

Sistem Komunikasi Otomatis Robot Sepak Bola Beroda dengan Komunikasi *Real-Time* melalui Infrastruktur Jaringan *Base Station*

Hendardi Yoga Pratama^{a,1,*}, Nuril Anwar^{b,2}

^{a,b} Program Studi Informatika, Universitas Ahmad Dahlan, Kabupaten Bantul, Daerah Istimewa Yogyakarta 55191, Indonesia

¹ hendardi.yogapratama2721@gmail.com *; ² nuril.anwar@tif.uad.ac.id

* Penulis Korespondensi

ABSTRAK

Ajang Kontes Robot Sepak Bola Indonesia Beroda adalah kompetisi robotika yang melibatkan kemampuan *softskill* dan *hardskill* mahasiswa dari seluruh Indonesia. Konfigurasi robot milik Universitas Ahmad Dahlan dilakukan menggunakan terminal di dalam sistem operasi Ubuntu yang membuat kinerja kurang efektif dan efisien. Bentuk konfigurasi yang tidak efektif adalah kesalahan konfigurasi untuk menggerakkan robot dan bentuk konfigurasi tidak efisien adalah durasi waktu konfigurasi oleh mahasiswa yang lama untuk memberikan perintah kepada robot. Fokus penelitian ini adalah pengembangan sistem komunikasi pada aplikasi *base station*.

Proses pengembangan sistem dilakukan menggunakan metode *waterfall*, yang mencakup tahapan analisis, desain, pengkodean, pengujian, dan pemeliharaan. Sistem komunikasi melalui WLAN pada *base station* menggunakan model *multicast* dan sistem komunikasi *Transmission Control Protocol (TCP)*. Pengembangan sistem pada robot beroda dan pembuatan *Graphical User Interface (GUI)* pada *base station* dilakukan oleh operator robot.

Penelitian ini berhasil membangun aplikasi *base station* yang berperan sebagai penghubung penting antara *referee box* wasit dan robot. Penggunaan aplikasi *base station* juga mampu menggunakan protokol TCP/IP dengan metode *multicast* secara efektif dalam sistem komunikasi antara komputer *referee box*, *base station*, dan robot menggunakan teknologi *wireless* dengan waktu transmisi rata-rata 0.052458 ms.

Riwayat Artikel

Diterima 10 Januari 2024
Diperbaiki 15 Februari 2024
Diterbitkan 20 Maret 2024

Kata Kunci

Aplikasi
Base Station
Robot



This is an open-access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license

1. Pendahuluan

Ajang Kontes Robot Sepak Bola Indonesia Beroda (KRSBI-B) adalah kompetisi robotika tahunan untuk mahasiswa. Acara ini diinisiasi oleh Kementerian Riset, Teknologi, dan Pendidikan Tinggi dengan harapan agar mendorong minat dan kemampuan pada mahasiswa di dalam mengembangkan teknologi terkini, khususnya di dalam bidang robotika. Divisi yang terbaru dalam kontes tersebut adalah Kontes Robot Sepak Bola Indonesia (KRSBI) Beroda [1].

Selama pertandingan berlangsung, robot menerima arahan dan perintah dari wasit melalui sebuah perangkat lunak bernama *referee box*. Setiap robot harus mampu memahami dan melaksanakan perintah yang diberikan oleh wasit sebagai pengatur pertandingan, seperti melakukan *Free Kick*, *Goal Kick*, *Kick Off*, dan perintah lainnya. Oleh karena itu, diperlukan dengan adanya sebuah sistem komunikasi yang berfungsi sebagai penghubung antara *referee box* dan robot agar robot dapat menerima serta memproses instruksi yang disampaikan dari *referee box*. Dalam hal ini, diperlukan sebuah *software base station* agar robot dapat menerima instruksi dengan baik dari *referee box* [2].

