



[Home](#) / [Archives](#) /

[Vol 32 No 1 \(2022\): Sainstech : Jurnal Penelitian dan Pengkajian Sains dan Teknologi](#) / [Artikel](#)

## IMPLEMENTASI SISTEM MONITORING BUS TRANS SEMARANG BERBASIS LORA (LONG RANGE)

**Tasdik Darmana**

**Fadila Annas**

**Ariman .**

**DOI:** <https://doi.org/10.37277/stch.v32i1.1239>

### Abstract

**ABSTRACT** Trans Semarang buses as transportation used by residents the city of Semarang often experience delays at one corridor stop so that passenger waiting times are quite long, so monitoring is needed on each bus. LoRa with long-distance communication capabilities without an internet signal with low power over a wide area, so it can be used to monitor the position and speed of the Trans Semarang bus for efficient passenger waiting time. The tool used to monitor Trans Semarang bus consists of a transmitter section as a data sender and a receiver as a data receiver with 915 MHz LoRa communication media and there is a buzzer as a bus speed warning and an emergency button when the bus has problems on the road. The test results from this study are the device can send and receive data with the furthest distance of 2200 meters with a maximum speed that can be measured is 46 km/hour, then it can give speed limit warnings and bus emergency warnings. **Keywords :** LoRa 915 MHz, Trans Semarang Bus, Distance, Speed



2022-03-29

Issue

[Vol 32 No 1 \(2022\): Sainstech : Jurnal Penelitian dan Pengkajian Sains dan Teknologi](#)

Section

Artikel

### Most read articles by the same author(s)

- Titi Ratnasari, Tasdik Darmana, Jumiati Jumiati, Arif Sutyanegara, M Kahfi Fachelinno, Tri Purnama Putra, Ianatut Toyyibah, [Rancangan Alat Pengisi Baterai Gadget Dengan Menggunakan Pembangkit Listrik Tenaga Surya](#) , [SAINSTECH: JURNAL PENELITIAN DAN PENGKAJIAN SAINS DAN TEKNOLOGI: Vol 26 No 2 \(2016\): SAINSTECH](#)
- Ariman ., Mochamad Iklil, [PERANCANGAN JARINGAN KOMUNIKASI SISTEM INFORMASI DAN ADMINISTRASI BRIMOB \(SIAB\) BERBASIS SERAT OPTIK](#) , [SAINSTECH: JURNAL PENELITIAN DAN PENGKAJIAN SAINS DAN TEKNOLOGI: Vol 31 No 2 \(2021\): Sainstech : Jurnal Penelitian dan Pengkajian Sains dan Teknologi](#)

[ISSN CHECK](#)[Make a Submission](#)[FOCUS AND SCOPE](#)[EDITORIAL TEAM](#)[REVIEWERS](#)[AUTHOR GUIDLINES](#)[PUBLICATION ETHICS](#)[COPYRIGHT FORM](#)

## Language

[Bahasa Indonesia](#)

[English](#)

## Information

[For Readers](#)

[For Authors](#)

[For Librarians](#)

3 2 3 0 4



[Google Cendekia](#)



[View MyStat](#)

Platform &  
workflow by  
**OJS / PKP**

# IMPLEMENTASI SISTEM MONITORING BUS TRANS SEMARANG BERBASIS LORA (*LONG RANGE*)

Tasdik Darmana<sup>1</sup>, Fadila Annas<sup>2</sup>, Ariman<sup>3</sup>

<sup>1,2,3</sup>Fakultas Ketenagalistrikan dan Energi Terbarukan, Institut Teknologi PLN  
Menara PLN, Jl. Lingkar Luar Barat, Duri Kosambi, Jakarta 11750  
e-mail: [tasdik.darmana@itpln.ac.id](mailto:tasdik.darmana@itpln.ac.id) ; [annasfadila@gmail.com](mailto:annasfadila@gmail.com) ; [ariman@istn.ac.id](mailto:ariman@istn.ac.id)

---

## ABSTRACT

*Trans Semarang buses as transportation used by residents the city of Semarang often experience delays at one corridor stop so that passenger waiting times are quite long, so monitoring is needed on each bus. LoRa with long-distance communication capabilities without an internet signal with low power over a wide area, so it can be used to monitor the position and speed of the Trans Semarang bus for efficient passenger waiting time. The tool used to monitor Trans Semarang bus consists of a transmitter section as a data sender and a receiver as a data receiver with 915 MHz LoRa communication media and there is a buzzer as a busspeed warning and an emergency button when the bus has problems on the road. The test results from this study are the device can send and receive data with the furthest distance of 2200 meters with a maximum speed that can be measured is 46 km/hour, then it can give speed limit warnings and bus emergency warnings.*

**Keywords :** LoRa 915 MHz, Trans Semarang Bus, Distance, Speed

## ABSTRAK

Bus Trans Semarang sebagai transportasi yang digunakan oleh penduduk kota Semarang sering mengalami keterlambatan pada halte satu koridor sehingga waktu tunggu penumpang cukup lama, maka diperlukannya pemantauan pada masing – masing bis. LoRa dengan kemampuan komunikasi jarak jauh tanpa sinyal internet dengan daya rendah pada cakupan area yang luas, sehingga dapat dimanfaatkan untuk memantau posisi serta kecepatan bus Trans Semarang untuk efisiensi waktu tunggu penumpang. Alat yang digunakan untuk memonitoring bus Trans Semarang terdiri dari bagian *transmitter* sebagai pengirim data dan *receiver* sebagai penerima data dengan media komunikasi LoRa 915 MHz serta terdapat *buzzer* sebagai peringatan kecepatan bus dan tombol darurat ketika bus mengalami kendala di jalan. Hasil uji dari penelitian ini adalah alat dapat mengirim serta menerima data dengan jarak terjauh yaitu 2200 meter dengan kecepatan maksimal yang dapat diukur adalah 46 km/jam, kemudian dapat memberi peringatan batas kecepatan dan peringatan keadaan darurat bus.

**Kata Kunci :** LoRa 915 MHz, Bus Trans Semarang, Jarak, Kecepatan

### 1. Latar Belakang

Bus Trans Semarang seringkali mengalami keterlambatan kedatangan pada halte dalam satu koridor sehingga waktu tunggu penumpang menjadi cukup lama, maka dari hal ini diperlukannya pemantauan pada masing – masing bis. Pemantauan pada bus diperlukan untuk memprediksi waktu tunggu penumpang pada halte dalam satu koridor, memantau perilaku pengemudi bus dalam mengemudikan bus sesuai aturan

kecepatan yang ditetapkan atau tidak, mempermudah respon pada bus ke operator ketika bus mengalami kendala di jalan. Pemantauan pada bus memerlukan alat yang memiliki teknologi tanpa kabel (*wireless*) dengan jangkauan komunikasi dalam area yang cukup luas. Salah satu teknologi *wireless* zaman sekarang ini adalah LoRa (*Long Range*) yang menggunakan daya rendah dan tidak perlu menggunakan internet dalam penggunaannya serta merupakan teknologi

komunikasi jarak jauh yang cakupan jaringannya cukup luas. (Dan 2020)

Sistem monitoring bus Trans Semarang dirancang berupa alat untuk memantau kecepatan serta memonitoring posisi bus ketika bus berjalan pada jalurnya dengan menggunakan LoRa 915 MHz dan diperasikan oleh operator pemantau. Alat monitoring bus Trans Semarang merupakan pengembangan dari GPS *Tracker* yang memantau posisi kendaraan. Terdapat penambahan fitur keamanan ketika bus dalam keadaan darurat serta peringatan kecepatan ketika pengemudi melebihi batas kecepatan yang diatur. Sehingga dapat memprediksi waktu tunggu penumpang pada halte dalam satu koridor, memantau perilaku pengemudi dalam mengemudikan bus, mempermudah respon kepada operator ketika bus mengalami kendala di jalan. (Arijato 2008)

Bus Trans Semarang harus memiliki waktu keberangkatan dan kedatangan yang tepat. Waktu kedatangan dan keberangkatan yang terlambat dapat menyebabkan penumpang tidak nyaman, bahkan sampai terjadi penumpukan penumpang dalam satu jalur. Pada salah satu koridor bus Trans Semarang yaitu Koridor 6 (Universitas Diponegoro – Universitas Negeri Semarang).

Berdasarkan pada rumusan masalah, maka pada penelitian diharapkan dapat mengaplikasikan LoRa untuk memantau posisi, kecepatan, keamanan bus dari jarak jauh melalui *dashboard* dapat menganalisis jarak maksimal penggunaan alat untuk memantau bus, menganalisis kinerja alat untuk pemantauan batas kecepatan bus dan dapat mengevaluasi cara respon alat dalam hal keamanan bus ketika terjadi keadaan darurat pada bus saat di jalan

## **2. Penelitian yang Relevan**

### **2.1 Bus Trans Semarang**

Trans Semarang sering disebut BRT adalah sistem transportasi bus raya terpadu di Jawa Tengah yang beroperasi di Kota dan (sebagian) Kabupaten Semarang. Layanan ini dioperasikan guna mengurai kemacetan di Kota Semarang serta untuk mengakomodir para pengguna bus menuju pusat kota dan destinasi wisata yang ada di

Kota Semarang. Dalam pelaksanaannya, Trans Semarang dikelola oleh Badan Layanan Umum (BLU) Unit Pelaksana Teknis Dinas (UPTD) dibawah Dinas Perhubungan, Komunikasi, dan Informatika Kota kecuali kota Semarang, non- penor petugas tiket armada, petugas persiapan armada, petugas operation, petugas timer, hingga pengawas angkutan. Bus berjalan di jalur khusus, dan harga tiket disubsidi oleh pemerintah daerah. Trans Semarang merupakan salah satu layanan transportasi umum yang disediakan oleh Pemerintah Kota Semarang dan banyak digunakan oleh warga kota Semarang dan sekitarnya dalam bepergian dikarenakan tarif yang relatif terjangkau, terjangkau. (dihitung dari keberangkatan pertama dan keberangkatan terakhir dari masing masing pool / terminus), kecuali Koridor Bandara yang jam beroperasi 05.30-17.30 WIB.

