

## REALTIME TRAIN DATA RECORDER PADA KERETA LIGHT RAIL TRANSIT BERBASIS INTERNET OF THINGS

Ahmad Eka Fauzi<sup>1</sup>, Lilik Subiyanto<sup>2</sup>, Zindhu Maulana Ahmad Putra<sup>3</sup>

<sup>123</sup>Program Studi Teknik Otomasi, Jurusan Teknik Kelistrikan Kapal

Politeknik Perkapalan Negeri Surabaya

Jalan Teknik Kimia, Kampus ITS Sukolilo, Surabaya 60111

E-mail: ahmadekafauzi@gmail.com

### Abstrak

Kereta *Light Rail Transit* (LRT) merupakan angkutan kereta listrik yang menghubungkan Jakarta dengan kota-kota penyangganya seperti Bogor dan Bekasi. Kereta LRT sama seperti kereta pada umumnya yang kemungkinan mengalami terjadinya kecelakaan kereta maupun masalah pada sistem kereta. Hal ini dapat membahayakan keamanan dan keselamatan penumpang ketika kereta mengalami masalah sistem. Oleh karena itu jika terjadi kecelakaan kereta, petugas kontrol kereta perlu melakukan pengecekan terlebih dahulu untuk data sistem kereta yang valid. Melihat kondisi tersebut pada penelitian ini dibuatlah prototipe yang mampu *monitoring* atau memantau kondisi sistem kereta melalui jarak jauh. Prototipe ini dibuat dengan memperhitungkan suhu *bearing*, kelistrikan, kecepatan, kemiringan, dan waktu pada kereta LRT. Prototipe ini menerapkan metode *haversine formula* untuk mengetahui jarak yang ditempuh kereta LRT mulai ketika kereta berangkat hingga berhenti pada lokasi pemberhentian serta menggunakan protokol MQTT untuk komunikasi antara *hardware* dan *software*. Komunikasi protokol MQTT memiliki rata-rata *delay* waktu 0,801354 sekon ketika komunikasi antara *hardware* dan *software*. Serta data *packet loss* dari komunikasi MQTT sebesar 4,86%. Jarak yang ditempuh dengan menggunakan metode *haversine formula* memiliki nilai error rata-rata 0,5649%. Hasil *packet* data dari komunikasi akan terecord di memori dan *database* serta akan diolah dan divisualisasi pada *Website Server*.

**Kata Kunci:** *Haversine Formula*, Kereta LRT, *Monitoring Sistem*, *MQTT*, *Website Server*

### 1. PENDAHULUAN

Moda *transportasi* sudah sangat penting bagi kehidupan manusia dalam beraktivitas. Moda *transportasi* sendiri terbagi menjadi *transportasi* darat, air, dan udara. *Transportasi* di darat memiliki beberapa jenis *transportasi* yang terdiri dari *Motor*, mobil, truk, bis dan kereta api. Kereta api merupakan salah satu jenis moda *transportasi* umum yang efektif dan efisien dimana kereta api memiliki kelebihan dibandingkan dengan moda *transportasi* yang lain yaitu memiliki jalur jalan tersendiri (Putra 2009). Disamping kelebihan yang dimiliki kereta api yang cukup banyak, angka kecelakaan untuk moda *transportasi* kereta api juga cukup tinggi. Menurut pihak PT. Kereta Api Indonesia (PT.KAI) melalui (liputan6.com,2020) bahwa sepanjang tahun 2020 terjadi 198 kasus kecelakaan yang terjadi sejak awal tahun hingga awal bulan Oktober 2020.

Melihat dari permasalahan yang sering terjadinya kecelakaan kereta, maka dilakukan sebuah penelitian untuk membuat sebuah sistem pemantauan mengenai data sistem di

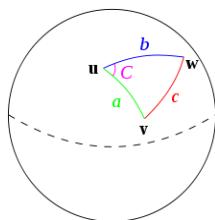
kereta api. Pemantauan dilakukan dengan cara memanfaatkan informasi dari hasil pembacaan sensor suhu *bearing*, kelistrikan, kecepatan, lokasi dan kemiringan dari kereta. Pembacaan sensor suhu *bearing*, kelistrikan, kecepatan, lokasi dan kemiringan akan diolah mikrokontroller untuk mempermudah dalam pembacaan secara *realtime*. Metode yang digunakan untuk mengetahui data hasil pembacaan sensor secara *realtime* pada kereta LRT adalah menggunakan *Internet of Things* (IoT). Data dari pembacaan mikrokontroller akan dikirim dan disimpan di *database* dan memori, untuk kemudian ditampilkan pada *Website Server* yang dapat di *monitoring* oleh pihak kontrol kereta maupun pihak stasiun yang akan dilewati kereta LRT. Dengan memanfaatkan media *internet* dengan komunikasi protokol MQTT untuk menghubungkan antara mikrokontroller dengan *Website Server* yang akan digunakan oleh masinis maupun pihak kontrol kereta dalam pemantauan dan informasi sebagai bahan analisa dan evaluasi dari pihak stasiun jika terjadi masalah sistem kereta.

