Gambar 5 dan Hasil mekanik *body/chasis* robot yang telah dibuat dapat dilihat pada Gambar 6.



Gambar 5. Desain perancangan mekanik *body/chasis* robot



Gambar 6. Hasil perancangan mekanik *body/chasis* robot

### Cara Kerja Sistem

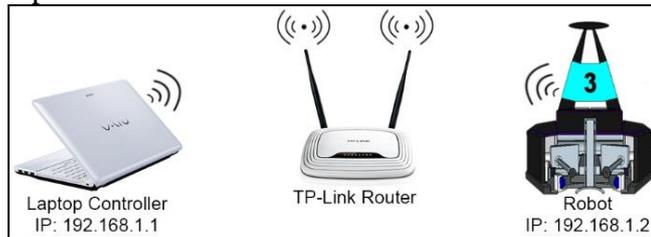
Secara garis besar robot hanya bisa dikendalikan oleh PC/laptop *controller* yang sudah saling terhubung dalam satu jaringan. Selain itu juga antara PC/laptop *controller* dan *mini PC* robot harus dalam alamat *IP address* yang sekelas. Robot hanya akan bergerak sesuai dengan data instruksi yang dikirim dari PC/laptop *controller*. Keterangan pergerakan robot dapat dilihat pada table 1.

Tabel 1. Keterangan pergerakan robot

Tombol GUI Controller	Tombol Keyboard	Instruksi Robot
<i>Forward</i>	W	Bergerak maju
<i>Backward</i>	X	Bergerak mundur
<i>Left</i>	A	Bergerak ke kiri

<i>Right</i>	D	Bergerak ke kanan
<i>DF Right</i>	E	Bergerak diagonal kanan depan
<i>DF Left</i>	Q	Bergerak diagonal kiri depan
<i>DB Right</i>	C	Bergerak diagonal kanan belakang
<i>DB Left</i>	Z	Bergerak diagonal kiri belakang
<i>Rot. Left</i>	J	Bergerak rotasi kiri
<i>Rot. Right</i>	L	Bergerak rotasi kanan
<i>Kick</i>	K	Menendang
<i>Stop</i>	S	Berhenti

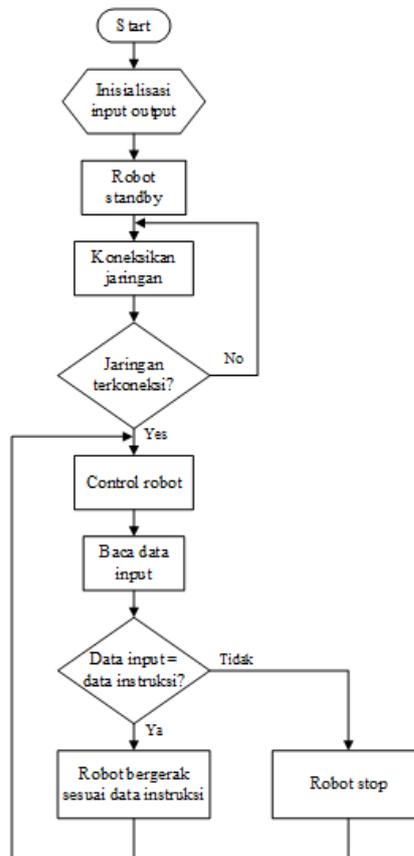
Jarak pengontrolan robot masih terbatas dan akan mempengaruhi pada saat robot mengeksekusi data yang diterima. Skema sistem koneksi antara robot dan PC/laptop controller dapat dilihat pada Gambar 7.



Gambar 7. Skema sistem koneksi antara robot dan PC/laptop controller

### Flowchart Program Robot

Flowchart adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program. Flowchart jalannya program ditunjukkan pada Gambar 8.



Gambar 8. Flowchart program

#### 4. HASIL PENELITIAN DAN PEMBAHASAN

Desain kontroler yang telah dibuat menggunakan *software NetBeans IDE* akan diuji terlebih dahulu, apakah *command* atau perintah pergerakan robot sudah sesuai dengan *button control* pada *GUI controller*. Mini PC robot akan menjalankan program *receiver* yang telah dibuat menggunakan *C# console application* menggunakan *software visual studio*. Mini PC robot akan terus *standby* menerima data yang akan dikirim oleh PC/laptop *controller* apabila antara robot dan PC/laptop *controller* sudah saling terhubung.