Saat ini Trans Semarang memiliki delapan koridor utama, satu koridor khusus, dan tiga koridor pengumpan. Terdiri dari koridor 1 jurusan Terminal Mangkang – Terminal Penggaron, koridor 2 jurusan Terboyo – Terminal Sisemut Ungaran, koridor 3A dan 3B jurusan Pelabuhan Tanjung Emas – Elizabeth, koridor 4 jurusan Terminal Cangkiran – Tajilau Marina – Tajirau Marina – Galaventes Station – Stargaraventes, koridor 6 jurusan Undip Tembalang – Unnes Sekaran, koridor 7 jurusan Terboyo – Pemuda Balaikota Semarang via Genuk, dan koridor 8 jurusan Terminal Cangkiran – Simpang Lima via Gunungpati. Selain itu, terdapat Koridor Bandara yang melayani rute Bandara-Simpang Lima dan koridor pengumpan.

Trans Semarang menerapkan tiga jenis halte (terkadang masih disebut *refuge*), yaitu halte permanen, portabel, serta rambu. Untuk halte permanen, dishediakan fasilitas berupa tempat duduk, jendela, tempat sampah, dan biasanya terdapatramp untuk membantu penyandang disabilitas. Sistem terpusat tertutup tanpa sistem pengumpan, Platform tinggi untuk naik dan turun cepat, Pengadaan bus Sektor Publik dan

pengoperasian bus Sektor Swasta, Beroperasi pada 400.000 penumpang/hari (2018) (www.transsemarang.co.id)

## 2.2 LoRa (LONG RANGE)

LoRa atau *LONG RANGE* merupakan teknologi untuk komunikasi *wireless* atau tanpa kabel yang dikembangkan dan dipatenkan oleh Semtech, yang digunakan untuk menciptakan suatu jaringan *wide-area low-power* (LPWAN) dan untuk aplikasi dari mesin ke mesin (M2M) serta memanfaatkan *Internet of Things* (IoT). Satu gateway LoRa mampu memuat hingga ribuan node.

Gateway biasanya mengacu pada box yang menampung perangkat keras dan perangkat lunak aplikasi yang melakukan tugas-tugas penting untuk menghubungkan perangkat IoT ke cloud. Perangkat IoT menggunakan gateway sebagai hub pusat untuk menghubungkan data tersebut ke jaringan eksternal. Gateway sangat mirip dengan router Wi-Fi. Ini memiliki konsentrator LoRa, yang memungkinkannya menerima sinyal RF yang dikirim oleh perangkat LoRaWAN, yang diubah menjadi sinyal yang kompatibel dengan server.

Gateway LoRaWAN dapat melayani beberapa grup perangkat sekaligus, meskipun gateway sering digunakan dalam grup yang tumpang tindih. Perangkat akan mengirimkan sinyalnya sebagai paket RF untuk diambil oleh gateway mana pun dalam jangkauan, dengan koneksi perangkat ke gateway yang paling kuat meneruskan pesan ke cloud. Memiliki beberapa gateway menambah ketahanan jaringan jika salah satu dari mereka gagal. Gateway LoRaWAN dapat diinstal dan dijalankan untuk jaringan pribadi yang dikendalikan oleh perusahaan. Untuk melakukan ini, data dikirim secara nirkabel terlebih dahulu ke gateway LoRaWAN. Gateway kemudian akan meneruskan data dari beberapa sensor array ke cloud dan akhirnya ke aplikasi user. Setelah diambil oleh gateway, sinyal RF dikonversi ke format yang memungkinkan kecepatan transfer lebih cepat. Laju data maksimum LoRa, 50Kb/dtk, cukup untuk komunikasi perangkat ke perangkat atau perangkat ke

gateway, tetapi akan menjadi hambatan serius saat mencoba mengirimkan ribuan pesan ke cloud – oleh karena itu perlunya gateway yang dirancang dengan baik untuk mengaktifkan tingkat transfer data yang lebih tinggi dari gateway ke cloud. Biasanya koneksi antara gateway dan cloud dibuat dengan ethernet untuk kecepatan tetapi LTE dan Wi-Fi juga merupakan pengganti praktis di lokasi luar ruangan.

Jangkauan LoRa lebih besar daripada jaringan seluler karena LoRa dapat digunakan untuk jaringan publik, *private*, maupun *hybrid*. LoRa menggunakan frekuensi yang tergantung dari wilayahnya. LoRa menggunakan frekuensi 433 MHz untuk wilayah Asia, frekuensi 868 MHz untuk wilayah Eropa, dan frekuensi 915 MHz untuk wilayah Amerika Utara. LoRa mudah dipasang ke infrastruktur yang ada sehingga banyak operator jaringan seluler yang telah menggunakan LoRa untuk melengkapi jaringan seluler / *wireless* dengan LPWAN.

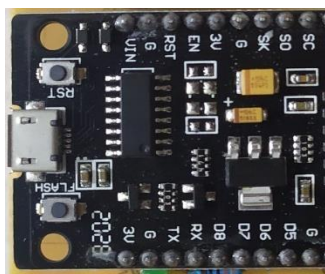
LoRa juga menyediakan berbagai macam fitur : Biaya yang cukup rendah sehingga dapat mengurangi biaya infrastruktur, biaya operasional, serta sensor-sensor yang mempunyai jaringannya masing-masing, Memungkinkan penggunaanya bisa mendeteksi lokasi keberadaan suatu benda tanpa biaya, Konsumsi dayanya rendah yaitu hanya berkisar dari 13mA sampai 15mA, Mampu menjangkau jarak hingga 100 Km. Memiliki sekuritas yang aman karena tertanam *end-to-end* enkripsi AES128, Dapat terhubung dengan ribuan perangkat node dalam suatu jaringan sehingga sangat sesuai untuk IoT, Mampu dengan cepat berinteraksi atau beradaptasi dengan sistem lain, Cukup ideal untuk operator jaringan publik yang melayani banyak pelanggan karena mampu mendukung jutaan pesan per *base station*.

Spesifikasi pada LoRa 915 MHz : Jangkauan maksimum 168 dB, Modulasi FSK, GFSK, MSK, GMSK, LoRaTM dan OOK, Sensitivitas tinggi : hingga -148 dBm, Ujung depan IIP3 = -12,5 dBm, +20 dBm - 100 mW output RF konstan, +14 dBm efisiensi tinggi PA, Bit rate yang dapat diprogram

hingga 300 kbps, Rentang Dinamis RSSI 127 dB, Sensor RF Otomatis dan CAD dengan AFC ultra cepat, Kekebalan pemblokiran yang sangat baik, Sensor suhu built-in dan indikator baterai rendah, Sinkronisasi bit built-in untuk pemulihan jam, Synthesizer terintegrasi penuh dengan resolusi 61 Hz, Arus RX rendah 10,3 mA, retensi register 200 nA, Deteksi pembukaan, Mesin paket hingga 256 byte dengan CRC.

Tabel 1 Perbandingan Media Transmisi Wireless

Media	Range	Frequency	Date Rate	Energy Consumption	Cost
Bluetooth	30 - 300 ft	2.4 GHz	1 Mbps	Medium	Low
Blee	Up to 10 ft	2.4 GHz	1 Mbps	Low	Low
Zigbee	0.3 - 1.6 km	2.4 GHz	250 Kbps	Low	Low
WiFi	100 - 150 ft	2.4 GHz	11 - 54 Mbps	High	High
LoRa	2 - 15 km	ISM band 868,915 MHz	0.3 - 50 Kbps	Low	Low



Gambar 1 LoRa (Long Range)

### 2.3 Adafruit LoRa

Adafruit LoRa merupakan modul radio dengan versi radio frekuensi 900 MHz sehingga dapat digunakan untuk transmisi frekuensi 868MHz atau penerimaan frekuensi 915MHz dilengkapi dengan LoRa untuk komunikasi serta tambahan 20dBm dengan modulasi radio khusus walaupun tidak kompatibel dengan modul radio RFM69, tetapi dapat digunakan lebih jauh dengan mudah menempuh jarak 2 Km menggunakan antena kabel sederhana, atau hingga 20 Km dengan antena terarah dan pengaturan tweaker sehingga lebih bagus untuk membuat jaringan nirkabel yang lebih fleksibel sebagai penerima data dan sebagai *host* untuk berkomunikasi dengan

*client*.

Radio Adafruit Feather M0 RFM95 LoRa (900MHz) adalah radio paket LoRa menggunakan Feather sebagai standar baru untuk inti mikrokontroler portabel. Pada area jantung Feather M0 terdapat beberapa komponen yaitu : prosesor ATSAM21G18ARM Cortex M0, clock pada 48 MHz dengan tegangan 3.3 V, konektor untuk salah satu baterai polimer Lithium 3.7 V dan pengisian baterai internal sehingga mempermudah dalam penggunaan portabel, memiliki 256K FLASH (8x lebih banyak dari Atmega328 atau 32u4) dan 32K RAM (16x lebih banyak) dan dilengkapi USB bawaan sehingga memiliki program USB-to-Serial serta kemampuan debug yang terpasang tanpa memerlukan FTDI.

Spesifikasi pada Radio Adafruit Feather M0 RFM95 LoRa (900MHz) adalah: Modul berbasis SX127x LoRa dengan antarmuka SPI, Paket radio kompatibel dengan Arduino, Menggunakan band ISM bebas lisensi (ITU "Eropa" @ 433MHz dan ITU "Americas" @ 900MHz), +5 hingga +20 dBm hingga 100 mW, kemampuan output daya dapat dipilih dalam perangkat lunak), ~ 300uA selama stanby, ~ 120mA puncak selama transmisi 20dBm, ~ 40mA selama mendengarkan radio aktif, Antena kawat sederhana atau tempat untuk konektor uFL



Gambar 2 Adafruit LoRa

### 2.4 LoRa Radio Module

LoRa Radio *Module* yaitu termasuk jenis modem data radio dengan data rate rendah untuk jarak jauh berbasis Sx1276 dari Semtech. LoRa Radio merupakan jenis modul *transceiver* sub-1 GHz berbiaya rendah yang dirancang untuk operasi di pita ISM (*Industrial Scientific Medical*) dan LPRD yang tidak berlisensi. Modulasi / demodulasi spektrum frekuensi, operasi multi-saluran, efisiensi *bandwidth* tinggi,



dan kinerja anti- pemblokiran membuat modul LoRa mudah untuk menerapkan koneksi nirkabel yang kuat dan andal.