Adapun tujuan dari tugas akhir ini antara lain dapat merancang sebuah alat yang mampu merecord data sistem di kereta *Light Rail Transit* (LRT) yang dapat dimonitoring jarak jauh, mengimplementasikan algoritma *Haversine formula* untuk mengetahui jarak yang telah ditempuh kereta *Light Rail Transit* (LRT) selama beroperasi, mengetahui infomasi secara *realtime* kondisi suhu *bearing*, kelistrikan, kecepatan, lokasi dan kemiringan dari kereta *Light Rail Transit* (LRT) serta mengimplementasikan *Internet of Things* dengan komunikasi protokol MQTT pada kereta *Light Rail Transit* (LRT).

## 2. METODOLOGI

### 2.1 Haversine Formula

Algoritma *Haversine formula* pada tugas akhir ini digunakan untuk mengukur jarak yang telah ditempuh kereta selama beroperasi. *Haversine formula* menggunakan garis bujur dan garis lintang sebagai *variable* masukannya dengan mengasumsikan bentuk bumi bulat sempurna seperti bola (Abadi Nugroho 2020).



Gambar 2.1 Ilustrasi Haversine Formula

Rumus dari *Haversine formula* sendiri sebagai berikut (Yulianto, Ramadiani, and Kridalaksana 2018).:

$$R = 6,371 \text{ km}$$

$$1 \text{ derajat} = 0.017453293 \text{ radian}$$

$$\Delta \text{latitude} = \text{latitude}_2 - \text{latitude}_1$$

$$\Delta \text{longitude} = \text{longitude}_2 - \text{longitude}_1$$

$$a = \sin^2\left(\frac{\Delta \text{latitude}}{2}\right) + \cos(\text{latitude}_1) \cdot \cos(\text{latitude}_2) \cdot \sin^2\left(\frac{\Delta \text{longitude}}{2}\right)$$

$$\cos(\text{latitude}_2) \cdot \sin\left(\frac{\Delta \text{longitude}}{2}\right)$$

$$c = 2 \cdot \arctan^2(\sqrt{a}, \sqrt{1-a})$$

$$d = R \cdot c$$

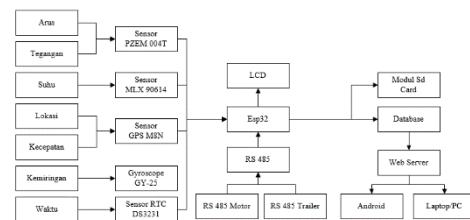
### 2.2 Message Queuing Telemetry Transport (MQTT)

*Message Queuing Telemetry Transport* atau biasa disingkat MQTT merupakan sistem protokol *publish* atau *subscribe* yang sangat

sederhana, dirancang untuk perangkat dan jaringan terbatas yang memiliki *bandwidth* rendah dan jaringan tidak dapat diandalkan. Komunikasi MQTT bersifat *machine to machine* atau M2M dan bekerja pada aplikasi yang bersifat *lightweight message*. Semua data yang dikirim akan terjamin oleh protokol MQTT meskipun koneksi jaringan dalam keadaan terputus (Abilovani, Yahya, and Bakhtiar 2018). Protokol komunikasi MQTT berlangsung dengan menggunakan TCP/IP dari *internet* dalam pemrosesan pengiriman data.

### 2.3 Diagram Blok Sistem

Diagram blok sistem merupakan penyajian bergambar dari fungsi yang dilakukan tiap komponen. Pada diagram blok semua variabel saling berhubungan dengan saling dihubungkan menggunakan blok fungsional dan mengandung informasi karakteristik tiap komponen dari sistem. Diagram blok sistem pada penelitian ini digambarkan seperti berikut.