Berdasarkan penelitian yang dilakukan oleh Tjoanapessy pada tahun 2019, telah berhasil mengembangkan *software base station* yang memungkinkan komunikasi *real-time* antara *referee box* dan robot melalui aplikasi *real-time base station* [3]. Dalam konteks robot sepak bola beroda, pentingnya koordinasi antar robot ditekankan, yang dapat dicapai melalui pengambilan keputusan dan kendali yang tepat agar dapat dilakukan oleh satu robot dengan memanfaatkan data kondisi

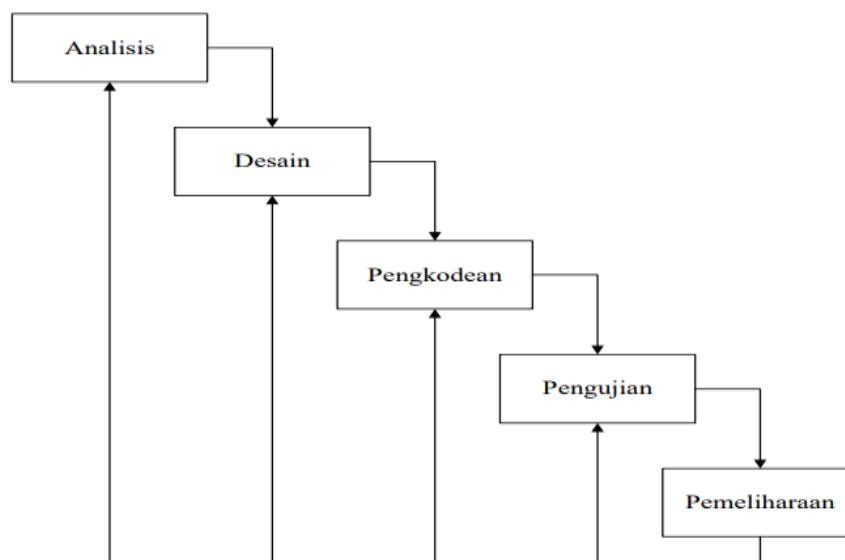
lapangan dari semua robot. Oleh karena itu, diperlukan sistem komunikasi data robot yang handal agar informasi dapat disampaikan secara efisien dan tepat waktu antar robot, memungkinkan mereka untuk berkolaborasi secara efektif selama pertandingan [4].

Pada Tim Robot Divisi KRSBI UAD memiliki permasalahan pada komunikasi robot beroda yaitu kurangnya efisiensi pada pengiriman data robot yaitu bagaimana cara agar tiga robot sebagai *client* dan satu *base station* sebagai *server* bisa berkomunikasi dengan efisien adalah dengan memastikan sistem komunikasi yang digunakan telah terintegrasi dengan baik. Namun, masalah muncul ketika tampilan komunikasi *base station* hanya tertampil di terminal Ubuntu, menyebabkan kendala dalam mengendalikan robot. Dalam rangka mengatasi kekurangan yang masih ada pada robot beroda, Oleh karena itu, komunikasi data dalam jaringan robot memegang peranan penting agar robot dapat bergerak secara otomatis dengan efisien. Diperlukan metode pengiriman *multicast* untuk menerima data dari *referee box*, serta proses pengiriman data dari robot ke *base station*. Integrasi sistem komunikasi data ini memungkinkan robot beroperasi dengan efektif dan koordinasi yang baik dalam menjalankan tugasnya [5].

Adanya integrasi GUI dalam sistem komunikasi memudahkan pengguna untuk mengontrol dan memantau kegiatan robot secara visual dan intuitif. Tampilan pada *base station* juga dapat ditingkatkan untuk memberikan informasi yang lebih jelas dan mudah dipahami. Implementasi sistem komunikasi harus mampu mendukung komunikasi dua arah antara *Referee Box* dan robot. Dengan adanya sistem ini, robot dapat menerima instruksi dari *Referee Box* dan menginterpretasikannya dengan akurat. Hal ini memastikan bahwa robot dapat merespons instruksi tepat waktu dan menjalankan tugasnya dengan efektif.