LoRa Radio Modul adalah modem radio data transparan yang berkomunikasi dengan host pada format data preset tanpa perlu encoding / decoding. LoRa Radio Modul digunakan untuk media transmisi pengirim data dan sebagai *client* untuk berkomunikasi dengan *host*. Komunikasi antara modul pusat dan modul node bersifat dua arah tetapi modul node tidak dapat saling berbicara.

Fitur - fitur pada LoRa Radio diantaranya : Spektrum Frekuensi LoRa, Pita frekuensi ISM 433 / 490Mhz, -137 dBm menerima sensitivitas, Maksimal 20 dBm. daya keluaran, Pengaktifan port serial, Pengaktifan nirkabel, Kemampuan jaringan bintang, Tegangan suplai 3.4 ~ 5.5,

Spesifikasi dari LoRa Radio yaitu : Tegangan operasi : 2.1 ~ 3.6V (bertenaga baterai) atau 4.5 ~ 5.5V, Pita frekuensi kerja : 433MHz, Modulasi : spektrum penyebaran LoRa, Daya keluaran :  $\leq$  20dBm, Menerima sensitivitas : -148dBm, Arus emisi : 120mA, Menerima arus :  $\leq$  15.2mA, Arus sleep :  $\leq$  3.9uA, Waktu sleep : Opsional 2S, 4S, 6S, 8S, 10S, Jeda aktif : opsional 2ms, 4ms, 8ms, 16ms, 32ms, 64ms, Waktu aktif : segera bangun, Communication Interface : komunikasi serial TTL, Kecepatan baud serial : 1200 ~ 57600,9600bps (default), Verifikasi port serial : 8E1,801,8N1 (opsional), Suhu pengoperasian : -40 °C - + 80 °C



Gambar 3 LoRa Radio Modul

## 2.5 Global Positioning System (GPS)

*Global Positioning System* (GPS) merupakan sistem navigasi dengan satelit minimal 24 satelit yang dipakai ke semua keadaan cuaca, di lokasi manapun di bumi dengan pemakaian 24 jam sehari, tidak memerlukan biaya sewa atau biaya operasional. GPS digunakan sebagai

pengukur kecepatan dan penentuanposisi / lokasi.

Satelit GPS memutar bumi dua kali sehari sesuai dengan orbit. Setiap satelit mengirimkan sinyal unik dan parameter orbit ke perangkat GPS dapat menyesuaikan kode serta perkiraan lokasi yang tepat dari penerima GPS menghitung lokasi pengguna yang tepat. Penerima GPS mengukur jarak ke setiap satelit dalam jumlah waktu yang dibutuhkan untuk menerima sinyal yang dipancarkan. Dengan pengukuran jarak dari beberapa satelit lagi, penerima dapat mengatur pengguna dan menampilkannya secara elektronik. Setelah posisi penerima GPS ditentukan, perangkat GPS dapat menghitung informasi lain, seperti : Kecepatan Arah, Jalan, Jarak perjalanan, Jarak ke tujuan

Satelit GPS mengirimkan setidaknya 2 sinyal radio berdaya rendah.Sinyal GPS berisi 3 jenis informasi, yaitu: Kode pseudorandom adalah I.D. kode sebagai sinyal satelit yang akan mengirimkan informasi, Data Ephemeris diperlukan untuk menentukan posisi satelit serta memberikan informasi penting tentang kesehatan satelit, waktu dan tanggal saat ini, Data Almanac memberikan informasi kepada penerima GPS pada setiap satelit. GPS memiliki keharusan setiap saat sepanjang hari menunjukkan informasi orbit untuksatelit itu dan setiap satelit lain dalam sistem.



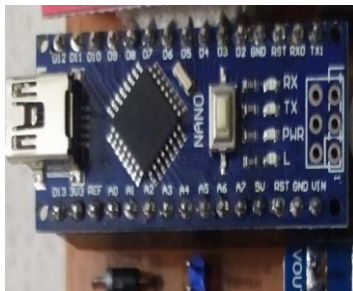
Gambar 4 Global Positioning System (GPS)

## 2.6 Arduino Nano

Arduino Nano adalah mikroprosesor kecil, lengkap, dan ramah berbasis ATmega328. Arduino Nano dilengkapi dengan 30 header I/O *male*, dalam konfigurasi seperti dip-30, yang dapat diprogram menggunakan lingkungan pengembangan terintegrasi Perangkat Lunak Arduino (IDE), yang umum untuk

semua papan Arduino dan berjalan baik online maupun offline. *Board* dapat diberi daya melalui kabel micro-USB tipe-B, atau melalui baterai 9V. Arduino Nano memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lainnya. ATmega328 menyediakan komunikasi serial UART TTL (5V), yang tersedia pada pin digital 0 (RX) dan 1 (TX). FTDI FT232RL di papan menyalurkan komunikasi serial ini melalui USB dan driver FTDI (disertakan dengan perangkat lunak Arduino) menyediakan *port com* virtual ke perangkat lunak di komputer. Perangkat lunak Arduino mencakup monitor serial yang memungkinkan data tekstual sederhana dikirim ke dan dari papan Arduino. LED RX dan TX di papan akan berkedip ketika data sedang dikirim melalui chip FTDI dan koneksi USB ke komputer (tetapi tidak untuk komunikasi serial pada pin 0 dan 1). Pustaka *Software Serial* memungkinkan komunikasi serial pada pin digital Nano mana pun. ATmega328 juga mendukung komunikasi I2C (TWI) dan SPI. Perangkat lunak Arduino menyertakan perpustakaan kawat untuk menyederhanakan penggunaan bus I2C.

Spesifikasi Arduino Nano :  
Mikrokontroler: Microchip ATmega328P,  
Tegangan Operasi: 5 Volt, Tegangan Input: 6 hingga 20 Volt, Pin I/O Digital: 14 (ditambah 6 pin keluaran PWM kaleng), Pin Input Analog: 8, Arus DC per Pin I/O: 40 mA, Arus DC untuk 3.3V Pin: 50 mA, Memori Flash: 32 KB di mana 0,5 KB digunakan oleh bootloader, SRAM: 2 KB, EEPROM: 1 KB, Kecepatan Jam: 16 MHz, Panjang: 45 mm, Lebar: 18 mm, Berat: 7 gram



Gambar 5 Arduino Nano

## 2.7 Baterai

Baterai adalah sumber tenaga listrik yang terdiri dari satu atau lebih sel elektrokimia dengan koneksi eksternal untuk menyalakan perangkat listrik. Baterai dengan tipe 18650 VCC berjenis Lithium dengan arus sebesar 3,7 A dirangkai secara seri dihubungkan menggunakan kotak baterai yang digunakan digunakan sebagai sumber daya pada untuk LoRa dapat bekerja di segala tempat ketika melakukan pengujian alat.



Gambar 6 Baterai

## 2.8 Regulator

Regulator tegangan adalah sistem yang dirancang untuk secara otomatis mempertahankan tegangan konstan.

Regulator digunakan sebagai pengatur tegangan keluaran dari adaptor 5VDC menuju LoRa. Regulator sebagai pengatur tegangan keluaran dari adaptor 5V menuju LoRa dan meningkatkan tegangan (*step up*) dari baterai ke komponen dengan melakukan proses penaikan tegangan (*voltage multiplier*) yang dipengaruhi kapasitor dan dioda.

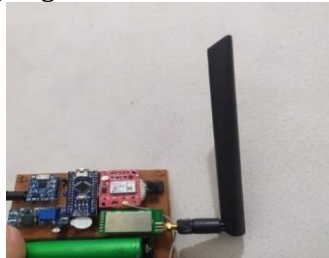


Gambar 7 Regulator

## 2.9 Antena

Antena pengarah, memodifikasi atau "mengarahkan" sinyal, ke lokasi yang sangat spesifik, tidak hanya meningkatkan penguatan, tetapi juga mengurangi penerimaan atau interferensi yang tidak diinginkan dari perangkat Nirkabel di sekitarnya. Terkadang, antena directional digunakan bahkan dalam situasi di mana pelanggan tidak perlu menyiarkan sinyal di

area yang tepat, tetapi lebih untuk mengurangi atau menghilangkan gangguan terdekat atau sinyal WiFi /Nirkabel pada saluran yang sama.



Gambar 8 Antena

### 2.10 Buzzer

Buzzer adalah perangkat sinyal audio, yang mungkin mekanis, elektromekanis, atau piezoelektrik (disingkat piezo). Penggunaan khas dari buzzer dan pager termasuk perangkat alarm, timer, dan konfirmasi input pengguna seperti klik mouse atau keystroke.

Buzzer digunakan sebagai indikator peringatan kepada pengemudi ketika melebihi kecepatan. Buzzer sebagai saklar elektronik dengan menggunakan transistor NPN, ketika buzzer berlogika 1 dengan adanya arus dari basis ke transistor sehingga terjadi saturasi dan collector emittor hubung singkat lalu arus dari vcc ke buzzer dan ke ground, sedangkan ketika buzzer berlogika 0 basis transistor terjadi cut off dan collector emittor terjadi open circuit.



Gambar 9 Buzzer

### 2.11 Node MCU

Node MCU adalah firmware berbasis Lua open source untuk ESP8266 WiFi SOC dari Espressif dan menggunakan sistem file SPIFFS berbasis flash pada modul. Node MCU diimplementasikan dalam C dan berlapis pada Espressif NON-OS SDK. Node MCU digunakan memproses dan mengolah data yang diterima bagian receiver untuk dapat tersambung SSID / WiFi dan ditampilkan pada Blynk

(dashboard) yang membutuhkan sinyal internet

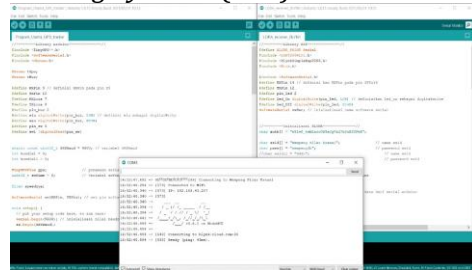


Gambar 10 Node MCU

### 2.12 Perangkat Lunak (Software)

#### A. Arduino IDE

Arduino IDE sebagai software yang digunakan untuk memprogram Arduino Nano, Adafruit LoRa, LoRa Radio, Global Positioning Systems (GPS).



