

Analisis Delay Pada Prototype Alat Running Text LRT JABODEBEK Menggunakan Software Wireshark

Gita Angraini¹, Vian Priandhika², Sulis Setiowati³, Elitaria Bestri Agustina Siregar⁴

Instrumentasi dan Kontrol Industri, Teknik Elektro, Politeknik Negeri Jakarta, Jl. Prof. DR. G.A. Siwabessy, Kampus
Universitas Indonesia Depok, Kota Depok, 16425, Indonesia

E-mail:

gita.angraini.te20@mhs.w.pnj.ac.id¹, vian.priandhika.te20@mhs.w.pnj.ac.id², sulis.setiowati@elektro.pnj.ac.id³, elitaria.gustina@elektro.pnj.ac.id⁴

Abstrak

Penelitian ini menggunakan hasil data yang diperoleh dari *prototype* komunikasi data pada kereta LRT JABODEBEK. Performa dari suatu alat sistem komunikasi yang baik dapat ditentukan dari beberapa faktor, salah satunya *delay*. Besaran nilai *delay* yang baik untuk sistem komunikasi memiliki angka yang cukup. Penelitian ini dilakukan dengan tujuan menganalisis *delay* untuk mengetahui bagaimana indeks *delay* yang dihasilkan pada *prototype*. Tujuan yang hendak dicapai pada penelitian ini antara lain untuk mengetahui seberapa besar pengaruh *delay* terhadap kinerja sistem dan dapat memberikan saran agar sistem dapat bekerja lebih baik. Jenis metode penelitian yang digunakan adalah penelitian pengembangan atau Research and Development (R&D) dengan mengembangkan sistem *running text* P4 pada LRT agar lebih efisien, dengan meminimalisir komponen dalam pembuatan sistem komunikasinya. Hasil yang di dapat dari perhitungan *delay* sebesar 292,856 ms dengan rata – rata *delay* 7,321 ms. Dengan persentase data yang memiliki *indeks* baik 85% yang memiliki rentang nilai 0 s/d 300 ms. dan *indeks* buruk sebesar 12,5% dengan rentang nilai sebesar <450 ms.

Keywords: Delay, Running Text, Prototype

Abstrak

This research uses data obtained from a data communication prototype on the JABODEBEK LRT train. The performance of a good communication system tool can be determined from several factors, one of which is delay. A good delay value for a communication system has a sufficient number. This research was carried out with the aim of analyzing delay to find out how the delay index is produced on the prototype. The objectives to be achieved in this research include finding out how much influence delay has on system performance and being able to provide suggestions so that the system can work better. The type of research method used is research and development (R&D) by developing the P4 running text system on the LRT to make it more efficient, by minimizing components in making the communication system. The results obtained from the delay calculation were 292,856 ms with an average delay of 7,321 ms. With a percentage of data that has a good index of 85% with a value range of 0 to 300 ms and a bad index of 12.5% with a value range of <450 ms.

Keywords: Delay, Running Text, Prototype

I. Pendahuluan

Kereta merupakan salah satu moda transportasi publik yang secara umum digunakan masyarakat perkotaan di berbagai negara. Di Jakarta, jenis moda transportasi publik ini sudah mengalami kemajuan dimulai dari KRL (Kereta Rel Listrik) Commuter Line, lalu MRT (Mass Rapid Transit/Moda Raya Terpadu) yang akan beroperasi tahun 2019, dan LRT (Light Rail Transit) yang kemudian menyusul dengan target operasi pada 2019[1][2].

Di dalam *system* kereta api, khususnya pada LRT (*Light Rail Transit*), terdapat banyak *system*. Salah satunya adalah PAS (*Public Adress System*). PAS merupakan amplifikasi suara elektronik dan sistem distribusi dengan mikrofon amplifier dan penguat suara yang dapat digunakan oleh individu untuk menjangkau komunikasi kepada publik dengan area yang luas, misalnya untuk pengumuman di berbagai ruang publik seperti stasiun KA, terminal, tempat perbelanjaan, sekolah, layanan publik dan sebagainya[3][4].