Ketika robot pertama kali dihidupkan PC/laptop *controller* bertugas untuk mengkoneksikan *IP address* robot melalui *button connect* yang terdapat pada *text box GUI controller*. Apabila PC/laptop *controller* dan robot sudah saling terhubung maka robot siap untuk dikontrol. Pengujian desain *controller* ini dilakukan pada komputer dan juga laptop dengan tujuan untuk merealisasikan sistem *controller* ini agar bisa digunakan pada komputer dan juga laptop.

##### Pengujian Mode Kontrol Mouse

Mode kontrol *mouse* adalah pengontrolan robot menggunakan *mouse* klik, yaitu dengan mengklik *button GUI* yang terdapat pada *grup box button control*. Pengujian ini dilakukan untuk mengetahui apakah *button control* sudah sesuai dengan pergerakan robot atau belum. Hasil pengujian menggunakan *mode control mouse* dapat dilihat pada Tabel 2.

Tabel 2. Pengujian menggunakan *mode control mouse*.

Button control	Pergerakan robot	PC (Komputer)		Laptop	
		Berhasil	Gagal	Berhasil	Gagal
<i>Forward</i>	Robot bergerak maju	✓	-	✓	-
<i>Backward</i>	Robot bergerak mundur	✓	-	✓	-
<i>Left</i>	Robot bergerak ke kiri	✓	-	✓	-
<i>Right</i>	Robot bergerak ke kanan	✓	-	✓	-
<i>DF Right</i>	Robot bergerak diagonal kanan depan	✓	-	✓	-
<i>DF Left</i>	Robot bergerak diagonal kiri depan	✓	-	✓	-
<i>DB Right</i>	Robot bergerak diagonal kanan belakang	✓	-	✓	-
<i>DB Left</i>	Robot bergerak diagonal kiri belakang	✓	-	✓	-
<i>Rot. Left</i>	Robot bergerak rotasi kiri	✓	-	✓	-
<i>Rot. Right</i>	Robot bergerak rotasi kanan	✓	-	✓	-
<i>Kick</i>	Robot menendang	✓	-	✓	-
<i>Stop</i>	Robot berhenti	✓	-	✓	-

Dari hasil pengujian yang telah dilakukan bahwa robot bisa dikoneksikan dengan menggunakan komputer dan laptop. Pengujian *button control* pada *GUI controller* sesuai dengan instruksi pergerakan robot.

##### Pengujian Mode Kontrol Keyboard

Mode kontrol *keyboard* adalah pengontrolan robot menggunakan tombol *keyboard*, yaitu dengan menekan tombol *keyboard* yang terdapat pada PC/laptop *controller* yang tombol *keyboard*-nya telah ditentukan dipemrogramannya. Pengujian ini dilakukan untuk mengetahui apakah *input* tombol *keyboard* sudah sesuai dengan pergerakan robot atau belum. Hasil pengujian menggunakan *mode control keyboard* dapat dilihat pada Tabel 3.

Tabel 3. Pengujian menggunakan *mode control keyboard*.

Button control	Pergerakan robot	PC (Komputer)		Laptop	
		Berhasil	Gagal	Berhasil	Gagal
<i>W</i>	Robot bergerak maju	✓	-	✓	-
<i>X</i>	Robot bergerak mundur	✓	-	✓	-
<i>A</i>	Robot bergerak ke kiri	✓	-	✓	-

<i>D</i>	Robot bergerak ke kanan	✓	-	✓	-
<i>E</i>	Robot bergerak diagonal kanan depan	✓	-	✓	-
<i>Q</i>	Robot bergerak diagonal kiri depan	✓	-	✓	-
<i>C</i>	Robot bergerak diagonal kanan belakang	✓	-	✓	-
<i>Z</i>	Robot bergerak diagonal kiri belakang	✓	-	✓	-
<i>L</i>	Robot bergerak rotasi kiri	✓	-	✓	-
<i>J</i>	Robot bergerak rotasi kanan	✓	-	✓	-
<i>K</i>	Robot menendang	✓	-	✓	-
<i>S</i>	Robot berhenti	✓	-	✓	-

Dari hasil pengujian yang telah dilakukan bahwa robot bisa dikoneksikan dengan menggunakan komputer dan laptop. Pengujian dengan menggunakan tombol *keyboard* ini sesuai dengan instruksi pergerakan robot.