Gambar 11 Arduino IDE.

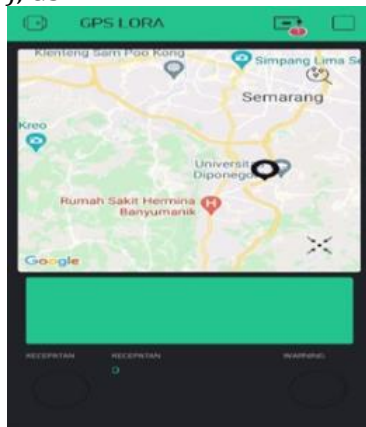
Program Arduino IDE yang dibuat terdiri dari dua program yang saling terhubung sehingga data dapat dipantau melalui serial monitor dan data diteruskan ke software Blynk. Terdapat dua program yaitu : LoRa Receiver Blynk dan GPS LoRa kemudian data tampil pada serial monitor sebagai Database.

#### B. Blynk

Blynk software sebagai dashboard yang digunakan untuk menampilkan data serta memantau kerja alat berdasarkan data yang didapatkan. Blynk dirancang untuk Internet of Things sehingga dapat mengontrol perangkat keras dari jarak jauh, dapat menampilkan data sensor, dapat menyimpan data, memvisualisasikannya, dan melakukan banyak hal keren lainnya.

Fitur - Fitur Blynk antara lain : API & UI serupa untuk semua perangkat keras & perangkat yang didukung, Kumpulan Widget yang mudah digunakan, Mudah diintegrasikan dan menambahkan fungsionalitas baru menggunakan pin virtual Koneksi ke cloud menggunakan: Wifi, Bluetooth dan BLE, Ethernet, USB

(Serial), GSM.



Gambar 12 Blynk

Data yang tampil pada Blynk berupa tampilan *maps* untuk memperlihatkan posisi pemantau (*operator*) sebagai *receiver* dan yang dipantau (*bus*) sebagai *transmitter*, titik hijau menunjukkan lokasi yang dipantau dengan GPS yang terhubung sedangkan titik hitam menunjukkan lokasi pemantau. Terdapat beberapa indikator yaitu untuk nilai kecepatan yang dapat ditampilkan oleh Blynk, indikator peringatan ketika melebihi batas kecepatan dan ketika tombol darurat ditekan, keterangan yang akan tampil ketika indikator peringatan aktif.

### 3. Metodologi Penelitian

Untuk mencapai tujuan agar sesuai yang diharapkan serta pembahasan mengenai penelitian yang dilakukan menjadi terarah, dalam penelitian ini pembahasan dibatasi pada merancang alat yang dapat memonitoring bus, merancang alat berbentuk *Prototype*, dan menggunakan LoRa sebagai komunikasi

#### 1. Tempat dan Waktu Penelitian

Pada salah satu koridor bus Trans Semarang yaitu Koridor VI (Universitas Diponegoro - Universitas Negeri Semarang) dengan jadwal bus dan mobilitas yang tinggi, waktu kedatangan yang lebih lama akan memperlambat mobilitas masyarakat penggunaannya serta perlunya pemantauan untuk keamanan bus.

Penelitian dilakukan di dua tempat yaitu Koridor 6 Bus Trans Semarang (Universitas Diponegoro - Universitas Negeri Semarang) difokuskan pada halte

BNI Polines sampai halte Stadion Undip sebagai tempat pengambilan data pada bus dan Kost Irda Tembalang sebagai pusat pemantauan pada *dashboard*. Waktu penelitian yang akan dilaksanakan oleh penulis selama 70 hari, mulai tanggal 01 Mei 2021 - 10 Juli 2021.

## 2. Desain Penelitian



Gambar 13 Diagram Alir Penelitian

Studi literatur untuk mendapat pemahaman dasar teori dan konsep dari penelitian yang dibahas. Jurnal ilmiah, buku-buku, dan laporan penelitian sebagai literatur agar memperoleh referensi untuk pembahasan juga penulisan penelitian ini seperti yang terdapat dalam daftar pustaka. Analisa kebutuhan merupakan tahapan melakukan riset, mengumpulkan, menghimpun, kemudian mengolah informasi yang telah diperoleh sehingga mendapatkan data untuk melakukan penelitian. Referensi- referensi yang diperoleh akan diaplikasikan dalam proses perancangan alat pada penelitian ini supaya mengetahui bagaimana cara kerja alat yaitu dengan merakit komponen-komponen serta seluruh modul menjadi suatu sistem, melakukan pengujian alat untuk mendapatkan data yang dibutuhkan, dan penulisan.

Perancangan penelitian merupakan tahap untuk merencanakan penelitian dengan spesifik dan rinci seputar bagaimana memproses dan menganalisa data yang diperoleh. *Prototype* dibuat dan

melakukan pengujian alat sebagai metode yang dipakai pada penelitian ini.

### 3.1 Metode Pengumpulan Data

Pengujian alat dilakukan oleh penulis dengan melakukan uji ukur untuk mendapat data kecepatan dan jarak optimum yang dapat dijangkau pada beberapa keadaan yaitu pada keadaan darurat atau tidak, dan ketika pengemudi melebihi kecepatan atau tidak dari alat ini dengan mengambil data pada bus di Koridor VI Bus Trans Semarang (Universitas Diponegoro – Universitas Negeri Semarang) difokuskan pada halte BNI Polines sampai halte Stadion Undip untuk dipantau dari Kost Irda Tembalang sebagai pusat pemantauan pada *dashboard* (Blynk). Terdapat pemantau yang mengamati masuknya data dari alat bagian *Receiver* yaitu posisi bus berupa *latitude* dan *longitude*, kecepatan bus, peringatan kecepatan, respon alat untuk keadaan darurat ketika tombol darurat aktif atau tidak pada *serial monitor* aplikasi Arduino IDE pada laptop dan data terlihat tampil pada Blynk (*dashboard*). Kemudian salah satu dari pemantau lainnya naik ke dalam bus dengan membawa alat bagian *Transmitter* untuk mengambil data ketika bus berjalan dan dikirimkan ke pemantau dengan media LoRa (Long Range).

### 3.2 Metode Analisis Data

Dalam melakukan uji analisa dengan percobaan atau pengujian alat agar dapat melihat bagaimana kinerja alat tersebut. Untuk mencapai tujuan agar sesuai yang diharapkan serta pembahasan mengenai penelitian yang dilakukan menjadi terarah, dalam penelitian ini pembahasan dibatasi pada :

1. Merancang alat yang dapat memonitoring bus.
2. Merancang alat berbentuk Prototype.
3. Menggunakan LoRa sebagai komunikasi jarak jauh dengan frekuensi 915 MHz.
4. Mengetahui kinerja alat berupa : posisi, kecepatan, keamanan.

### 4. Hasil dan Pembahasan

Alat dibuat dan melakukan pengujian alat sebagai metode yang dipakai pada penelitian ini.

Penelitian direncanakan dilakukan di dua tempat yaitu Koridor VI Bus Trans Semarang (Universitas Diponegoro – Universitas Negeri Semarang) difokuskan pada halte BNI Polines sampai halte Stadion Undip untuk dipantau dari operator.

### 4.1 Perancangan Alat

Alat dibuat dan melakukan pengujian alat sebagai metode yang dipakai pada penelitian ini. Perancangan alat yang terdiri dari dua bagian yaitu pengirim data (*transmitter*) dan penerima data (*receiver*) yang digunakan untuk melakukan uji dengan pengambilan data.

#### A. Kebutuhan Perangkat Keras (Hardware)

- Adafruit LoRa : Modul LoRa sebagai *receiver* digunakan untuk menerimadata (*Host*).
- LoRa Radio Modul : Modul LoRa sebagai *transmitter* digunakan untukmengirimdata dan berkomunikasi dengan host (*Client*).
- Arduino Nano : digunakan sebagai media pemrograman, memproses data yangtersedia kemudian data dikirimkan ke modul LoRa.
- Laptop : digunakan untuk, menampilkan data, dan menyimpan datayangdidapatkan.
- Baterai : digunakan sebagai sumber daya pada untuk LoRa *transmitter* yang dapat bekerja di segala tempat ketika melakukan pengujian alat.
- Regulator : digunakan sebagai pengatur tegangan keluaran dari adaptor 5Vmenuju LoRa
- Antena : digunakan sebagai media transmisi dan menerima data darijangkauan frekuensi yang berbedabeda
- *Buzzer* : digunakan sebagai indikator peringatan kepada pengemudi ketika melebihikecepatan
- *Global Positioning Sysytem* (GPS) : digunakan sebagai pengukur kecepatan danpenentuan posisi / lokasi

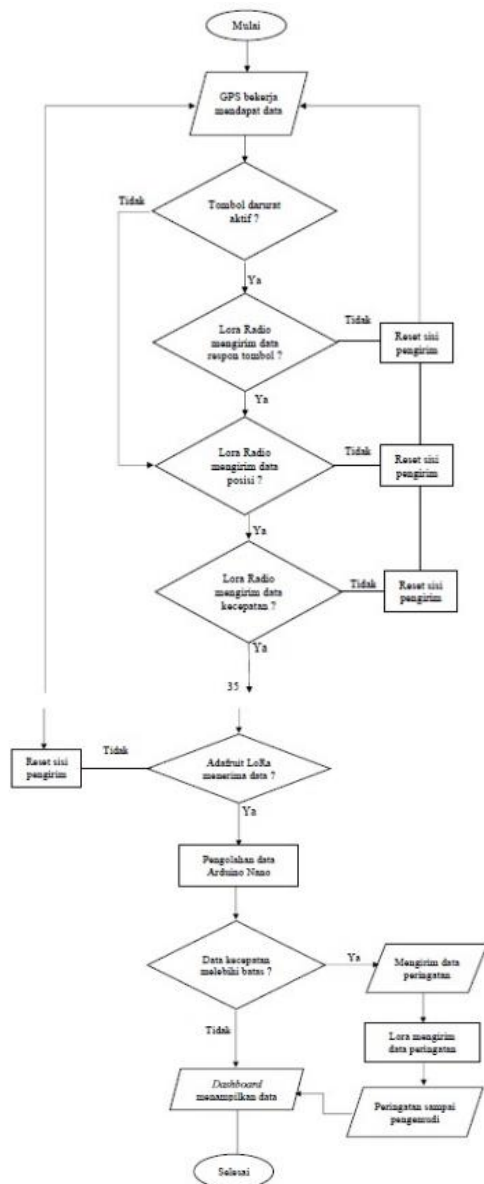
#### B. Kebutuhan Perangkat Lunak (Software)