2. Metode

Pada penelitian ini, Metode *waterfall* digunakan untuk merancang dan mengembangkan sistem kendali otomatis pada robot sepak bola beroda dengan kemampuan komunikasi *real-time* melalui infrastruktur jaringan *base station*. Model *waterfall* adalah model sistematis berurutan dalam membangun *software* [6].



Gambar 1 Metode Waterfall

Berdasarkan Gambar 1, tahapan-tahapan pengembangan yang dilakukan dengan metode *waterfall*:

1. Analisis Kebutuhan. Pada tahapan ini dilakukan analisis untuk menemukan kebutuhan apa saja yang diperlukan dalam merancang sistem komunikasi robot dengan menggunakan *base station*.

2. Perancangan Sistem. Pada tahap ini, sistem mulai dirancang dengan menggunakan perangkat pemodelan sistem yaitu diagram alir data (*flowchart*). Langkah ini bertujuan untuk merancang struktur dan alur kerja sistem secara visual, memungkinkan pengembang untuk memahami secara jelas bagaimana setiap komponen sistem berinteraksi dan saling terhubung.
3. Penulisan Kode Program. Pada tahap ini dilakukan seorang *programmer* yang akan mengerjakan sistem yang telah dirancang. Dalam artian penggunaan komputer akan selalu dimaksimalkan dalam tahap ini.
4. Pengujian Sistem. Pada tahap ini di mana sistem diuji kemampuan dan keefektifannya sehingga didapatkan kekurangan dan kelemahan pada sistem yang kemudian dilakukan pengkajian ulang dan perbaikan terhadap aplikasi menjadi lebih baik dan sempurna.
5. Pemeliharaan. Pada tahap ini perangkat lunak akan terus dipelihara untuk memastikan bahwa sistem tetap berjalan dengan baik dan optimal dalam jangka panjang.

2.1. Metode Pengumpulan Data

2.1.1. Studi Literatur

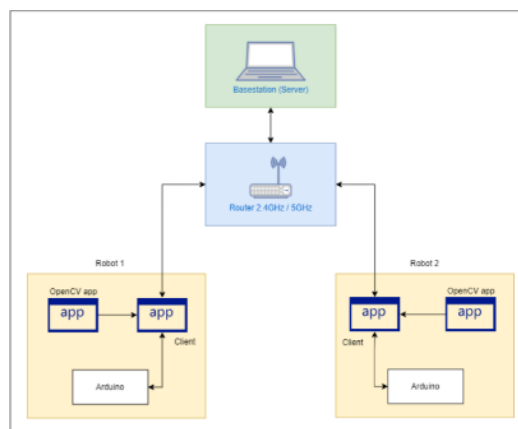
Melakukan tinjauan literatur untuk mengumpulkan informasi dilakukan studi literatur mendalam tentang penelitian terkait yang telah dilakukan sebelumnya, terkait dengan sistem kendali otomatis robot beroda, komunikasi *real-time* melalui infrastruktur jaringan *base station*, serta teknologi dan metode terkait lainnya. Tujuannya adalah untuk memperoleh pemahaman yang komprehensif tentang konsep dan teknologi yang relevan.

2.1.2. Observasi

Metode observasi langsung memungkinkan pengamatan langsung terhadap robot sepak bola selama pertandingan atau sesi latihan, dengan merekam gerakan robot, respons terhadap perintah, dan interaksi dengan lingkungan dan robot lainnya. Metode ini memberikan data *real-time* yang penting untuk menganalisis performa robot.

2.2. Sistem Desain

Pada rancangan sistem komunikasi, terdapat *Referee Box* yang bertindak sebagai *server* yang mengirimkan informasi ke *Base Station*. Pengiriman data dilakukan melalui jaringan nirkabel pada frekuensi 2.4GHz/5GHz [7]. *Base Station* berfungsi sebagai penerima dan pengirim data, serta mengelola pergerakan robot dan strategi permainan. Data dapat dikirim ke [8]. Pada penelitian ini, *Base Station* menggunakan protokol TCP/IP dengan pendekatan *Multicast*.