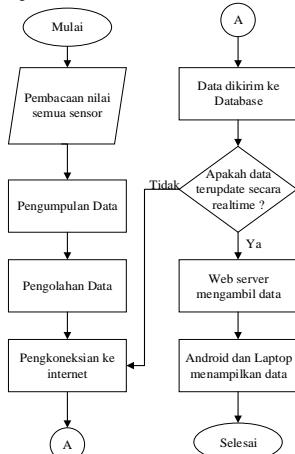
Gambar 2.2 Diagram Blok Sistem Motor Car

Gambar 2.2 merupakan diagram blok sistem pada *Motor Car*. Sistem dirancang menggunakan sensor arus, tegangan, suhu, lokasi, kecepatan, kemiringan dan waktu sebagai *input*. Kemudian nilai *input* di baca oleh sensor untuk kemudian dikirim ke Esp32. Esp32 memproses dan mengolah data hasil pembacaan sensor selain itu Esp32 sebagai titik central komunikasi mikrokontroler dengan RS 485. Hasil pengolahan Esp32 disimpan pada modul *sd card* dan *database* serta data akan ditampilkan pada *LCD*. Dengan data yang sudah tersimpan pada *database*, *Website Server* dapat memonitor nilai parameter kelistrikan, suhu *bearing*, lokasi, kecepatan, kemiringan dan waktu pada saat kereta LRT beroperasi. *Website Server* sendiri dapat diakses melalui *Operating System* Android maupun Laptop atau komputer.

### 2.4 Flowchart Alur Kerja Sistem

Pada tahapan ini terdapat langkah-langkah untuk mengidentifikasi jika terjadi kendala.

Perancangan sistem awal pada penelitian ini dapat ditunjukkan di bawah ini.

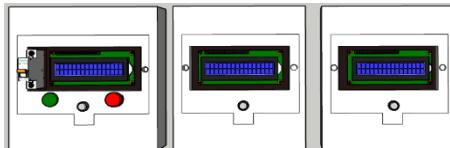


Gambar 2.4 Flowchart Kerja Sistem

### 3. HASIL DAN PEMBAHASAAN

#### 3.1 Hasil Perancangan Hardware

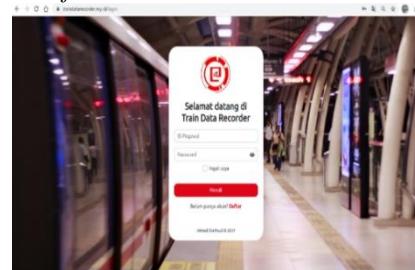
Perancangan *hardware* dari sistem *train data recorder* yang terdiri dari proses *wiring* komponen dan *programming* komponen. Dalam proses pemrograman komponen digunakan aplikasi *code editor* Arduino IDE untuk mengontrol sensor menggunakan mikrokontroller ESP32 dan Arduino Uno.



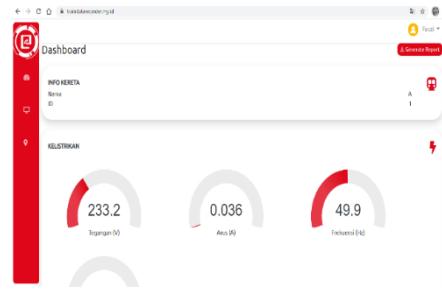
Gambar 3.1 Rancangan Hardware

#### 3.2 Hasil Perancangan Software

Perancangan *software* dari sistem *train data recorder* merupakan tampilan *interface Website Server*. Dalam proses pemrograman *website* digunakan aplikasi *code editor* Visual Studio Code untuk mendesain tampilan dari sebuah *website*. Berikut merupakan hasil rancangan desain *software Website Server* dan sistem *software* secara keseluruhan.

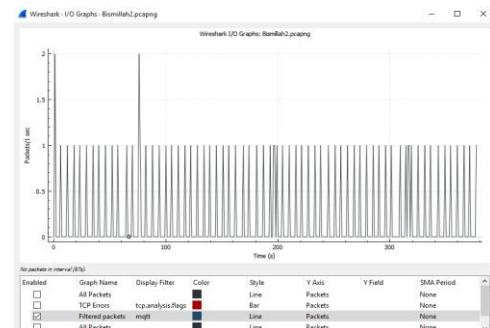


Gambar 3.2 Halaman Login



Gambar 3.2 Halaman Dashboard

#### 3.3 Hasil Perancangan Komunikasi Protokol MQTT



Gambar 3.3 Grafik I/O Komunikasi MQTT

Gambar 3.3 merupakan grafik I/O komunikasi MQTT berdasarkan *packet* data terhadap waktu. Berdasarkan dari grafik pada Gambar 3.3 bahwa *delay* atau selang waktu adanya 1 *packet* setiap ±5 sekon sesuai dengan *delay* yang telah di setting pada mikrokontroller ketika mengirim data ke *broker* MQTT. Ketika berada pada detik ke-1 dan detik ke-76 ada 2 *packet* data yang terkirim hal ini disebabkan request untuk dapat *connect* ke jaringan. Sehingga ada 2 *packet* yaitu *packet* data dan *packet* untuk request pada jaringan *server broker* MQTT.