- Arduino IDE merupakan *software* yang

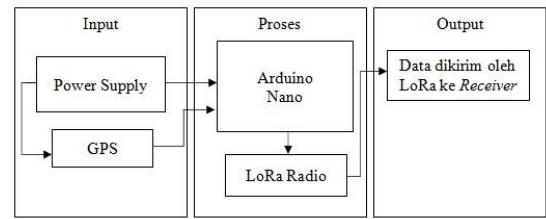
digunakan untuk memprogram Arduino Nano, Adafruit LoRa, LoRa Radio, *Global Positioning Systems*( GPS), .

- Blynk merupakan *software* sebagai *dashboard* yang digunakan untuk menampilkan data serta memantau kerja alat berdasarkan data yang didapatkan.

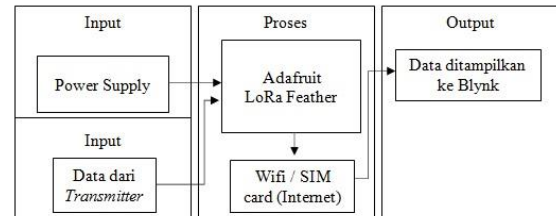
Dalam memudahkan untuk melihat sistem kerja alat ini, jadi disusunlah diagram alir (*flowchart*) sebagai berikut :



Gambar 14 Diagram Alir Alat

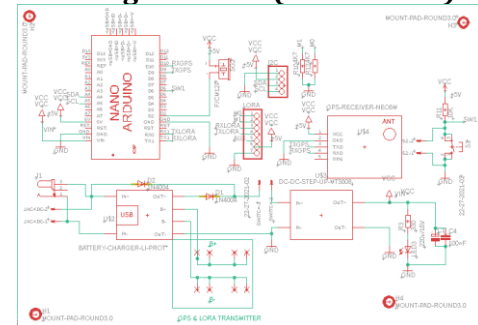


Gambar 15 Blok Diagram Transmitter



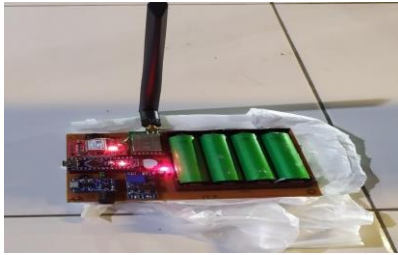
Gambar 16 Blok Diagram Receiver

#### 4.1.2 Pengirim Data (Transmitter)



Gambar 17 Skema Wiring Diagram Transmitter

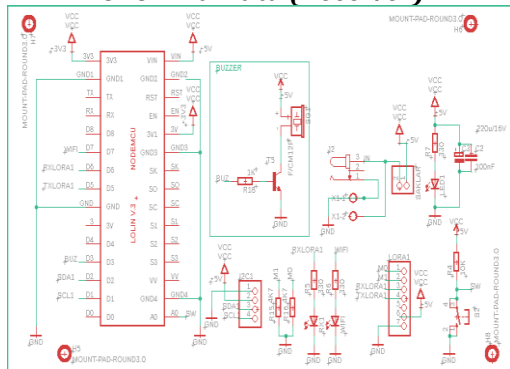
Pada Gambar 17 merupakan *wiring diagram* dari rangkaian LoRa bagian *Transmitter* dengan terdapat komponen yaitu Arduino Nano dengan masing – masing pin, salah satunya Vin untuk masuknya arus dari sumber baterai, pin RX dan TX LoRa sebagai penghubung untuk mengirimkan dan menerima data dari LoRa, pin RX dan TX GPS sebagai penghubung untuk mengirimkan dan menerima data dari GPS. Terdapat *push button* sebagai tombol darurat yang terhubung pada Arduino Nano. Regulator terhubung ke beberapa komponen untuk mengatur tegangan dari sumber yaitu baterai agar tegangan masuk dan keluaran sesuai pada rangkaian, serta *port USB* untuk menghubungkan atau tempat masukkan data dari komputer.



Gambar 18 Bagian Transmitter

Untuk bagian Transmitter gambar 18 sebagai pengirim data membutuhkan beberapa komponen sebagai bahan untuk perancangan alat seperti Arduino Nano berfungsi pengolah dan pemroses data, Modul Adafruit LoRa 915 MHz berfungsi sebagai pengirim data, *Global Positioning System* (GPS) berfungsi untuk mendeteksi posisi dan kecepatan sebagai data, Baterai berfungsi sebagai sumber arus, LoRa Radio Module berfungsi media transmisi pengirim data. Proses pengirim data (*Transmitter*) bekerja diawali dari *Global Positioning System* (GPS) bekerja mendapat data kemudian data diolah oleh Arduino Nano dikirimkan oleh Adafruit LoRa menggunakan LoRa Radio dan antena sebagai media transmisi, lalu seluruh komponen mendapat sumber arus dari baterai.

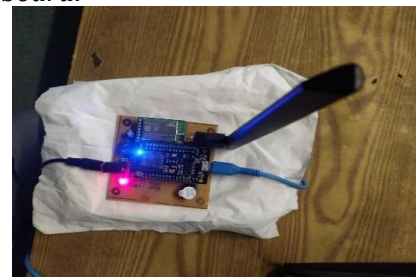
#### 4.4.1.2 Penerima Data (Receiver)



Gambar 19 Skema Wiring Diagram Receiver

Pada Gambar 19 merupakan wiring diagram dari rangkaian LoRa bagian Receiver dengan terdapat komponen yaitu Adafruit LoRa Feather dengan masing – masing pin, salah satunya Vin untuk masuknya arus dari sumber, pin RX dan TX LoRa sebagai penghubung untuk mengirimkan dan menerima data dari LoRa. Terdapat pin D3 yang terhubung ke buzzer sebagai saklar elektronik. Buzzer sebagai saklar elektronik pada rangkaian

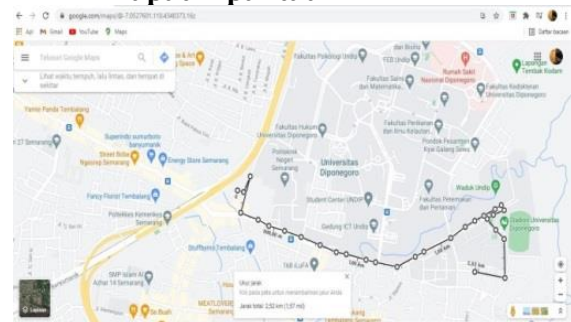
berarti menggunakan prinsip transistor NPN, jadi ketika buzzer berlogika 1 maka ada arus dari basis ke transistor sehingga terjadi saturasi sedangkan *collector emittor* mengalami hubung singkat, kemudian ketika buzzer berlogika 0 maka arus dari vcc ke *ground* sehingga basis transistor mengalami *cut off* sedangkan *collector emittor* mengalami *open circuit*. Pin D7 menghubungkan pada wifi untuk mendapatkan sinyal wifi sehingga Node MCU dapat meneruskan data yang telah diolah dan diperoleh menuju Blynk sebagai *dashboard*.



Gambar 20 Bagian Receiver

Untuk bagian Receiver pada Gambar 20 sebagai penerima data membutuhkan beberapa komponen sebagai bahan untuk perancangan alat seperti Modul Adafruit LoRa 915 MHz berfungsi sebagai penerima data, LoRa Radio Module berfungsi media transmisi penerima data. Proses penerima data (*Receiver*) bekerja diawali dari Adafruit LoRa menggunakan LoRa Radio dan antena sebagai media transmisi menerima data dari pengirim data (*Transmitter*) yang telah dproses oleh Adafruit LoRa, kemudian data dimasukkan ke program untuk ditampilkan pada *dashboard* (Blynk).