### Pengujian Jarak Kontrol

Pengujian sistem secara keseluruhan ini adalah tahap akhir dalam proses pengujian pada robot. Pada saat robot pertama kali dihidupkan PC/laptop *controller* bertugas untuk mengkoneksikan jaringan ke robot melalui button *connect* yang terdapat pada *text box GUI controller*. Setelah robot dan PC/laptop *controller* sudah saling terhubung maka robot siap untuk dikontrol.

Pengujian ini dilakukan dengan cara bertahap dengan jarak pengontrolan pertama adalah 5 meter. Setelah pengujian pertama selesai, jarak pengontrolan akan terus ditambah 5 meter dan sehingga nantinya jarak pengontrolan mencapai jarak 100 meter.

Pengujian jarak pengontrolan robot dilakukan dengan menggunakan tiga buah laptop yang berbeda *type* dan dengan jarak pengontrolan yang sama. Pengujian pertama menggunakan laptop *type SONY VAIO*, pengujian kedua menggunakan laptop *notebook Acer* dan pengujian ketiga menggunakan laptop *ASUS*. Pengujian ini telah dilakukan berulang kali dan waktu eksekusi yang diterima oleh robot tidak jauh berbeda meskipun jarak terus ditambah dengan jarak 5 meter. Hasil pengujian jarak kontrol tanpa ada halangan dapat dilihat pada Tabel 4.

Tabel 4. Pengujian jarak kontrol

Jarak pengontrolan	Pengujian 1 (SONY VAIO)		Pengujian 2 (Notebook ACER)		Pengujian 3 (ASUS)	
	Berhasil	Gagal	Berhasil	Gagal	Berhasil	Gagal
5 Meter	✓	-	✓	-	✓	-
10 Meter	✓	-	✓	-	✓	-
15 Meter	✓	-	✓	-	✓	-
20 Meter	✓	-	✓	-	✓	-
25 Meter	✓	-	✓	-	✓	-
30 Meter	✓	-	✓	-	✓	-
35 Meter	✓	-	✓	-	✓	-
40 Meter	✓	-	✓	-	✓	-
45 Meter	✓	-	✓	-	✓	-
50 Meter	✓	-	✓	-	✓	-
55 Meter	✓	-	✓	-	✓	-
60 Meter	✓	-	✓	-	✓	-
65 Meter	✓	-	✓	-	✓	-
70 Meter	✓	-	✓	-	✓	-
75 Meter	✓	-	✓	-	✓	-
80 Meter	✓	-	✓	-	✓	-
85 Meter	✓	-	✓	-	✓	-
90 Meter	✓	-	✓	-	✓	-
95 Meter	✓	-	✓	-	✓	-

100 Meter	✓	-	✓	-	✓	-
-----------	---	---	---	---	---	---

Berdasarkan spesifikasi, jarak yang dapat dijangkau oleh *TP-Link Wireless N Router TL-WR841N* adalah 10-100 meter. Setelah melakukan pengujian jarak pengontrolan dengan jarak hingga 100 meter, penulis juga menguji jarak pengontrolan ini dengan jarak hingga 130 meter. Pengujian ini dilakukan dengan tujuan ingin mengetahui batas maksimal yang dapat dijangkau oleh *TP-Link Wireless N Router TL-WR841N*. Hasil pengujian jarak kontrol di atas 100 meter dapat dilihat pada Tabel 5.

Tabel 5. Pengujian jarak kontrol di atas 100 meter.