Saat ini komunikasi antar *running text* pada

kereta LRT menggunakan beberapa *controller*, yaitu Arduino yang telah diprogram dan selanjutnya akan diteruskan ke dalam STM32 yang akan menampilkan kode program tersebut ke *running text*. Tentu saja hal tersebut menjadi kurang efisien dalam penerapan sistem komunikasi pada kereta LRT, karena saat ini *system* yang digunakan pada kereta LRT menggunakan 1 buah *microcontroller* yaitu Arduino Uno sebagai master nya dan 1 buah *microcontroller* STM 32 yang berada dalam satu PCB (*Printed Circuit Board*) pada tiap panel *running text* P4 berukuran 16x32 pixel.

Beberapa peneliti telah melakukan upaya peningkatan efisiensi sistem komunikasi pada kereta.

Pada penelitian Emil Naf'an (2019) yang bertujuan untuk menguji akurasi sistem penjadwalan sholat digital menggunakan Arduino sebagai pengendali, ditemukan bahwa pembuatan sistem tersebut berjalan dengan baik dan mendapat hasil akurasi yang baik tetapi mengalami kendala dengan penambahan program kalibrasi karena jadwal sholat terhambat ± 3 detik[5]

Penelitian Abd Wahid, Syahrir Abdussamad, dan Iskandar Z.nasibu bertujuan untuk merancang aplikasi android *update data running text* pada panel LED matrix P10 perangkat smartphone dengan aplikasi via bluetooth dengan metode eksperimen. Tahapan pembuatan alat menggunakan bluetooth HC 05, sensor suhu (DHT11) untuk ditampilkan pada papan informasi display. Hasil penelitian ini adalah aplikasi android dengan pengontrolan bluetooth, dinilai dapat meningkatkan kecepatan dan efisiensi saat penggantian text pada papan display LED matrix 16x160. Rata-rata kecepatan penggantian text bisa dicapai 1,0 detik secara real time dengan jarak < 15 meter, dan mampu menampung maksimal 255 karakter[6].

Penelitian Nanik Triwahyuni dan Samuel Beta (2022) berfokus pada perancangan sistem informasi *running text/running text* berbasis internet dengan menggunakan mikrokontroler ESP32 serta papan display (Panel LED Display P10). Berdasarkan analisis kebutuhan, dapat dihasilkan sebuah sistem informasi berbasis *running text* yang dapat dikendalikan dari jarak jauh menggunakan *microcontroller* ESP 32[7].

Chandra Wiharya, Priya Surya harjianto, Galuh Prawestri C.H, Lukman Hakim, dan Dhimas Dhesah Kharisma (2023) merancang sebuah sistem berbasis *running text* yang memiliki tujuan untuk menjadikan daya tarik dan minat konsumen cafe yang melihat tampilan dari sistem tersebut. Pengujian tersebut melibatkan beberapa tahap pengujian yaitu, uji fungsionalitas, keandalan dan antarmuka. Hasil dari penelitian tersebut memberikan hasil implementasi dari penampil informasi elektronik *running text* berbasis WiFi menggunakan HD-WF1 sebagai pengolah data. Rangkaian LED dapat memberikan informasi penting yaitu, menu, kegiatan dan tujuan sponsor kegiatan pada pemea kafe kampung wisata mbesuk[8].

Saat ini, sistem *running text* masih menggunakan dua mikrokontroler pada tiap *running text*.

Hal ini menyebabkan borosnya pemakaian mikrokontroler yang digunakan. Kerusakan yang terjadi pada sistem ini terjadi karena waktu pemakaian atau *durability* dari komponen yang digunakan salah satunya PCB (*printed circuit board*). Untuk mengganti salah satu komponen yang ada pada PCB harus melakukan beberapa penyesuaian atau konfigurasi IP ulang dan memerlukan waktu dan proses yang cukup lama pada saat *maintenance*.

Peningkatan efisiensi komunikasi antar *running text* pada kereta LRT perlu terus dikembangkan. Oleh karena itu, penelitian ini bertujuan untuk meng-efisienkan mikrokontroler dengan membuat sistem komunikasi display *running text* menggunakan arduino mega sebagai master dan beberapa ESP32 sebagai slave. Protokol komunikasi yang digunakan adalah modbus RS-485. Modbus RS-485 dipilih karena apabila terjadi kerusakan dan akan dilakukan proses *maintenance* tanpa perlu mengubah konfigurasi IP karena sudah menggunakan komunikasi serial.