#### 4.1.3 Pengujian Posisi dan Jarak yang Dapat Dipantau



Gambar 21 Tampilan Peta Jarak yang Ditempuh

Pengujian yang dilakukan seperti Gambar 21 untuk memantau posisi dari bus dan mengetahui jarak maksimal yang dapat dipantau oleh alat yang diukur dari posisi pemantau dengan pergerakan posisi bus seberapa jauh. Uji alat dilakukan pada Bus Trans Semarang koridor 6 (Universitas Diponegoro – Universitas Negeri Semarang) dari halte BNI Polines sampai halte Stadion Undip berjarak 2500 meter (2,5 km) dan kembali lagi. Bagian *transmitter* dibawa naik ke dalam bus untuk mendapatkan data posisi dari bus ketika berjalan sesuai dari halte BNI Polines sampai halte Stadion Undip dengan jarak kurang lebih 2500 meter (2,5 km) dan kembali lagi. Bagian *receiver* menerima data yang dikirimkan berupa posisi dari pemantau dan bus menggunakan *latitude* dan *longitude* dari posisi tersebut sehingga dapat juga diketahui jarak yang dapat diukur dari yang ditempuh bus. Dalam uji alat juga mengamati *delay* dalam pengiriman data dari posisi.

```
17:16:45.785 -> Longitude = 110.431610
17:16:45.838 -> latitude = -7.052449
17:16:45.838 -> Kecepatan = 8.90
17:16:45.885 -> Warning = 0
17:16:45.885 -> Max kecepatan = 0
17:16:50.802 -> Longitude = 110.431571
17:16:50.802 -> latitude = -7.052474
17:16:50.855 -> Kecepatan = 14.45
17:16:50.855 -> Warning = 0
17:16:50.855 -> Max kecepatan = 0
17:16:55.775 -> Longitude = 110.431580
17:16:55.822 -> latitude = -7.052523
17:16:55.822 -> Kecepatan = 19.43
17:16:55.875 -> Warning = 0
17:16:55.875 -> Max kecepatan = 0
17:17:00.795 -> Longitude = 110.431587
17:17:00.842 -> latitude = -7.052547
17:17:00.842 -> Kecepatan = 22.23
17:17:00.842 -> Warning = 0
17:17:00.896 -> Max kecepatan = 0
17:17:05.816 -> Longitude = 110.431602
17:17:05.816 -> latitude = -7.052590
17:17:05.863 -> Kecepatan = 26.10
17:17:05.863 -> Warning = 0
17:17:05.863 -> Max kecepatan = 0
17:17:10.786 -> Longitude = 110.432603
17:17:10.834 -> latitude = -7.052299
17:17:10.834 -> Kecepatan = 21.25
17:17:10.881 -> Warning = 0
17:17:10.881 -> Max kecepatan = 0
17:17:15.804 -> Longitude = 110.432582
17:17:15.804 -> latitude = -7.052608
17:17:15.852 -> Kecepatan = 10.69
17:17:15.852 -> Warning = 0
17:17:15.852 -> Max kecepatan = 0
17:17:20.823 -> Longitude = 110.432303
17:17:20.823 -> latitude = -7.053247
17:17:20.823 -> Kecepatan = 13.96
17:17:20.871 -> Warning = 0
17:17:20.871 -> Max kecepatan = 0
```

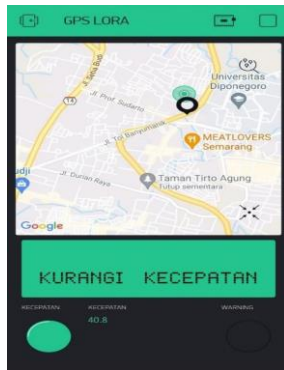
Dari pengujian alat untuk posisi dan jarak, didapatkan data dari *serial monitor*

seperti di atas bahwa terdapat data posisi dari pemantau dan bus berupa posisi *latitude* dan *longitude* sehingga dapat diukur dan diketahui jarak dari pemantau sampai pergerakan posisi bus yang dapat diukur. Data posisi ditampilkan pada *serial monitor* setiap 5 detik yaitu *longitude* dan *latitude* dari posisi bus. Untuk data posisi bus terjauh yang dapat terukur pada alat ialah pada posisi -7.054684, 110.445838 dengan jarak 2200 meter dari posisi pemantau -7.052978, 110.431972, hal ini disebabkan karena adanya hambatan berupa pepohonan yang cukup banyak, jaringan telekomunikasi yang cukup padat, kepadatan bangunan pada wilayah tersebut. *Delay* pada pengiriman dan penerimaan data terjadi setiap data yang masuk pada *serial monitor* berbeda – beda dari 0,8; 5,8; 5; 10; 10,8 pada data terinput setiap 5 detik, jika *delay* 10 detik berarti data yang terinput masih sama seperti sebelumnya. Jarak yang semakin jauh berbanding lurus dengan hambatan yang akan memberi pengaruh pada transmisi sinyal untuk pengirim dan penerimaan data dan menyebabkan *delay* dikarenakan terhambat untuk perambatan sinyalnya.

#### 4.1.4 Pengujian Kinerja Alat Mendeteksi Kecepatan

Pengujian yang dilakukan untuk mendeteksi kecepatan maksimal dari bus yang dapat dipantau oleh alat dan memberi sinyal peringatan pada bus ketika kecepatannya melebihi batas kecepatan yang ditentukan. Batas kecepatan pada pengujian alat diatur sebesar 40 Km/Jam, sehingga saat bus melaju pada kecepatan yang terukur pada alat melebihi 40 Km/Jam akan ada indikator peringatan berupa *buzzer* yang berbunyi sehingga pengemudi bus dapat mengetahui apabila bus melebihi batas kecepatan dan terdapat notifikasi yang muncul pada pemantau yaitu "Kurangi Kecepatan" juga kecepatan yang melebihi batas maka pemantau dapat mengetahui dan memperingatkan pengemudi.





Gambar 22 Tampilan Kecepatan dan Peringatan Kecepatan pada Blynk

Pada Gambar 22 menunjukkan uji alat kecepatan dan peringatan kecepatan yang dilakukan pada Bus Trans Semarang koridor 6 (Universitas Diponegoro – Universitas Negeri Semarang) dari halte BNI Polines sampai halte Stadion Undip berjarak 2500 meter (2,5 km) dan kembali lagi sama seperti pengujian sebelumnya yang dilakukan bersamaan. Bagian *transmitter* dibawa naik ke dalam bus untuk mendapatkan data kecepatan dari bus ketika berjalan sesuai dari halte BNI Polines sampai halte Stadion Undip dengan jarak kurang lebih 2500 meter (2,5 km) dan kembali lagi juga terdapat peringatan berupa *buzzer* yang berbunyi ketika kecepatan bus yang terukur alat ketika melebihi batas kecepatan yang diatur.

```

17:23:12.236 -> Longitude = 110.439394
17:23:12.236 -> latitude = -7.056026
17:23:12.236 -> Kecepatan = 10.96
17:23:12.282 -> Warning = 0
17:23:12.282 -> Max kecepatan = 0
17:23:17.202 -> Longitude = 110.440135
17:23:17.256 -> latitude = -7.056132
17:23:17.256 -> Kecepatan = 14.81
17:23:17.303 -> Warning = 0
17:23:17.303 -> Max kecepatan = 0
17:23:22.214 -> Longitude = 110.440854
17:23:22.214 -> latitude = -7.055930
17:23:22.261 -> Kecepatan = 22.64
17:23:22.261 -> Warning = 0
17:23:22.313 -> Max kecepatan = 0
17:23:27.220 -> Longitude = 110.441444
17:23:27.220 -> latitude = -7.055653
17:23:27.267 -> Kecepatan = 28.31
17:23:27.267 -> Warning = 0
17:23:27.267 -> Max kecepatan = 0
17:23:32.195 -> Longitude = 110.442377
17:23:32.231 -> latitude = -7.055206
17:23:32.278 -> Kecepatan = 34.17
17:23:32.278 -> Warning = 0
17:23:32.278 -> Max kecepatan = 0
17:23:37.206 -> Longitude = 110.442710
17:23:37.242 -> latitude = -7.055057
17:23:37.242 -> Kecepatan = 40.56
17:23:37.289 -> Warning = 0
17:23:37.289 -> Max kecepatan = 1
17:23:42.207 -> Longitude = 110.443074
17:23:42.242 -> latitude = -7.054854
17:23:42.242 -> Kecepatan = 45.62
    
```

```

17:23:42.289 -> Warning = 0
17:23:42.289 -> Max kecepatan = 1
    
```

Dari pengujian alat untuk kecepatan dan peringatan kecepatan, didapatkan data dari *serial monitor* seperti di atas bahwa terdapat data kecepatan bus yang dapat dipantau oleh alat dengan kecepatan maksimal yang terpantau adalah 46 Km/Jam. Kecepatan yang terukur oleh alat ketika melebihi batas kecepatan yang diatur pada alat maka akan ada peringatan yang muncul pada bagian *transmitter* yaitu *buzzer* akan berbunyi sebagai pengingat bahwa melebihi batas kecepatan dan terdapat notifikasi masuk berupa tulisan “Kurangi Kecepatan” juga indikator batas kecepatan yang menyala untuk pemantau pada Blynk sebagai *dashboard* pemantau. Data kecepatan ditampilkan sama seperti dan bersamaan seperti pengujian sebelumnya pada *serial monitor* setiap 5 detik yaitu besarnya kecepatan yang terukur alat dan jumlah peringatan batas kecepatan yang muncul. Untuk data kecepatan bus maksimal yang dapat terukur pada alat ialah 46 Km/Jam pada posisi bus -7.054317, 110.444194, hal ini disebabkan karena adanya hambatan berupa pepohonan yang cukup banyak, jaringan telekomunikasi yang cukup padat, juga pergerakan bus yang cepat menyebabkan kekuatan transmisi data berkurang dan pengukuran kecepatan kurang akurat. *Delay* pada pengiriman dan penerimaan data terjadi setiap data yang masuk pada *serial monitor* berbeda – beda dari 0,8; 5,8; 5; 10; pada data terinput setiap 5 detik, jika *delay* 10 detik berarti data yang terinput masih sama seperti sebelumnya.

#### 4.1.5 Pengujian Respon Tombol Darurat Pada Bus

Pengujian yang dilakukan untuk mengamati respon alat ketika tombol darurat ditekan pada uji alat pada bus saat berjalan sesuai rute. Dalam pengujian respon alat pada keadaan darurat disimulasikan ketika bus sedang berhenti dan dimisalkan bus berhenti dikarenakan kerusakan mesin juga mati mesin sehingga posisi bus pada Blynk (*dashboard*) tidak berubah, kemudian disimulasikan

pengemudi bus menekan tombol darurat pada alat sehingga alat akan merespon dan mengirimkan sinyal untuk memberi notifikasi kepada pemantau pada Blynk (*dashboard*) bahwa bus dalam keadaan darurat.



