

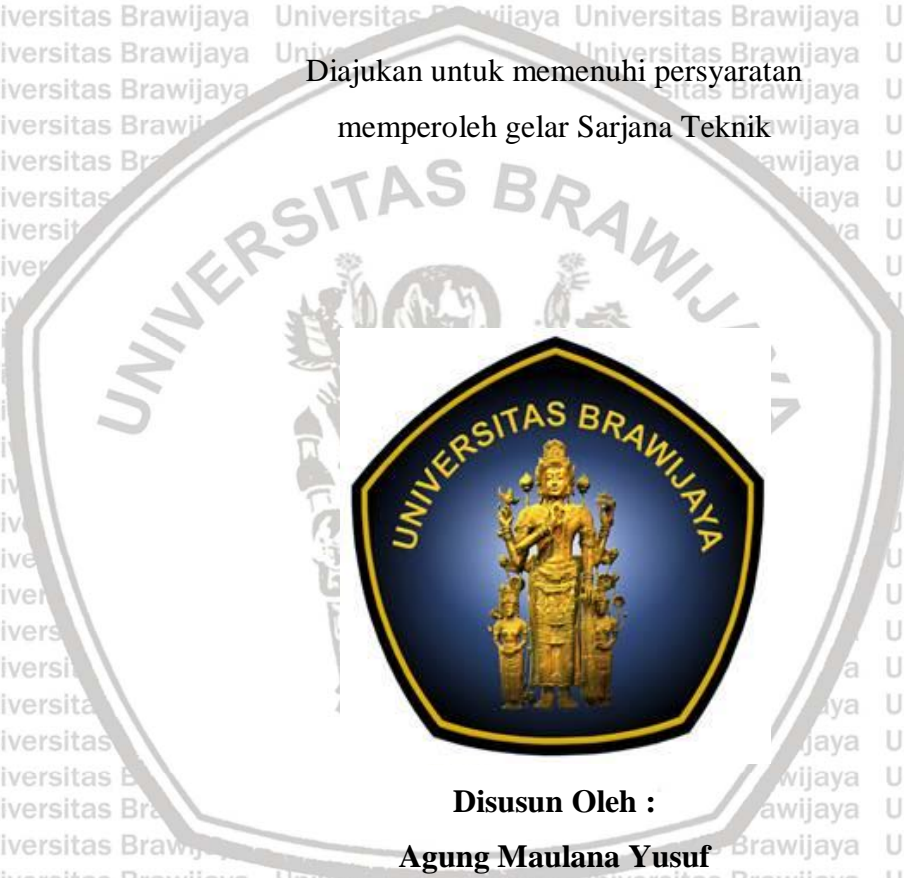
**Analisis Tingkat Akurasi Implementasi Metode *Great Circle*
Distance GPS Ublox Neo-6M Pada Sistem Pelacakan Dan
Deteksi Sepeda Motor Dari Pencurian**

SKRIPSI

TEKNIK ELEKTRO

KONSENTRASI TEKNIK KONTROL

Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik



Disusun Oleh :

Agung Maulana Yusuf

NIM. 175060300111032

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN

UNIVERSITAS BRAWIJAYA

FAKULTAS TEKNIK

JURUSAN TEKNIK ELEKTRO

MALANG

2021



LEMBAR PENGESAHAN

**Analisis Tingkat Akurasi Implementasi Metode *Great Circle Distance* GPS Ublox Neo-6M
Pada Sistem Pelacakan Dan Deteksi Sepeda Motor Dari Pencurian**

SKRIPSI

TEKNIK ELEKTRO KONSENTRASI TEKNIK KONTROL

Ditujukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Teknik



Agung Maulana Yusuf

NIM. 175060300111032

Skripsi ini telah diperiksa dan disetujui oleh dosen
pembimbing pada tanggal 28 Juli 2021

Dosen Pembimbing 1

Dr. Ir. Sholeh Hadi Pramono, MS.
NIP. 195807281987011001

Dosen Pembimbing 2

Goegoes Dwi Nusantoro, ST., MT.
NIP. 197110132006041001

Mengetahui

Pt. Ketua Jurusan Teknik Elektro



Muhammad Ariz Muslim, S.T., M.T., Ph.D.

NIP. 19741203 200012 1 001

JUDUL SKRIPSI:

Analisis Tingkat Akurasi Implementasi Metode *Great Circle Distance* GPS Ublox Neo-6M Pada Sistem Pelacakan Dan Deteksi Sepeda Motor Dari Pencurian

Nama Mahasiswa : Agung Maulana Yusuf

NIM : 175060300111032

Program Studi : Teknik Elektro

Konsentrasi : Teknik Kontrol

KOMISI PEMBIMBING

Ketua : Dr. Ir. Sholeh Hadi Pramono M.S.