#### 3.4 Hasil Metode Haversine Formula

Berikut merupakan tabel hasil pengujian metode *haversine formula* berdasarkan data sensor gps m8n. Perhitungan tersebut menggunakan rumus *haversine formula* yang telah ditentukan. Hasil tersebut kemudian dibandingkan dengan jarak pada *Google Maps* berdasarkan data lokasi latitude dan longitude.

Tabel 1. Hasil Pengujian Metode Haversine formula Berdasarkan Data Sensor GPS M8N

Ket. Lokasi	Latitude	Longitude	a	c	Jarak (km)
Lokasi Awal-Lokasi 1	-7.410578	112.259841	3.63643E-10	3.814E-05	0.24998
Lokasi Awal-Lokasi 2	-7.410578	112.259841	2.81159E-08	0.0003354	2.13655
Lokasi Awal-Lokasi 3	-7.410578	112.259841	9.23532E-08	0.0006084	3.87606
Lokasi Awal-Lokasi 4	-7.410578	112.259841	1.24259E-07	0.000705	4.49161
Lokasi Awal-Lokasi 5	-7.410578	112.259841	1.55961E-07	0.0007898	5.03206

Tabel 2. Perbandingan Jarak Hasil Perhitungan dengan Google Maps

Ket. Lokasi	Jarak Perhitungan (km)	Jarak Google Maps (km)	Error (%)
Lokasi Awal-Lokasi 1	0.2430	0.24	1.2428
Lokasi Awal-Lokasi 2	2.1366	2.1	1.7405
Lokasi Awal-Lokasi 3	3.8761	3.9	0.6137
Lokasi Awal-Lokasi 4	4.4916	4.5	0.1865
Lokasi Awal-Lokasi 5	5.0321	5	0.6413
Error rata-rata			0.5649

Tabel 2 merupakan hasil perbandingan perhitungan jarak dengan menggunakan rumus metode *haversine formula* dibandingkan dengan jarak pada *Google Maps*. Berdasarkan 5x pengujian yang dilakukan dengan melakukan perjalanan mulai lokasi awal hingga 5 titik pemberhentian berbeda. Berdasarkan Tabel 2 dapat disimpulkan bahwa jarak perhitungan dengan metode *haversine formula* dibandingkan *Google Maps* memiliki perbedaan. Dengan 5x pengujian yang telah dilakukan diketahui nilai error rata-rata dari pengujian tersebut sebesar 0.5649%

### 3.5 Analisa Keseluruhan Sistem

Sistem ini sudah mampu melakukan *monitoring* dan mendsistribusikan data. *Website* sebagai *output* untuk visualisasi data yang telah terecord di *database*. User *website* dapat memonitoring data secara *realtime* ketika kereta beroperasi sedangkan user *server laptop* dapat mengolah data yang dikirim oleh mikrokontroller. Melalui *website* juga dapat mengetahui jarak yang telah ditempuh kereta LRT selama beroperasi dengan metode *haversine formula*. Tingkat kesulitan pada tugas akhir ini pada proses integrasi antara *hardware* dan *software* serta perhitungan metode *Haversine formula* agar menjadi suatu sistem yang dapat berjalan sesuai alur dari tugas akhir ini dan berkesinambungan dengan tepat.

### 4. KESIMPULAN

Berdasarkan hasil yang telah didapatkan pada analisa dan pengujian sistem yang telah dibuat, maka dapat disimpulkan bahwa:

1. Sistem *train data recorder* pada tugas akhir ini mampu merecord data yang berasal dari rangkaian mikrokontroller yang terpasang pada objek kendaraan. *Output record* data akan tersimpan di memori dan *database*. Memori terpasang di modul SD Card yang terhubung dengan mikrokontroller. Setiap

gerbang *Motor car*, *Motor* dan *Trailer* masing-masing terdapat satu memori yang merecord data dari gerbang tersebut. Data yang tersimpan di *database* merupakan data yang berasal dari setiap gerbang *Motor car*, *Motor* dan *Trailer* hasil komunikasi serial.