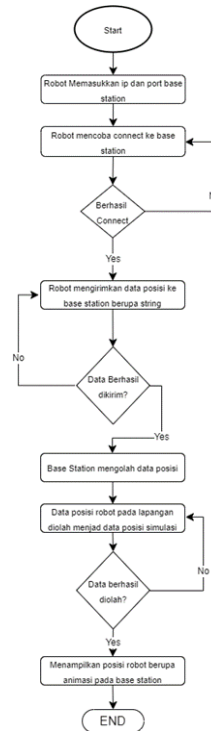


Gambar 2 Rancangan Desain Komunikasi

Pada Gambar 2 mengilustrasikan alur pengiriman data pada sistem, dimulai dari *referee box* sebagai pusat komunikasi yang mengirimkan intruksi ke Robot 1 dan Robot 2 melalui protokol koneksi tertentu. Instruksi diterima dari *base station*, yang bertindak sebagai penghubung antara *referee box* dengan robot 1 dan robot 2. Robot 1 dan robot 2 menggunakan arduino sebagai perangkat keras untuk menerima instruksi.

2.2.1. Flowchart Sistem Robot

Metode ini digambarkan pada Gambar 2. Prosesnya dimulai dengan robot memasukkan alamat IP dan *port* dari *base station* sebagai parameter koneksi [9]. Selanjutnya, robot mencoba terhubung ke *base station* melalui jaringan yang tersedia. Setelah koneksi berhasil, *base station* mengirimkan data ke robot berupa karakter "1" sebagai perintah pergerakan. Robot menerima perintah tersebut dan melakukan gerakan sesuai dengan instruksi yang diberikan. Kemudian robot mengirimkan informasi lokasi sebenarnya ke *base station* untuk memberikan umpan balik tentang lokasinya.



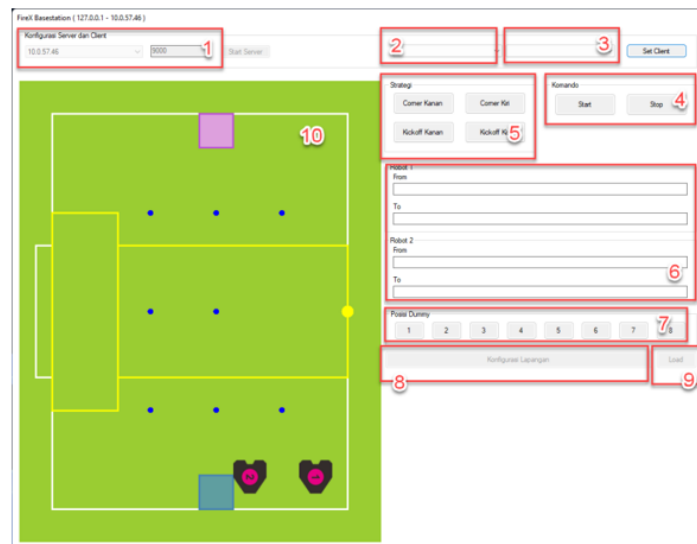
Gambar 3 Flowchart Sistem Robot

Pada Gambar 3 *Base Station* menerima 'W' jika berhasil terhubung ke kotak wasit, memintanya untuk menunggu data *kick-off* awal. Setelah menerima 'k', tim magenta memulai serangan, sementara 'K' memberi sinyal kepada tim *cyan* untuk mengadopsi strategi bertahan. Tanpa data *kick-off*, *Base Station* dengan sabar menunggu. Setelah mendapatkan data *kick-off* tim, *Base Station* menunggu perintah mulai; setelah menerima, *Base Station* menjalankan strategi yang sesuai. Jika kotak wasit mengirimkan data berhenti, *Base Station* merespons dengan 's' ke robot; jika tidak, ia menunggu sinyal berhenti.