Anggota : Goegoes Dwi Nusantoro S.T., M.T.



TIM DOSEN PENGUJI:

Dosen Penguji I : Dr. Ir. Bambang Siswojo M.T.

 disetujui tanggal 28 Juli 2021

Dosen Penguji II : M. Aziz Muslim, S.T., M.T., Ph.D.

 disetujui tanggal 28 Juli 2021

Tanggal Ujian : 19 Juli 2021

SK Penguji : No. 1073 Tahun 2021

RINGKASAN

Agung Maulana Yusuf, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya, Mei 2021, *Analisis Tingkat Akurasi Implementasi Metode Great Circle Distance GPS Ublox Neo-6M Pada Sistem Pelacakan Dan Deteksi Sepeda Motor Dari Pencurian*, Dosen Pembimbing 1: Dr. Ir. Sholeh Hadi Pramono, MS., Dosen Pembimbing 2: Goegoes Dwi Nusantoro, ST., MT.

Global Positioning System (GPS) adalah sistem untuk menentukan letak di permukaan bumi dengan bantuan penyelarasan (*synchronization*) sinyal satelit. Sistem ini menggunakan 24 satelit yang mengirimkan sinyal gelombang mikro ke Bumi. Sinyal ini diterima oleh alat penerima di permukaan, dan digunakan untuk menentukan posisi, kecepatan, arah, dan waktu. Koordinat hasil keluaran GPS U-blox Neo 6M dan koordinat lokasi yang mendekati posisi aslinya dengan berdasar pada GPS *smartphone* akan dimasukkan ke persamaan *Great Circle Distance* untuk menghitung perbedaannya (meter) sehingga bisa ditarik kesimpulan ketidakakuratan dari GPS Ublox Neo-6M. Metode *Great Circle Distance* adalah suatu metode untuk mencari jarak terdekat antara dua titik di permukaan bumi. Dengan menggunakan metode *Great-Circle Distance*, error atau ketidakakuratan pembacaan koordinat oleh GPS Ublox Neo-6M bisa terlihat dalam satuan meter dan cara menganalisisnya pun memiliki beberapa syarat yaitu memiliki GPS/device lain yang tingkat akurasi lebih tinggi sehingga bisa dijadikan tolok ukur, data koordinat dari GPS referensi dan GPS Ublox Neo-6M dibandingkan sehingga terlihat perbedaannya. Rata-rata tingkat akurasi dari keluaran GPS Ublox Neo-6M yaitu akurat sampai 35,316 meter yang berarti keakuratan GPS Ublox Neo-6M masih rendah.

Kata Kunci: *Great Circle Distance*, GPS, Sistem Pelacakan dan Deteksi Sepeda Motor Dari Pencurian.

SUMMARY

Agung Maulana Yusuf, *Department of Electrical Engineering, Faculty of Engineering, University of Brawijaya, May 2021, Accuracy Analysis Level of the Implementation with the Great Circle Distance Method Ublox Neo-6M GPS on the Motorcycle Tracking and Detection System from Theft, Academic Advisor 1: Dr. Ir. Sholeh Hadi Pramono, MS., Academic Advisor 2: Goegoes Dwi Nusantoro, ST., MT.*

The Global Positioning System (GPS) is a system for determining the position on the earth's surface with the help of satellite signal synchronization. This system uses 24 satellites that send microwave signals to Earth. This signal is received by the receiver on the surface, and is used to determine position, speed, direction and time. The coordinates of the Ublox Neo-6M GPS output and location coordinates that are close to the original position based on the smartphone GPS will be entered into the Great Circle Distance equation to calculate the difference (meter) so that conclusions can be drawn from the Ublox Neo-6M GPS. The Great Circle Distance method is a method for finding the closest distance between two points on the earth's surface. By using the Great-Circle Distance method, errors or inaccuracies in reading coordinates by GPS Ublox Neo-6M can be seen in meters and the way to analyze it also has several requirements, namely having a GPS / other device with a higher accuracy level so that it can be used as a benchmark, coordinate data from the reference GPS and GPS Ublox Neo-6M compared so that the difference is visible. The average accuracy rate of the Ublox 6-M GPS output is accurate up to 35,316 meters.

Keywords: *Great Circle Distance, GPS, Motorcycle Tracking and Detection System from Theft.*

LEMBAR ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya dan berdasarkan hasil penelusuran berbagai karya ilmiah, gagasan dan masalah ilmiah yang diteliti dan diulas di dalam Naskah Skripsi ini adalah asli dari pemikiran saya. tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu Perguruan Tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata di dalam naskah Skripsi ini dapat dibuktikan terdapat unsur-unsur jiplakan, saya bersedia Skripsi dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, pasal 25 ayat 2 dan pasal 70).