Salah satu permasalahan dalam komunikasi data ialah *delay*. *Delay* adalah suatu tindakan yang menunda, menghalangi, atau bahkan menyebabkan sesuatu terjadi lebih lambat dari biasanya dari waktu yang telah ditentukan. *Delay* dapat disebabkan oleh berbagai faktor, seperti jarak, media fisik rendahnya kecepatan atau gangguan teknis.

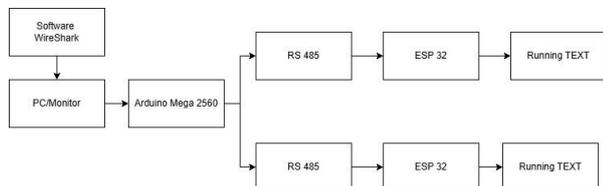
Arduino mega adalah jenis board *microcontroller* yang dikembangkan oleh arduino.cc. Arduino MEGA memiliki 54 pin digital dan 16 pin analog, sehingga lebih banyak dibandingkan dengan seri lain dari Arduino. Hal ini memungkinkan untuk mengakses lebih banyak perangkat eksternal dalam suatu kebutuhan. Arduino MEGA memiliki 256 KB memori flash yang dapat digunakan untuk menyimpan program, sehingga lebih besar dibandingkan dengan seri lain[9].

Mikrokontroler board yang dipilih untuk modul ini berbasis ESP32. ESP32 merupakan mikrokontroler SoC (System on Chip) terpadu dengan WiFi 802.11 b/g/n, Bluetooth versi 4.2, dan berbagai periferal. Chip ini menggunakan mikroprosesor 32 bit Xtensa LX6 dual-core. Kelebihan utama mikrokontroler ini ialah harganya yang relatif murah, mudah diprogram, memiliki jumlah pin I/O yang memadai, serta memiliki adapter WiFi internal untuk mengakses jaringan Internet[10].

Software wireshark banyak digunakan untuk menangkap lalu lintas jaringan secara real time. Hal ini membantu untuk menganalisis paket data yang menyimpang dalam jaringan untuk memberikan informasi yang berguna terkait kinerja, pemeliharaan, dan keamanan jaringan[11].

II. Metode Penelitian

Perancangan alat dilakukan Penelitian ini menggunakan metode eksperimen dengan alur sebagai berikut:

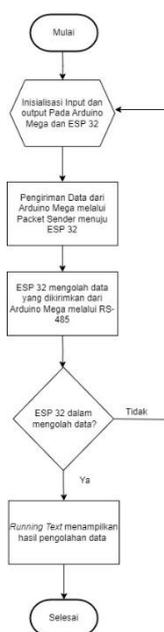


Gambar 1. Diagram Blok Alat

Sistem komunikasi dari *prototype* ini bekerja menggunakan protokol UDP (*User Datagram Protocol*). Data yang masuk melalui PC/Monitor yang akan dikirimkan ke Arduino Mega dan diteruskan ke dalam ESP 32 melalui serial RS-485 lalu ditampilkan pada panel *running text* P4. Kegunaan dari *software wireshark* ini untuk mendeteksi lalu lintas jaringan data yang dikirimkan selama proses pengiriman data berlangsung. Dalam skema tersebut, *software wireshark* akan menampilkan beberapa data UDP (*User Datagram Protocol*) yang mengalami kejadian *delay* selama proses berlangsung.

Jenis penelitian yang digunakan adalah penelitian pengembangan atau Research and Development (R&D). Alur rancangan penelitian ini mengacu pada Sugiyono (2008) dengan modifikasi yang diselaraskan dengan tujuan dan kondisi penelitian.

Perancangan alat ini dilakukan di *warehouse* PT. Restek yang berlokasi di daerah Jatimulya, Bekasi, Jawa Barat. Perancangan dan uji coba alat ini kami lakukan dengan estimasi 2-3 bulan.