Gambar 23 Pemantauan Data Masuk pada Laptop

Pada Gambar 23 ditunjukkan bahwa ketika uji alat dilakukan pada Bus Trans Semarang koridor 6. Bagian *transmitter* dibawa naik ke dalam bus untuk mensimulasikan keadaan darurat dari bus ketika berhenti pada beberapa lokasi sesuai dari halte BNI Polines sampai halte Stadion Undip dengan jarak kurang lebih 2500 meter (2,5 km) dengan mengaktifkan tombol darurat setiap 200 meter. Bagian *receiver* dipantau dari Kost Irda Tembalang untuk mengamati data yang masuk pada laptop dari menerima data yang dikirimkan berupa posisi dari bus menggunakan *latitude* dan *longitude* dari posisi bus sehingga dapat diketahui posisi bus ketika keadaan darurat, juga dapat diukur jarak pantauan bus dalam mengirim sinyal keadaan darurat. Keadaan darurat yang dimaksud adalah ketika bus mesin mati atau kerusakan mesin, ketika terjadi kecelakaan, dengan diketahui saat pengemudi menekan tombol darurat sehingga terdapat peringatan masuk pada *dashboard* dan ada selang waktu posisi bus berhenti pada peta pemantau selama 1 – 5 menit. Dalam uji alat juga mengamati *delay* dalam respon alat ketika tombol darurat aktif.

```
17:33:53.813 -> Longitude = 110.432675
17:33:53.813 -> latitude = -7.052414
17:33:53.866 -> Kecepatan = 28.47
17:33:53.866 -> Warning = 1
17:33:53.866 -> Max kecepatan = 0
17:33:58.790 -> Longitude = 110.436045
17:33:58.836 -> latitude = -7.055365
17:33:58.836 -> Kecepatan = 31.63
17:33:58.836 -> Warning = 1
17:33:58.890 -> Max kecepatan = 0
17:34:03.810 -> Longitude = 110.435616
```

```
17:34:03.810 -> latitude = -7.05523
17:34:03.857 -> Kecepatan = 31.63
17:34:03.857 -> Warning = 1
17:34:03.857 -> Max kecepatan = 0
17:34:08.780 -> Longitude = 110.435526
17:34:08.834 -> latitude = -7.055152
17:34:08.834 -> Kecepatan = 34.87
17:34:08.880 -> Warning = 1
17:34:08.880 -> Max kecepatan = 0
17:34:13.785 -> Longitude = 110.434800
17:34:13.838 -> latitude = -7.054940
17:34:13.838 -> Kecepatan = 36.63
17:34:13.838 -> Warning = 1
17:34:13.885 -> Max kecepatan = 0
17:34:17.793 -> Longitude = 110.434414
17:34:17.793 -> latitude = -7.054780
17:34:17.846 -> Kecepatan = 32.47
17:34:17.846 -> Warning = 1
17:34:17.893 -> Max kecepatan = 0
17:34:23.801 -> Longitude = 110.434199
17:34:23.801 -> latitude = -7.054737
17:34:23.855 -> Kecepatan = 13.56
17:34:23.855 -> Warning = 1
17:34:23.900 -> Max kecepatan = 0
17:34:28.794 -> Longitude = 110.431579
17:34:28.841 -> latitude = -7.052669
17:34:28.841 -> Kecepatan = 8.90
17:34:28.841 -> Warning = 1
17:34:28.879 -> Max kecepatan = 0
17:34:33.804 -> Longitude = 110.431579
17:34:33.804 -> latitude = -7.052669
17:34:33.851 -> Kecepatan = 12.53
17:34:33.851 -> Warning = 1
17:34:33.851 -> Max kecepatan = 0
17:34:38.801 -> Longitude = 110.431579
17:34:38.801 -> latitude = -7.052669
17:34:38.848 -> Kecepatan = 12.72
17:34:38.848 -> Warning = 1
17:34:38.886 -> Max kecepatan = 0
17:34:43.801 -> Longitude = 110.431579
17:34:43.801 -> latitude = -7.052669
17:34:43.847 -> Kecepatan = 16.48
17:34:43.847 -> Warning = 1
17:34:43.901 -> Max kecepatan = 0
17:34:48.811 -> Longitude = 110.431579
17:34:48.811 -> latitude = -7.052669
17:34:48.858 -> Kecepatan = 19.24
17:34:48.858 -> Warning = 1
17:34:48.858 -> Max kecepatan = 0
17:34:53.800 -> Longitude = 110.431579
17:34:53.800 -> latitude = -7.052669
17:34:53.836 -> Kecepatan = 20.73
17:34:53.836 -> Warning = 1
17:34:53.883 -> Max kecepatan = 0
17:34:58.779 -> Longitude = 110.431579
17:34:58.832 -> latitude = -7.052669
17:34:58.832 -> Kecepatan = 20.73
17:34:58.878 -> Warning = 1
17:34:58.878 -> Max kecepatan = 0
17:35:03.792 -> Longitude = 110.431579
17:35:03.846 -> latitude = -7.052669
17:35:03.846 -> Kecepatan = 17.41
17:35:03.846 -> Warning = 1
17:35:03.846 -> Max kecepatan = 0
17:35:08.813 -> Longitude = 110.431579
17:35:08.813 -> latitude = -7.052669
17:35:08.866 -> Kecepatan = 25.66
17:35:08.866 -> Warning = 1
17:35:08.866 -> Max kecepatan = 0
17:35:13.785 -> Longitude = 110.431579
17:35:13.832 -> latitude = -7.052669
17:35:13.832 -> Kecepatan = 37.82
17:35:13.885 -> Warning = 2
17:35:13.885 -> Max kecepatan = 0
17:35:17.805 -> Longitude = 110.431579
17:35:17.805 -> latitude = -7.052669
17:35:17.852 -> Kecepatan = 34.78
17:35:17.852 -> Warning = 2
17:35:17.852 -> Max kecepatan = 0
17:35:23.788 -> Longitude = 110.431579
17:35:23.825 -> latitude = -7.052669
```

17:35:23.825 -> Kecepatan = 28.94  
 17:35:23.872 -> Warning = 2  
 17:35:23.872 -> Max kecepatan = 0  
 17:35:28.789 -> Longitude = 110.431579  
 17:35:28.789 -> latitude = -7.052669  
 17:35:28.842 -> Kecepatan = 32.76  
 17:35:28.842 -> Warning = 2  
 17:35:28.889 -> Max kecepatan = 0  
 17:35:33.805 -> Longitude = 110.431579  
 17:35:33.805 -> latitude = -7.052669  
 17:35:33.858 -> Kecepatan = 29.12  
 17:35:33.858 -> Warning = 2  
 17:35:33.858 -> Max kecepatan = 0  
 17:35:38.778 -> Longitude = 110.431579  
 17:35:38.824 -> latitude = -7.052669  
 17:35:38.824 -> Kecepatan = 26.63  
 17:35:38.877 -> Warning = 2  
 17:35:38.877 -> Max kecepatan = 0  
 17:35:43.799 -> Longitude = 110.431579  
 17:35:43.799 -> latitude = -7.052669  
 17:35:43.846 -> Kecepatan = 27.89  
 17:35:43.846 -> Warning = 2  
 17:35:43.846 -> Max kecepatan = 0

Dari data *serial monitor* seperti di atas untuk mengetahui respon alat untuk fungsi tombol darurat, didapatkan data bahwa terlihat respon alat ketika tombol darurat diaktifkan setiap 200 meter pada simulasi bus dalam keadaan darurat saat bus berhenti. Tombol darurat ditekan maka akan ada data yang masuk pada *serial monitor* baik dari posisi bus berupa *latitude* dan *longitude* sehingga dapat diketahui jarak dari pemantau ke bus, jumlah notifikasi darurat (*warning*).



Gambar 24 Tampilan Peringatan Darurat pada Blynk

Pada Gambar 4.24 ditunjukkan terdapat notifikasi masuk berupa tulisan “Keadaan Darurat” juga indikator darurat (*warning*) yang menyala untuk pemantau pada Blynk sebagai *dashboard* pemantau. Data jarak dari posisi bus yang dapat terpantau untuk keadaan darurat adalah tidak lebih dari 2200 meter sama seperti dan bersamaan seperti pengujian sebelumnya pada *serial monitor* setiap 5 detik yaitu *latitude* dan *longitude* serta jumlah notifikasi darurat (*warning*). Delay pada pengiriman dan penerimaan data terjadi setiap data yang masuk pada *serial monitor* berbeda – beda

dari 0,8; 5,8; 5; 10. Delay dan keterbatasan jarak pantau sejauh 2200 meter disebabkan karena adanya hambatan berupa pepohonan yang cukup banyak, jaringan telekomunikasi yang cukup padat, juga pergerakan bus yang cepat menyebabkan kekuatan transmisi data berkurang dan pengukuran kecepatan kurang akurat. Jarak yang semakin jauh berbanding lurus dengan hambatan yang akan memberi pengaruh pada transmisi sinyal untuk pengirim dan penerimaan data sehingga menyebabkan delay dikarenakan terhambat untuk perambatan sinyalnya.

#### 4.2 Pembahasan

Alat berupa *Prototype* yang dirancang menjadi dua bagian yaitu pengirim data (*transmitter*) dan penerima data (*receiver*) yang digunakan untuk melakukan uji dengan pengambilan data memanfaatkan LoRa (*Long Range*) sebagai media komunikasi.