3. Hasil dan Pembahasan

3.1. Desain Base Station Server

Struktur desain tampilan yang digunakan akan dibagi menjadi dua bagian utama, yaitu tampilan utama yang meliputi visualisasi lapangan dan robot, konfigurasi *server*, konfigurasi *client*, kontrol dan strategi serta protokol antar robot, yang kedua adalah tampilan konfigurasi lapangan. Visualisasi lapangan *base station* dapat dilihat pada Gambar 4.



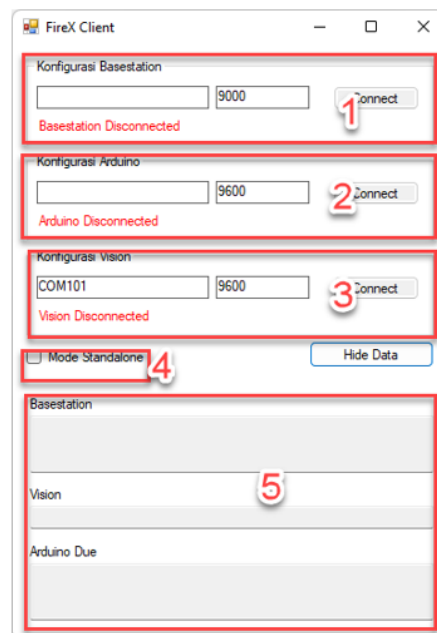
Gambar 4 Base Station Server

Pada Gambar 4 terdapat beberapa bagian, yaitu:

1. Konfigurasi *server* digunakan untuk memilih IP dan *port* yang digunakan sebagai *server*.
2. Konfigurasi robot klien 1 untuk mengatur Alamat IP robot 1.
3. Konfigurasi robot klien 2 untuk mengatur Alamat IP robot 2.
4. Tombol kontrol pergerakan robot.

3.2. Base Station Client

Desain dari tampilan aplikasi *client* sangatlah sederhana, dikarenakan aplikasi hanya digunakan sebagai sarana untuk meneruskan data ke Arduino maupun *base station*, disertai beberapa *textbox* yang digunakan untuk memantau arus transaksi data secara *real-time*. Berikut Gambar 5 adalah tampilan aplikasi *client*.



Gambar 5 Base Station Client

Bagian-bagian dalam aplikasi klien seperti yang dapat dilihat pada Gambar 5. Klien *Base Station* yaitu:

1. Koneksi ke *Base Station*.
2. Koneksi ke Arduino.
3. Mode Standalone, hanya akan mengeluarkan data dari *vision*, sedangkan data lainnya berupa data *dummy*.
4. Tampilan data.

3.3. Rancang Lapangan pada Visualisasi Lapangan

Rancangan lapangan yang dibuat lapangan dengan ukuran 6 meter panjang dan 4,5 meter lebar, serta gawang 3 meter panjang dan 1 meter lebar, akan disesuaikan ke dalam Visual Studio dengan konversi $1,486 \text{ cm} = 1 \text{ pixel}$ untuk memastikan representasi visual yang akurat.



Gambar 6 Visualisasi Lapangan

Pada gambar tersebut terdapat beberapa bagian diantaranya adalah:

1. Tipe lapangan, untuk mengganti desain lapangan, untuk regional dan nasional.
2. Konfigurasi ukuran, untuk mengatur ukuran lapangan menggunakan ukuran asli dalam satuan m.
3. Konfigurasi ikon robot, untuk mengganti ikon dari tiap robot.
4. Konfigurasi robot aktif dan sisi lapangan, untuk mengatur robot apa saja yang mau diaktifkan dan juga sisi lapangan mana robot akan bertanding.

3.4. Pengujian Komunikasi

Pada pengujian komunikasi ini, *user interface* (GUI) pada *Base Station* dibuat dengan menggunakan aplikasi *Microsoft Visual Studio*. Metode komunikasi antara *Base Station* dengan *Referee Box* menggunakan pendekatan *multicast* dengan protokol TCP/IP. Pengujian aplikasi komunikasi antara *Referee Box*, *Base Station*, dan Robot.