Gambar 2. Flow Chart Kerja Alat

Program diawali dengan inisialisasi input berupa data dari PC/Laptop yang akan dikirimkan ke arduino mega dan output berupa tampilan *display running text*. Data input dari PC/Laptop akan dikirim ke arduino mega lalu diolah agar dapat diproses ke ESP 32. Pengiriman data menggunakan komunikasi data modbus RS485 sehingga dapat mengirim langsung ke beberapa *slave* (ESP32). Jika data berhasil dikirim maka akan langsung menghasilkan output berupa tampilan pada *running text*.

Desain alat yang perlukan untuk *prototype* ini yaitu menggabungkan 2 panel *running text* P4 menjadi 1 *running text*. Dan pembuatan PCB (*Printed Circuit Board*) yang dipakai sesuai kebutuhan diletakkan pada bagian belakang panel *running text* P4. Perancangan ini dibuat agar *prototype* sampai menjadi alat cukup saat ditempatkan di pada kereta LRT.

III. Hasil dan Pembahasan

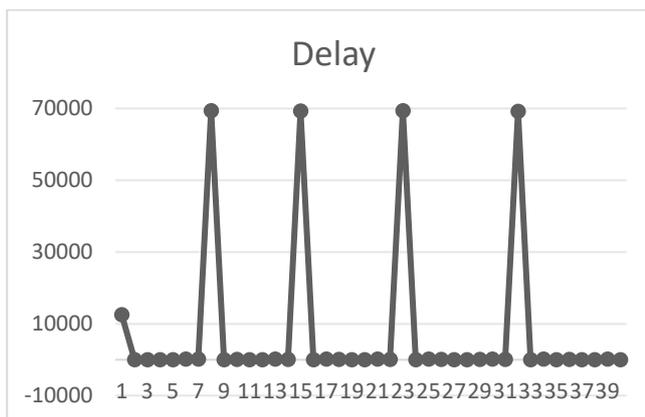
Suatu penelitian akan memperoleh hasil sesuai dengan tujuan yang ditetapkan. Tujuan penelitian ini untuk mengetahui *delay* yang didapatkan saat pengujian fungsional alat yang berupa *prototipe* terlebih dahulu. Data hasil penelitian *delay* didapatkan dengan menggunakan *software wireshark*, dari hasil pengukuran didapat data sebagai berikut :

Tabel 1. Data Hasil Perhitungan Delay

No.	Protocol	Time	Delay (s)
1		0,000	12,597
2	UDP	12,597	0,002
3	UDP	12,599	0,092
4	UDP	12,691	0,056
5	UDP	12,747	0,006
6	UDP	12,753	0,198
7	UDP	12,951	0,181
8	UDP	13,132	69,394
9	UDP	82,527	0,006
10	UDP	82,533	0,094
11	UDP	82,627	0,051
12	UDP	82,678	0,004
13	UDP	82,682	0,200
14	UDP	82,883	0,076
15	UDP	82,959	69,352
16	UDP	152,311	0,011
17	UDP	152,323	0,200
18	UDP	152,522	0,122

19	UDP	152,644	0,049
20	UDP	152,693	0,003
21	UDP	152,696	0,191
22	UDP	152,887	0,065
23	UDP	152,952	69,364
24	UDP	222,315	0,008
25	UDP	222,323	0,228
26	UDP	222,552	0,080
27	UDP	222,631	0,015
28	UDP	222,646	0,026
29	UDP	222,672	0,117
30	UDP	222,789	0,208
31	UDP	222,997	0,126
32	UDP	223,123	69,195
33	UDP	292,319	0,020
34	UDP	292,338	0,189
35	UDP	292,527	0,009
36	UDP	292,536	0,070
37	UDP	292,606	0,054
38	UDP	292,660	0,005
39	UDP	292,665	0,192
40	UDP	292,856	-
		Total Delay	292,856
		Rata - Rata Delay	7,321404

Grafik 1. Pembacaan Hasil Perhitungan Delay



Grafik ini menggambarkan hasil perhitungan *delay* yang terjadi dalam suatu sistem jaringan. Sumbu horizontal (sumbu X) mempresentasikan hasil perhitungan dalam satuan milidetik(ms), dan sementara sumbu vertikal (sumbu Y) mempresentasikan berapa banyak data yang diambil untuk proses perhitungan.