Jarak pengontrolan	Pengujian 1 (SONY VAIO)		Pengujian 2 (Notebook ACER)		Pengujian 3 (ASUS)	
	Berhasil	Gagal	Berhasil	Gagal	Berhasil	Gagal
105 Meter	✓	-	✓	-	✓	-
110 Meter	✓	-	✓	-	✓	-
115 Meter	✓	-	✓	-	✓	-
120 Meter	✓	-	✓	-	✓	-
121 Meter	✓	-	✓	-	✓	-
122 Meter	✓	-	✓	-	✓	-
123 Meter	✓	-	✓	-	✓	-
124 Meter	✓	-	-	✓	✓	-
125 Meter	✓	-	-	✓	✓	-
126 Meter	-	✓	-	✓	-	✓
127 Meter	-	✓	-	✓	-	✓
128 Meter	-	✓	-	✓	-	✓
129 Meter	-	✓	-	✓	-	✓
130 Meter	-	✓	-	✓	-	✓

Pada pengujian jarak kontrol dengan jarak di atas atau lebih dari 100 meter dilakukan dengan cara menambah jarak per 5 meter hingga mencapai jarak 130 meter. Setelah dilakukan beberapa kali percobaan pada pengujian kedua yaitu pengujian kontrol dengan menggunakan laptop *notebook Acer Aspire V5-132*, bahwa ketika pengiriman data instruksi ke robot pada jarak 125 meter robot sudah tidak bisa merespon data instruksi yang dikirim oleh laptop *controller* sedangkan untuk pengujian pertama yang menggunakan laptop *SONY VAIO* dan pengujian ketiga yang menggunakan laptop *ASUS* ketika pengiriman data instruksi pada jarak 125 meter robot masih bisa merespon dengan data instruksi yang dikirim. Jadi, penulis berinisiatif untuk mengurangi rentangan jarak kontrol dengan cara menambah jarak per 1 meter saja dari 120 meter hingga mencapai jarak 130 meter. Cara ini bertujuan agar mengetahui jarak pasti yang masih dapat dijangkau oleh *TP-Link N Router per-type* laptop itu berapa meter.

Terlihat pada Tabel 5. pengujian pertama dengan menggunakan laptop *SONY VAIO* dan pengujian ketiga dengan menggunakan laptop *ASUS* robot sudah tidak bisa merespon data instruksi yang dikirim oleh laptop *controller* pada saat jarak 126 meter. Sedangkan pada saat pengujian kedua, robot tidak bisa merespon data instruksi yang dikirim pada saat jarak 124 Meter. Jadi dapat di simpulkan bahwa jarak kontrol juga akan berpengaruh pada *type* laptop. Maksimal jarak yang dapat dijangkau oleh *TP-Link Wireless N Router* adalah 125 meter hanya berbeda atau lebih 25 meter saja dari spesifikasi. Jarak pengontrolan ini dilakukan robot berada di gedung elektro dan laptop *controller* berada di gedung C Politeknik Negeri Bengkalis. Ketika pengujian pada jarak 125 meter ke atas rata-rata robot sudah tidak bisa dikontrol lagi, karena karakter data yang dikirim dari laptop *controller* ke robot sudah tidak bisa dieksekusi oleh robot.

Kegagalan dalam pengontrolan robot merupakan suatu permasalahan yang biasanya terjadi pada saat proses pengontrolan berlangsung. Kegagalan yang dimaksud adalah dimana pada saat jaringan sedang mengalami gangguan yang menyebabkan pengiriman paket data yang dikirim dari PC/laptop *controller* ke *mini PC* robot menjadi tertunda. Namun ketika pada saat jaringan telah kembali stabil, paket data yang tertunda tadi akan masuk berurutan dan akan terus dieksekusi oleh robot sesuai dengan pengiriman instruksi yang dikirim dari PC/laptop *controller* tadi. Kegagalan dalam pengontrolan robot ini kemungkinan besar akan terjadi pada saat pengontrolan dari jarak jauh atau dengan jarak maksimal. Namun kegagalan ini juga akan bisa terjadi pada saat pengontrolan jarak dekat sekalipun.