3.4.1. Pengujian Sistem Komunikasi dengan Software Wireshark

Pengujian sistem komunikasi menggunakan aplikasi *Wireshark* dilakukan dengan tujuan untuk memonitoring aliran data, termasuk informasi terkait sumber IP, tujuan IP, protokol yang digunakan, dan meninjau isi data yang dikirim. Hasil pengamatan pada *Wireshark* dijelaskan melalui Gambar 7.

No.	Time	Source	Destination	Protocol	Length	Info
5171	119.024667	192.168.0.100	192.168.0.100	TCP	114	9000 → 60666 [PSH, ACK] Seq=98826 Ack=1 Win=4100 Len=60
5172	119.071396	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98886 Win=254 Len=0
5173	119.071512	192.168.0.110	192.168.0.100	TCP	84	9000 → 60666 [PSH, ACK] Seq=98886 Ack=1 Win=4100 Len=30
5174	119.118494	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98116 Win=254 Len=0
5175	119.118622	192.168.0.110	192.168.0.100	TCP	114	9000 → 60666 [PSH, ACK] Seq=98116 Ack=1 Win=4100 Len=60
5176	119.196481	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98176 Win=254 Len=0
5177	119.196484	192.168.0.110	192.168.0.100	TCP	99	9000 → 60666 [PSH, ACK] Seq=98176 Ack=1 Win=4100 Len=45
5178	119.244484	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98221 Win=253 Len=0
5179	119.244593	192.168.0.110	192.168.0.100	TCP	99	9000 → 60666 [PSH, ACK] Seq=98221 Ack=1 Win=4100 Len=45
5180	119.290100	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98266 Win=253 Len=0
5181	119.290189	192.168.0.110	192.168.0.100	TCP	99	9000 → 60666 [PSH, ACK] Seq=98266 Ack=1 Win=4100 Len=45
5182	119.336940	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98311 Win=253 Len=0
5183	119.337020	192.168.0.110	192.168.0.100	TCP	114	9000 → 60666 [PSH, ACK] Seq=98311 Ack=1 Win=4100 Len=60
5184	119.383774	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98371 Win=253 Len=0
5185	119.383845	192.168.0.110	192.168.0.100	TCP	99	9000 → 60666 [PSH, ACK] Seq=98371 Ack=1 Win=4100 Len=45
5187	119.459955	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98416 Win=253 Len=0
5188	119.431061	192.168.0.110	192.168.0.100	TCP	84	9000 → 60666 [PSH, ACK] Seq=98416 Ack=1 Win=4100 Len=30
5189	119.477583	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98446 Win=253 Len=0
5190	119.477682	192.168.0.110	192.168.0.100	TCP	99	9000 → 60666 [PSH, ACK] Seq=98446 Ack=1 Win=4100 Len=45
5191	119.540173	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98491 Win=252 Len=0
5192	119.540275	192.168.0.110	192.168.0.100	TCP	99	9000 → 60666 [PSH, ACK] Seq=98491 Ack=1 Win=4100 Len=45
5194	119.602465	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98536 Win=252 Len=0
5195	119.602542	192.168.0.110	192.168.0.100	TCP	99	9000 → 60666 [PSH, ACK] Seq=98536 Ack=1 Win=4100 Len=45
5196	119.680680	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98581 Win=252 Len=0
5197	119.680689	192.168.0.110	192.168.0.100	TCP	129	9000 → 60666 [PSH, ACK] Seq=98581 Ack=1 Win=4100 Len=75
5198	119.727417	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98656 Win=252 Len=0
5199	119.727500	192.168.0.110	192.168.0.100	TCP	99	9000 → 60666 [PSH, ACK] Seq=98656 Ack=1 Win=4100 Len=45
5200	119.774588	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98701 Win=252 Len=0
5201	119.774695	192.168.0.110	192.168.0.100	TCP	99	9000 → 60666 [PSH, ACK] Seq=98701 Ack=1 Win=4100 Len=45
5202	119.836761	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98746 Win=251 Len=0
5203	119.836836	192.168.0.110	192.168.0.100	TCP	84	9000 → 60666 [PSH, ACK] Seq=98746 Ack=1 Win=4100 Len=30
5204	119.883642	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98776 Win=251 Len=0
5205	119.883717	192.168.0.110	192.168.0.100	TCP	99	9000 → 60666 [PSH, ACK] Seq=98776 Ack=1 Win=4100 Len=45
5206	119.930728	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98821 Win=251 Len=0
5207	119.930821	192.168.0.110	192.168.0.100	TCP	99	9000 → 60666 [PSH, ACK] Seq=98821 Ack=1 Win=4100 Len=45
5208	119.977397	192.168.0.100	192.168.0.110	TCP	54	60666 → 9000 [ACK] Seq=1 Ack=98866 Win=251 Len=0