2. *Train data recorder* yang dirancang penulis mampu menghitung jarak yang telah ditempuh kereta LRT atau pada penelitian ini berupa objek kendaraan. Dengan menerapkan metode *Haversine formula* untuk menghitung jarak mulai lokasi awal berangkat hingga lokasi dimana kereta berada saat ini. Dengan berdasarkan data *latitude* dan *longitude* dari koordinat kereta sehingga algoritma dari metode *Haversine formula* dapat menghitung jarak yang telah ditempuh kereta selama beroperasi.
3. Sistem *train data recorder* pada tugas akhir ini mampu mengetahui kondisi dari suhu *bearing*, kelistrikan, kecepatan, lokasi dan kemiringan dari kereta atau pada penelitian ini berupa objek kendaraan secara *realtime*. Hal ini karena parameter dari kondisi tersebut diukur oleh sensor yang masing-masing terpasang pada titik tertentu. Hasil data pengukuran dari sensor akan didistribusikan ke *website* oleh esp32 melalui media internet. Mikrokontroller melakukan pengiriman data ke *website* dengan selang waktu ±5 sekon setiap pengiriman paket data.
4. *Train data recorder* yang dirancang pada penelitian ini mampu berkomunikasi menggunakan protokol komunikasi MQTT untuk pertukaran paket pesan data dengan menggunakan modul *Wifi esp32* sehingga sistem sudah meliputi proses *Internet of Things* dengan membaca paket pesan data yang diterima dari mikrokontroller. Sistem dapat melakukan *publish* dan *subscribe* topik yang diperlukan, termasuk juga untuk merecord data yang telah diterima dari mikrokontroller.

### 5. DAFTAR PUSTAKA

- Abadi Nugroho, Abadi Nugroho. 2020. “Penerapan Metode *Haversine formula* Untuk Penentuan Titik Kumpul Pada Aplikasi Tanggap Bencana.” *Metik Jurnal* 4(2): 69–75.  
 Abilovani, Zavero Brillianata, Widhi Yahya, and Fariz Andri Bakhtiar. 2018. “Implementasi Protokol MQTT Untuk Sistem Monitoring Perangkat IoT.” *Jurnal Pengembangan Teknologi Informasi dan*

- Ilmu Komputer (J-PTIIK) Universitas Brawijaya* 2(12): 7521–27.
- Aldian, Dwi Pamungkas. 2018. “ELEKTRONIK PADA RUANG PENUMPANG KAPAL FERRY BERBASIS LONG RANGE WIRELESS COMMUNICATIONS ( LORA ) ABSTRAK.” : 220.
- Herianaa, Octa, and Heru Rahmat Mulyadib. 2014. “Sistem Data Logger Lima Channel Input Untuk Sensor Navigasi Kapal Maritim.” *Jurnal Elektronika dan Telekomunikasi* 14(2): 61–66.
- Mahendra Sanjaya, I Putu Gede, Cok Gede Indra Partha, and Duman Care Khrisne. 2017. “Rancang Bangun Sistem Data Logger Berbasis Visual Pada Solar Cell.” *Majalah Ilmiah Teknologi Elektro* 16(3): 114.
- Mulatsih, Retno, and Sulistyowati. 2016. “STUDI KEPUASAN PENUMPANG KERETA API KAMANDAKA STIMART ‘ AMNI ’ Semarang Kehadiran Kereta Api Di Indonesia Bawah Lingkungan PT Kereta Api Indonesia.” *Jurnal Dinamika Ekonomi dan Bisnis* 13(2): 119–30.
- Putra, Estrada Witrias. 2009. “STUDI KESELAMATAN DAN KEAMANAN TRANSPORTASI DI PERLINTASAN SEBIDANG ANTARA JALAN REL DENGAN JALAN UMUM (Studi Kasus Perlintasan Kereta Api Dijalan Kaligawe Kota Semarang).” : 1–123. <https://lib.unnes.ac.id/113/>.
- Romdloni, Moh Nazilus Sa’din, Riza Alfta, and Rosida Vivin Nahari. 2017. “Prototype Sistem Monitoring Dan Pengendalian Pintu Air Otomatis Sebagai Peringatan Dini Bahaya Banjir Berbasis Internet of Things.” *Seminar Nasional Matematika dan Aplikasinya*: 377–85.
- Ulin Nuha. 2018. “Rancang Bangun Alat Ukur Iluminansi Penerangan Berbasis Data Logger Terintegrasi Dengan Iot ( Internet of Things ).”
- Yulianto, Yulianto, Ramadiani Ramadiani, and Awang Harsa Kridalaksana. 2018. “Penerapan Formula Haversine Pada Sistem Informasi Geografis Pencarian Jarak Terdekat Lokasi Lapangan Futsal.” *Informatika Mulawarman : Jurnal Ilmiah Ilmu Komputer* 13(1): 14.