Malang, 16 Juni 2021

Mahasiswa,

Agung Maulana Yusuf
NIM.175060300111032



9. Mbak Ega Putri Anatasya Br. Bangun selaku pacar yang pengertian, sabar dan pengasih dalam rentang waktu penulisan dokumen skripsi ini.
10. Teman-teman ELCO 2017 atas segala bantuan dan kebersamaan yang telah diberikan selama masa studi.
11. Teman-teman kelompok CDP (Agung, Ghozy, dan Okki) atas bimbingan, pengalaman, perhatian, canda, tawa, sedih dan semangat yang telah kami lalui bersama dalam mengerjakan tugas akhir CDP.
12. Teman-teman grup sosial media Line Profesional Skripsi Player atas segala bantuan, curahan hati dan kebersamaan yang telah diberikan selama masa SMA sampai sekarang.
13. Teman-teman grup sosial media Line biji atas segala pendewasaan, wawasan, berbagi pengalaman dan kebersamaan yang telah diberikan selama masa SMA sampai sekarang.
14. Seluruh teman-teman serta semua pihak yang tidak mungkin bagi penulis untuk mencantumkan satu-persatu, terimakasih banyak atas bantuan dan dukungannya.

Dalam penyusunan ini, penulis menyadari bahwa skripsi ini belum sempurna karena keterbatasan ilmu yang dimiliki serta terdapat kendala-kendala lain yang terjadi selama pengerjaan skripsi ini. Oleh karena itu penulis berharap kritik dan saran untuk penyempurnaan tulisan dimasa mendatang. Semoga tulisan ini dapat bermanfaat dan dapat digunakan untuk pengembangan lebih lanjut.

Malang, Juni 2021

Penulis

DAFTAR ISI

RINGKASAN	iii
SUMMARY	iv
LEMBAR ORISINALITAS SKRIPSI.....	v
KATA PENGANTAR	vi
DAFTAR ISI.....	viii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Penelitian	2
1.4 Tujuan	3
1.5 Manfaat	3
1.6 Sistematika Penulisan Skripsi	3
BAB II TINJAUAN PUSTAKA	5
2.1 Mikrokontroler	5
2.1.1 Node MCU ESP8266	5
2.2 <i>Global Positioning System (GPS)</i>	6
2.2.1 Modul GPS Ublox Neo 6M.....	9
2.2.2 Koordinat GPS (<i>Latitude & Longitude</i>).....	9
2.2.3 Konsep Triangulasi	10
2.2.4 Geometri Satelit GPS	11
2.2.5 GPS Clock Receiver.....	11
2.3 Wifi Modul ESP 8266	11
2.4 <i>Google Maps</i>	12
2.5 Perhitungan Triangulasi	13
2.6 Analisis	13
2.7 Akurasi dan Presisi	14
2.8 <i>Haversine Formula</i>	14
BAB III METODE PENELITIAN.....	17
3.1 Skenario Pengujian.....	17
3.1.1 Rancangan Proses Kerja Aplikasi Android.....	17
3.1.2 Pengujian Sistem GPS	18



3.1.3	Sistem Pelacakan Operasional Menggunakan <i>Great Circle Distance</i>	20
3.2	Metode Pengumpulan Data	21
3.2.1	Observasi	21
3.2.2	Penentuan Variabel Tetap	21
3.2.3	Pengujian Variabel Bebas	21
3.3	Dokumentasi	22
BAB IV DESAIN DAN RANCANGAN SISTEM.....		23
4.1	Pembuatan Perangkat Keras	23
4.1.1	Skema Pembuatan Perangkat Keras	23
4.1.2	Penentuan Modul Elektrik Yang Digunakan	24
4.2	Spesifikasi Sistem	26
4.3	Perancangan Blok Diagram Sistem	28
4.3.1	Rancangan Perangkat Lunak	28
4.4	Perancangan Aplikasi Android	29
4.4.1	Rancangan Antarmuka Aplikasi Android	29
4.4.2	Rancangan Proses Kerja Aplikasi Android	31
BAB V HASIL PENELITIAN DAN PEMBAHASAN		33
5.1	Hasil Penelitian	33
5.1.1	Data Koordinat Pada Ruang Terbuka Padat Gedung	33
5.1.2	Data Koordinat Pada Ruang Terbuka Minim Gedung (Lapangan)	35
5.1.3	Data Koordinat Pada Ruang Terbuka Beratap/Kanopi	36
BAB VI KESIMPULAN DAN SARAN.....		39
6.1	Kesimpulan	39
6.2	Saran	40
DAFTAR PUSTAKA.....		41
LAMPIRAN		43