Gambar 7 Monitoring Wireshark

Pada Gambar 7 terdapat 16 frame yang dapat dianalisis untuk melihat waktu pengiriman dan penerimaan data, IP source, IP destination, protokol yang digunakan, panjang data dalam byte, dan port yang digunakan. Pada protokol TCP terdapat flag seperti ACK untuk mengetahui status penerimaan data pada komputer client, dan PSH untuk menandakan isi TCP yang diterima pada komputer client.

3.4.2. Pengujian Waktu Transmisi

Pengujian waktu transmisi ini bertujuan untuk mengukur durasi rata-rata proses transmisi atau delay data yang dikirim dari Base Station. Dari Gambar 7. dapat dihitung durasi waktu transmisi untuk setiap perintah yang dikirim dari Base Station ke robot.

Untuk menghitung waktu transmisi, dapat menggunakan rumus [10]:

$$t = \text{waktu diterima (ms)} - \text{waktu dikirim (ms)}$$

Berdasarkan monitoring command pada Base Station pada Gambar 7 dapat dilihat waktu transmisi untuk setiap paket yang dikirim. Tabel pengujian waktu transmisi dapat dilihat pada Tabel 1.

Tabel 1 Pengujian Waktu Transmisi

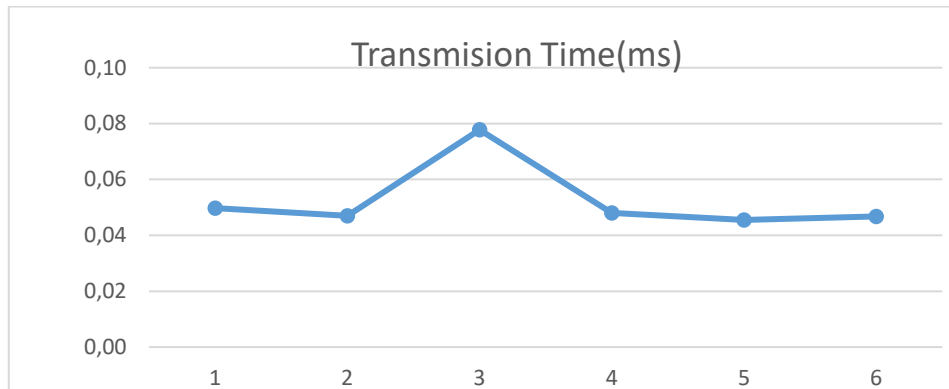
Tes ke-	Waktu Dikirim (ms)	Waktu Diterima (ms)	Waktu rata-rata (ms)	Paket Data yang Diterima
1	119,024667	119,071396	0,049729	Start
2	119,071512	119,118494	0,046982	Stop
3	119,118622	119,196401	0,077779	Corner Kanan
4	119,196484	119,244484	0,048000	Corner Kiri
5	119,244593	119,290100	0,045507	KickOff Kanan
6	119,290189	119,336940	0,046751	KickOff Kiri
Waktu rata-rata transmisi			0,052458	

Berdasarkan Tabel 1, waktu transmisi rata-rata atau delay dapat dihitung dengan persamaan:

$$\text{Rata – rata waktu transmisi} = \frac{\text{Total waktu transmisi}}{\text{Jumlah paket data yang dikirim}}$$

$$\text{Rata – rata waktu transmisi} = \frac{0,314748}{6} = 0,052458$$

Dari Tabel 1, gambar grafik rata-rata transmisi atau *delay* dapat dibuat seperti yang ditunjukkan pada Gambar 8.