## 5. KESIMPULAN DAN SARAN

Berdasarkan beberapa pengujian yang dilakukan dapat disimpulkan bahwa jarak maksimal yang dapat dijangkau oleh *TP-Link Wireless N Router TL-WR841N* rata-rata adalah 125 meter. Jika lebih dari 125 meter, maka paket data yang dikirim oleh PC/laptop *controller* tidak akan sampai atau robot sudah tidak bisa dikontrol lagi. Pengontrolan jarak juga akan berpengaruh dengan *type* PC/laptop. Semakin jauh jarak pengontrolan kemungkinan besar robot akan lambat menerima data yang dikirim oleh PC/laptop *controller*. Robot hanya bisa dikontrol melalui komunikasi *wireless* jaringan *wifi*, serta robot dan PC/laptop *controller* harus dalam alamat IP *address* yang sekelas. Gangguan pada jaringan terkadang membuat robot akan mengalami kegagalan dalam mengeksekusi data masuk. Kegagalan ini biasanya terjadi pada saat pengontrolan dengan batas jarak yang maksimal, namun juga akan terjadi pada saat pengontrolan jarak dekat sekali pun.

Adapun saran dalam penelitian ini adalah Menambahkan sensor kamera pada robot yang berfungsi untuk memantau atau *me-monitoring* robot dari aplikasi *controler* pada saat pengontrolan jarak yang terlalu jauh. Jarak pengontrolan ini masih terbatas yaitu 125 meter, maka dari itu penulis menyarankan kepada peneliti selanjutnya apabila ingin menambah jangkauan jarak pengontrolan, maka gunakanlah pengontrolan *wireless* melalui *radio frekuensi*. Robot sebaiknya dikontrol dengan menggunakan sistem *continuous*.

## 6. DAFTAR PUSTAKA

- Finnatara Bayu Samudra, Fernando Ardila, Setiawardhana. 2011. "Pengendalian *robot soccer* dalam mengambil bola dengan metode *evolutionary artificial potential field*". Politeknik Elektronika Negeri Surabaya (PENS). Jurusan Teknik Komputer. Surabaya.
- Giyanto, Dr. Ir. Aulia Siti Aisyah, MT., Suwito, ST.MT. 2008. "Perancangan *server* Menggunakan *Visual Basic 6.0* untuk Mendukung Sistem *Monitoring* dan Pengendalian *Autopilot* pada Kapal MCST-1". Institut Teknologi Bandung (ITB). Jurusan Teknik Elektro. Bandung.
- Gusti Arif Hanifah Pawitan, Alinda Nur Fitriyani, Lovila Arina Manasikana. 2016. "Rancang Bangun Robot Sepak Bola Beroda". Institut Teknologi Bandung (ITB). Jurusan Teknik Elektro. Bandung.
- Mannes, Poel. 2003. "*Motion Planning in A Robot Soccer System*", *A Master's Thesis in Computer Science. Language, Knowledge and Interaction Group Department of Compute Science University of Twente the Netherlands*.
- Muhammad Nurdin, Mohammad Adnan, Muhammad Luthfi Musra. 2016. "Penerapan Metode *Disturbance Observer* Pada Sistem Kontrol Pergerakan *Motor DC Robot Soccer*". Politeknik Negeri Ujung Pandang (PNUP). Jurusan Teknik Elektro. Makasar.
- Muhammad Rizki Setiawan, M. Aziz Muslim, Goegoes Dwi Nusantoro. 2012. "Kontrol Kecepatan *Motor DC* dengan Metode *PID* Menggunakan *Visual Basic 6.0* dan Mikrokontroler ATmega 16". Universitas Brawijaya (UB). Jurusan Teknik Listrik dan Mekatronika. Malang.

Purwanto, Fernando Ardila, Iwan Kurnianto Wibowo. 2016. “Implementasi *FPGA-Soft Processor* untuk Kontrol *Low Level* pada *Holonomic Robot*”. Politeknik Elektronika Negeri Surabaya (PENS). Surabaya.

Sahrul Nizam. 2016. “Desain Algoritma Pada Robot *Soccer Beroda*”. Politeknik Negeri Bengkalis. Jurusan Teknik Elektronika. Bengkalis.