Gambar 25 Transmitter dan Receiver

Tabel 2 Tabel Data Posisi dan Jarak yang Dipantau

Posisi Pemantau		Posisi Bus		Jarak (m)	Delay (Detik)	Ket
Longitude	Latitude	Longitude	Latitude			
110.431972	-7.052978	110.432675	-7.052414	100	0.8	Alat Bekerja
110.431972	-7.052978	110.432407	-7.053031	102	0.8	Alat Bekerja
110.431972	-7.052978	110.432269	-7.053473	217	0.8	Alat Bekerja
110.431972	-7.052978	110.432151	-7.053926	306	5,0	Alat Bekerja
110.431972	-7.052978	110.433063	-7.054224	409	5,0	Alat Bekerja
110.431972	-7.052978	110.434194	-7.054591	522	5,0	Alat Bekerja
110.431972	-7.052978	110.434967	-7.054825	611	0.8	Alat Bekerja
110.431972	-7.052978	110.435997	-7.055193	737	0.8	Alat Bekerja
110.431972	-7.052978	110.436775	-7.055390	826	0.8	Alat Bekerja
110.431972	-7.052978	110.437687	-7.055693	921	0.8	Alat Bekerja

110.431972	-7.052978	110.438454	-7.055847	1014	0.8	Alat Bekerja
110.431972	-7.052978	110.438937	-7.055890	1122	5.8	Alat Bekerja
110.431972	-7.052978	110.440492	-7.056007	1241	5.8	Alat Bekerja
110.431972	-7.052978	110.441045	-7.055815	1304	5.0	Alat Bekerja
110.431972	-7.052978	110.441554	-7.055587	1398	0.8	Alat Bekerja
110.431972	-7.052978	110.442558	-7.055102	1511	10.0	Alat Bekerja
110.431972	-7.052978	110.443389	-7.054708	1619	0.8	Alat Bekerja
110.431972	-7.052978	110.444172	-7.054330	1712	5.8	Alat Bekerja
110.431972	-7.052978	110.445122	-7.053883	1829	5.8	Alat Bekerja
110.431972	-7.052978	110.445379	-7.053760	1908	10.8	Alat Bekerja
110.431972	-7.052978	110.445245	-7.053973	2047	10.0	Alat Bekerja
110.431972	-7.052978	110.445460	-7.054663	2154	10.0	Alat Bekerja
110.431972	-7.052978	110.445838	-7.054684	2200	10.0	Alat Bekerja
110.431972	-7.052978	-	-	2300	-	Tak Bekerja
110.431972	-7.052978	-	-	2400	-	Tak Bekerja

Pada Tabel 2 menunjukkan data posisi bus terjauh yang dapat terukur pada alat ialah pada posisi -7.054684, 110.445838 dengan jarak 2200 meter dari posisi pemantau -7.052978, 110.431972, hal ini disebabkan karena adanya hambatan berupa pepohonan, jaringan telekomunikasi yang cukup padat, kepadatan bangunan pada wilayah tersebut.

*Delay* pada pengiriman dan penerimaan data terjadi setiap data yang masuk pada *serial monitor* berbeda – beda dari 0,8; 5,8; 5; 10; 10,8. Jarak yang semakin jauh berbanding lurus dengan hambatan yang akan memberi pengaruh pada transmisi sinyal untuk pengirim dan penerimaan data dan menyebabkan *delay* dikarenakan terhambat untuk perambatan sinyalnya.

Tabel 3 Tabel Data Kecepatan dan Batas Kecepatan yang Dipantau

Posisi Bus		Kecepatan Bus (Km/jam)	Peringatan Kecepatan	Delay (S)	Ket
Longitude	Latitude				
110.432660	-7.052629	0.00	Tak Aktif	0.8	Alat Bekerja
110.432603	-7.052299	0.00	Tak Aktif	5,0	Alat Bekerja
110.432582	-7.052608	3.85	Tak Aktif	5.8	Alat Bekerja
110.432274	-7.053694	8.34	Tak Aktif	5.8	Alat Bekerja
110.432177	-7.053918	17.12	Tak Aktif	5,0	Alat Bekerja
110.432446	-7.053971	21.72	Tak Aktif	5,0	Alat Bekerja

110.432692	-7.054088	20.67	Tak Aktif	0.8	Alat Bekerja
110.434806	-7.054833	13.96	Tak Aktif	0,8	Alat Bekerja
110.435975	-7.055153	10.69	Tak Aktif	5,0	Alat Bekerja
110.437070	-7.055693	21.72	Tak Aktif	0.8	Alat Bekerja
110.439162	-7.055909	23.35	Tak Aktif	0.8	Alat Bekerja
110.440135	-7.056132	14.81	Tak Aktif	0.8	Alat Bekerja
110.440854	-7.055930	22.64	Tak Aktif	0.8	Alat Bekerja
110.441444	-7.055653	28.31	Tak Aktif	5,0	Alat Bekerja
110.442377	-7.055206	34.17	Tak Aktif	5,0	Alat Bekerja
110.442710	-7.055057	40.56	Aktif	5,0	Alat Bekerja
110.443074	-7.054854	45.62	Aktif	10	Alat Bekerja
110.444172	-7.054330	40.8	Aktif	5,8	Alat Bekerja
110.444870	-7.054008	31.08	Tak Aktif	5,8	Alat Bekerja
110.445803	-7.053699	19.65	Tak Aktif	10,0	Alat Bekerja

Pada Tabel 3 menunjukkan data kecepatan bus yang dapat dipantau oleh alat dengan kecepatan maksimal yang terpantau adalah 46 Km/Jam posisi bus -7.054317, 110.444194. Kecepatan yang terukur oleh alat ketika melebihi batas kecepatan yang diatur pada alat maka akan ada peringatan muncul bagian *transmitter* pengingat bahwa melebihi batas kecepatan yang ditentukan.

*Delay* pada pengiriman dan penerimaan data terjadi setiap data yang masuk pada *serial monitor* berbeda – beda dari 0,8; 5,8; 5; 10. Kecepatan yang semakin cepat berbanding lurus dengan hambatan yang akan memberi pengaruh pada transmisi sinyal untuk TX dan RX data juga akurasi pengukuran kecepatan dan menyebabkan *delay* dikarenakan terhambat untuk perambatan sinyalnya.

Tabel 4 Tabel Data Respon Alat Tombol Darurat

Posisi Bus		Jarak Pantauan (Meter)	Peringatan Darurat	Delay (Detik)
Longitude	Latitude			
110.432647	-7.052408	200	Aktif	0,8
110.433057	-7.054218	400	Aktif	0,8
110.434963	-7.054815	600	Aktif	5,0
110.436765	-7.055380	800	Aktif	5,8
110.438444	-7.055837	1000	Aktif	5,0
110.440472	-7.056023	1200	Tak Aktif	10,8
110.441538	-7.055578	1400	Aktif	5,0
110.443379	-7.054716	1600	Aktif	5,8
110.445092	-7.0538183	1800	Tak Aktif	10,8
110.445235	-7.053963	2000	Aktif	10,0
110.445828	-7.054674	2200	Aktif	10,0

-	-	2400	Tak Aktif	-
---	---	------	-----------	---

Pada Tabel 4 menunjukkan data jarak dari posisi bus yang dapat terpantau untuk keadaan darurat adalah tidak lebih dari 2200 meter dengan dilakukan pengujian penekanan tombol darurat setiap 200 meter. Delay pada pengiriman dan penerimaan data terjadi setiap data yang masuk pada *serial monitor* berbeda – beda dari 0,8; 5,8; 5; 10.

Delay dan keterbatasan jarak pantau sejauh 2200 meter serta terdapat jarak pantau yang tidak dapat mengirimkan data disebabkan karena adanya hambatan berupa pepohonan yang cukup banyak, jaringan telekomunikasi yang cukup padat, juga pergerakan bus yang cepat menyebabkan kekuatan transmisi data berkurang dan pengukuran kecepatan kurang akurat.

## 5. Simpulan

Berdasarkan data yang diperoleh dari pengujian alat pada penelitian yang berjudul “Implementasi Sistem Monitoring Bus Trans Semarang Berbasis LoRa (Long Range)” dapat disimpulkan bahwa :

1. Alat berupa *Prototype* terdiri dari dua bagian yaitu pengirim data (*transmitter*) dan penerima data (*receiver*) yang dapat memantau posisi, kecepatan, memberi peringatan batas kecepatan, memberi respon ketika bus dalam keadaan darurat sehingga dapat memprediksi kedatangan bus pada halte.
2. Pengujian alat untuk data posisi bus terjauh yang dapat terukur pada alat ialah pada posisi -7.054684, 110.445838 dengan jarak 2200 meter dari posisi pemantau - 7.052978, 110.431972, sama halnya data untuk keadaan darurat dapat merespon adalah tidak lebih dari 2200 meter.
3. Data kecepatan bus yang dapat dipantau oleh alat dengan kecepatan maksimal yang terpantau adalah 46 Km/Jam posisi bus -7.054317, 110.444194. Kecepatan yang terukur oleh alat ketika melebihi batas kecepatan yang diatur pada alat maka akan ada peringatan yang muncul pada bagian *transmitter* yaitu *buzzer* akan berbunyi dan

notifikasi masuk berupa tulisan “Kurangi Kecepatan” juga indikator batas kecepatan yang menyala untuk pemantau pada Blynk sebagai *dashboard* pemantau.

## DAFTAR PUSTAKA

- A.Pasaribu, The Use of Geospatial Technology to Monitor the Transportation of Crude Palm Oil Trucks., 2020.
- Ahmad Roihan, Muhammad Sri Bintang Prasetyo, A. R. (2017). Monitoring Location Tracker Untuk Kendaraan Berbasis Raspbery Pi, 3(2), 148–161.
- Alfeno, S., Eka, R., & Devi, C. (2017). Implementasi Global Positioning System (GPS) dan Location Based Service (LSB) pada Sistem Informasi Kereta Api untuk Wilayah Jabodetabek. Sisfotek Global, 7(2), 29.
- Ardiani, Y. mila. (2019). Fakultas Teknik – Universitas Muria Kudus. Prosiding SNATIFKe-6 Tahun 2019, 5 (2007), 96–101.
- Dan, L., Pada, W., Angkut, K., Jimmy, M., Arifianto, F., Muhim, D. M., Rosyidi, E., Astra, P. M., Gaya, J., Raya, M., & Ii, S. (2020). Rancang bangun sistem pemantauan lokasi berbasis gps, lora dan wifi pada kendaraan angkut perkebunan. September, 63–69.
- Hariyanto, F., & Widyantara, H. (2008). Sistem Pemantauan Posisi Mobil Menggunakan Global Positioning System ( GPS ) Berbasis Radio Frekuensi, (1), 11–20.
- Nengsi, S. R. I. W., Sains, F., Teknonogi, D. A. N., UIN Alauddin Makassar. (2019). Monitoring Kendaraan Menggunakan Long Range (LORA) Radio Frekuensi Berbasis Web.
- Putra, R. W., Rahayu, Y., Teknik, F., Riau, U., & Widya, K. B. (2019). Jom FTeknik Volume 6 Edisi 2 Juli s / d Desember 2019 Jom FTeknik Volume 6 Edisi 2 Juli s/d Desember 2019. 6, 1–9.
- Wikanta, P., Wijanarko, H., & Palufi, Z. (2014). Pemantauan Posisi Armada Bus Pada Busway Berbasis Android, E-59 E-60. 59–64.