Gambar 8 Transmisi Waktu

Pada Gambar 8 dapat dilihat bahwa waktu transmisi atau *delay* terlama saat pengiriman data uji ke-1 adalah 0,049729 ms, sedangkan waktu transmisi atau *delay* tercepat saat pengiriman data uji ke-5 adalah 0,045507 ms.

4. Kesimpulan

Hasil penelitian menunjukkan bahwa berhasil dikembangkan aplikasi *base station* yang berperan sebagai penghubung penting antara *Referee Box* wasit dan robot, serta penggunaan protokol TCP/IP dengan metode *multicast* efektif dalam sistem komunikasi antara komputer *Referee box*, *base station*, dan robot menggunakan teknologi *wireless* dengan waktu transmisi rata-rata 0.052458 ms.

Daftar Pustaka

- [1] R. Agitasani and R. Dwi, "Implementation of Base Station communication systems on wheels football robots," vol. 4, no. 2, pp. 49–57, 2022.
- [2] F. Tjoanapessy *et al.*, "Aplikasi Base Station Untuk Robot Sepak Bola Beroda," *J. Tek. Inform.*, vol. 14, no. 3, pp. 285–290, 2019.
- [3] K. Edy Surya Prabowo, Y. Divayana, and P. Rahardjo, "Perancangan Aplikasi Base Station Dalam Sistem Koordinasi Robot Sepak Bola Beroda Dengan Multi Thread," *J. SPEKTRUM*, vol. 9, no. 4, p. 17, 2022, doi: 10.24843/spektrum.2022.v09.i04.p3.
- [4] D. Dwiyanto, N. Fath, R. Sepak, and B. Beroda, "Analisa Sistem Komunikasi Data Pada," vol. 3, no. 2, pp. 419–430, 2020.
- [5] A. Makinun Amin, J. Sahertian, and A. Sanjaya, "Perancangan Sistem Komunikasi Data Robot Sepak Bola Dalam Kontes Robot Sepak Bola Indonesia Beroda (Krsbi)," 2019.
- [6] A. Nurseptaji, "Implementasi Metode Waterfall Pada Perancangan Sistem Informasi Perpustakaan," *J. Dialekt. Inform.*, vol. 1, no. 2, pp. 49–57, 2021, doi: 10.24176/detika.v1i2.6101.
- [7] Y. Rahmawati and N. S. Widodo, "Implementation of Wireless Communication System in R-SCUAD Humanoid Soccer Robot with Checksum Error Detection Method Based on UDP Protocol," vol. 4, no. 3, pp. 194–203, 2023, doi: 10.12928/biste.v4i3.5402.
- [8] W. Firmansyah, J. Sahertian, and J. Sulaksono, "Implementasi Fitur Manual Keyboard Menggunakan Header Pada Basestation Robot Sepak Bola Beroda Abimanyu," *Semin. Nas. Inov. Teknol.*, pp. 247–252, 2022.
- [9] M. A. Fouk, H. Hudiono, and A. Hariyadi, "Implementation of Base Station as An Intermediary Referee Box in the delivery of Wheeled Football Robot Movement Commands,"

Jartel, vol. 12, no. 3, pp. 114–120, 2022, doi: 10.33795/jartel.v12i3.314.

- [10] O. Prasetyo and R. Dwi Puriyanto, “Wheeled robot automated system through communication with Referee Box using Base Station,” *Signal Image Process. Lett.*, vol. 4, no. 1, pp. 11–19, 2022, [Online]. Available: <https://simple.ascee.org/11https://simple.ascee.org/simple@ascee.org>.