Pada awal perhitungan, kita dapat melihat bahwa *delay* berada pada level rendah yaitu diantara data ke-1 sampai data ke-7. Pada data ke-8 mengalami kenaikan yang cukup tinggi karena mengalami hambatan yang signifikan yang mempengaruhi waktu pengiriman data.

Namun, pada titik waktu tertentu, *delay* tersebut telah mencapai puncak nya yaitu pada 69195 m/s dan *delay* mengalami penurunan yang cepat dan responsif dan dapat kembali keperforma awal.

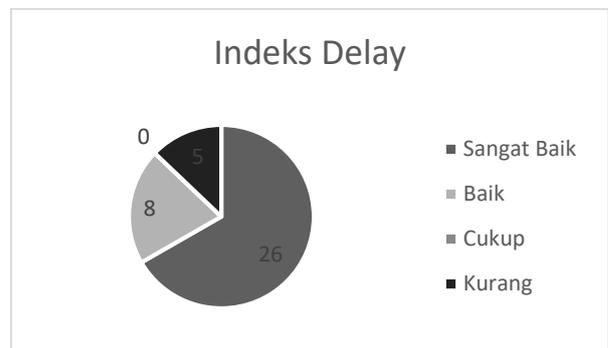
Pada akhir periode pengukuran, *delay* cenderung stabil kembali, meskipun mengalami beberapa lonjakan kembali yang tinggi dan dapat cenderung stabil kembali. Hal ini dapat diakibatkan oleh penyesuaian sistem terhadap kondisi jaringan yang perlu dioptimalkan agar *delay* dapat berkurang secara bertahap.

Tabel 2. Indeks Delay

Kategori Latensi	Besar Delay	Indeks
Sangat Baik	< 150 ms	26
Baik	150 s/d 300 ms	8
Cukup	300 s/d 450 ms	-
Kurang	> 450 ms	5

Dari hasil analisis berdasarkan *indeks delay*, dapat disimpulkan bahwa mayoritas waktu,jaringan dalam kondisi yang sangat bagus hingga bagus sebesar 85% . Namun terdapat kondisi yang kurang sebesar 12,5% yang mengindikasikan adanya sistem jaringan yang perlu diperbaiki.

Grafik 2. Indeks Delay



Berdasarkan data yang diambil menggunakan *software wireshark* peneliti mengambil 40 data untuk diuji dan dihitung indeks *delay*. Setelah peneliti menghitung *delay* tersebut maka menghasilkan indeks *delay* sangat bagus, semakin kecil *delay* semakin cepat data yang dikirim.

Peneliti menemukan beberapa indikasi dari hasil perhitungan *delay* yang menunjukkan bahwa data

tersebut memiliki kategori sangat bagus sebanyak 26 data, kategori bagus sebanyak 8 data, dan kategori kurang sebanyak 5 data. Hasil dari perhitungan yang dilakukan peneliti tersebut perlu mendapatkan perhatian khusus untuk melakukan perbaikan, antara lain dengan melakukan peningkatan kapasitas jaringan untuk mengurangi kemacetan saat pengiriman data, melakukan analisis konfigurasi jaringan untuk mengurangi *delay*, dan melakukan pemantauan dan pemeliharaan rutin untuk mendeteksi dan menangani masalah teknis sejak dini.

IV. Simpulan

Berdasarkan hasil penelitian dan pembahasan pengujian *delay* pada *prototype running text* di dapatkan hasil *delay* 40 data sebesar 292,856 ms dengan rata – rata *delay* 7,321 ms. Setelah di klasifikasikan sesuai dengan kategori latensi pengujian tersebut mendapatkan hasil 26 data sangat baik, 8 data bagus dan 5 data kurang. Dari hasil kesimpulan data diatas *delay* tersebut masih dikategorikan sangat baik. Penyebab terjadinya *delay* latensi kualitas jaringan dan pemrosesan data yang kurang baik. Semakin kecil nilai *delay* semakin bagus latensi jaringan.

Berdasarkan hasil pengujian diatas peneliti memberikan saran untuk pengujian selanjutnya dapat melakukan perbaikan pada pemrosesan data. Salah satunya menggunakan optimasi algoritma pemrosesan data dan peningkatan infrastruktur jaringan. Implementasi jaringan diharapkan dapat lebih meningkatkan performa *prototype* sampai menjadi alat.

Daftar Acuan

- [1] A. W. Finaka, "Apa Bedanya MRT, LRT, dan KRL," *IndonesiaBaik*, 2020. <https://indonesiabaik.id/infografis/apa-bedanya-mrt-lrt-dan-krl>
- [2] O. Wahyu Widayanti, "Perbedaan LRT, MRT, dan KRL yang ada di Indonesia," *TribunNews*. <https://www.tribunnews.com/travel/2021/10/25/perbedaan-lrt-mrt-dan-krl-yang-ada-di-indonesia-berikut-penjelasan>
- [3] T. D. M. L. M. W., "Rancang Bangun Integrasi Aplikasi Public Address System Text To Speech Dan Flight Information Display System Berbasis Wireless Guna Meningkatkan Kualitas Pelayanan Jasa Informasi Di Bandar Udara Halim Perdana Kusum," *J. Ilm. Aviassi*, vol. Vol 12 N, pp. 39–44, 2019, [Online]. Available: <https://journal.ppicurug.ac.id/index.php/jurnal-ilmiahaviassi/article/view/141/134>
- [4] D. grik, "Public Address System: Def, Evolution, Component, Setup, FAQ," *SearchTurntables*, 2022. <https://searchturntables.com/turntablism/public-address-system/>
- [5] E. Naf'an, "Akurasi Sistem Penjadwalan Sholat Digital Menggunakan Arduino Sebagai Pengendali," *J. Sismet Inf. dan Teknol.*, vol. vol.1 No.4, pp. 81–88, 2019, [Online]. Available: <https://jsisfotek.org/index.php/JSisfotek/article/view/13/13>
- [6] A. W. A. Antu, S. Abdussamad, and I. Z. Nasibu, "Rancang Bangun Running Text pada Dot Matrix 16X160 Berbasis Arduino Uno Dengan Update Data System Menggunakan Perangkat Android Via Bluetooth," *Jambura J. Electr. Electron. Eng.*, vol. 2, no. 1, pp. 8–13, 2020, doi: 10.37905/jjee.v2i1.4321.

- [7] N. Tri Wahyuni and S. Beta, "Running Text Information System Design Internet-Based for Small Outlets," *J. Appl. Inf. Commun. Technol.*, vol. vol.7 No.2, 2022.
- [8] C. Wiharya, P. Surya harjianto, G. Prawestri C.H, L. Hakim, and D. Dhesah Kharisma, "Implementasi Penampil Informasi Elektronik running text berbasis Wifi pada Pamean Cafe Kampung Wisata Mbesuk," *J. Apl. dan Inov. Ipteks Solidar.*, vol. vol.6 No.2, pp. 232–240, 2023, [Online]. Available: https://www.researchgate.net/publication/374797057_IMPLMENTASI_PENAMPIL_INFORMASI_ELEKTRONIK_RUNNING_TEXT_BERBASIS_WIFI_PADA_PEMEAN_CAFE_KAMPUNG_WISATA_MBESUK
- [9] R. Agustio Efransah, "Pengertian, Kelebihan, dan Kekurangan Arduino MEGA," *Kelas,teknisi*, 2023. [https://www.kelasteknisi.com/2023/01/kelebihan-dan-kekurangan-arduino-mega.html#:~:text=Arduino MEGA adalah jenis board mikrokontroler yang dikembangkan,khususnya dalam hal jumlah pin dan kapasitas memori.](https://www.kelasteknisi.com/2023/01/kelebihan-dan-kekurangan-arduino-mega.html#:~:text=Arduino%20MEGA%20adalah%20jenis%20board%20mikrokontroler%20yang%20dikembangkan,khususnya%20dalam%20hal%20jumlah%20pin%20dan%20kapasitas%20memori.)
- [10] I. Rifky, "mikrokontroler ESP32," *universitas Raharja*, 2021. <https://raharja.ac.id/2021/11/16/mikrokontroler-esp32-2/>
- [11] S. Gautam, "What is Wireshark? Applications, Features & How It Works," *KnowledgeHut*. <https://www.knowledgehut.com/blog/security/what-is-wireshark>