DAFTAR TABEL

Tabel 5.1 Data koordinat Ruang Terbuka Padat Gedung 33
Tabel 5.2 Data Koordinat Ruang Terbuka Minim Gedung (Lapangan)..... 35
Tabel 5.3 Data Koordinat Pada Ruang Terbuka Beratap/Kanopi 36



DAFTAR GAMBAR

Gambar 2.1 NodeMcu ESP8266.....6

Gambar 2.2 How GPS Works7

Gambar 2.3 Konsep Triangulasi.....7

Gambar 2.4 Modul Fisik GPS Ublox Neo 6M.....9

Gambar 2.5 Latitude & Longitude.....10

Gambar 2.6 Triangulasi Koordinat Kapal.....10

Gambar 2.7 Modul Wifi ESP8266.....12

Gambar 2.8 Logo Google Maps.....12

Gambar 2.9 Ilustrasi GPS Receiver dan Satelit.....13

Gambar 2.10 Diagram Ilustrasi Jarak Great-Circle.....15

Gambar 3.1 Blok diagram proses kerja aplikasi.....17

Gambar 3.2 Flowchart Pengambilan Data.....19

Gambar 3.3 Arsitektur Sistem.....20

Gambar 4.1 Diagram blok umum dari sistem yang dirancang.....23

Gambar 4.2 Node Mcu ESP8266.....24

Gambar 4.3 Sensor Ultrasonik.....25

Gambar 4.4 Buzzer.....25

Gambar 4.5 Driver Motor L298N.....25

Gambar 4.6 *Smartphone*.....26

Gambar 4.7 Diagram Blok Sistem.....28

Gambar 4.8 Diagram Alir Perangkat Lunak.....29

Gambar 4.9 Tampilan Link Antarmuka di Browser *Smartphone*.....30

Gambar 4.10 Halaman aplikasi disaat dibuka dengan laptop.....31

Gambar 4.11 Blok diagram proses kerja aplikasi.....31

Gambar 5.1 Pengujian Ruang Terbuka Padat Gedung Tinggi.....34

Gambar 5.2 Pengujian Ruang Terbuka Minim Gedung (Lapangan).....36

Gambar 5.3 Pengujian Ruang Terbuka Beratap/Kanopi.....37

BAB I PENDAHULUAN

1.1 Latar Belakang

Tahun 2018, tercatat total jumlah kendaraan bermotor sebesar 136.542.034 dan ditahun tersebut jumlah kasus pencurian sepeda motor mencapai 27.731. Jumlah tersebut sudah menurun apabila dibandingkan dengan tahun 2014, ketika jumlah mobil penumpang dan sepeda motor masih berjumlah 105.575.278, dan kasus curanmor sebanyak 42.165. Akan tetapi jumlah kasus pencurian sepeda motor cenderung masih tinggi jika dibandingkan dengan kasus pencurian sepeda motor di negara lain. Di kota Malang sendiri, di tahun 2017 terdapat 892 pelaporan kasus pencurian sepeda motor, dan hanya terselesaikan sebanyak 309 kasus, sedangkan di tahun 2018 terjadi penurunan kasus pencurian sepeda motor, terdapat 264 kasus pencurian sepeda motor, dan hanya dapat diselesaikan sebanyak 160 kasus. Walaupun terjadi penurunan dari tahun ke tahun, tetap saja kasus pencurian sepeda motor ini masih banyak terjadi di sekitar kita.

Pada permasalahan di atas dengan ini diimplementasikan suatu sistem yang membantu pengendara sepeda motor agar dapat mengetahui lokasi motor ketika sedang meninggalkan/memarkirkan motornya dengan perantara ponsel sebagai monitor. Sistem ini menggunakan *Global Positioning Sistem* (GPS) untuk mengetahui posisi sepeda motor ketika berada di posisi awal maupun posisi baru yang nantinya akan terhubung ke *smartphone* pengguna.

Sistem ini dirancang menggunakan sebuah mikrokontroler Node MCU ESP8266 sebagai pengolahan data dari GPS serta mengirimkan data tersebut dengan menggunakan modul Wifi ke HTTP *server* kemudian diakses oleh pengguna melalui *smartphone*. Sistem ini nantinya akan disimpan di motor agar dapat mendeteksi dan mengirimkan notifikasi kepada pengguna ketika motor tersebut menunjukkan indikasi pencurian.

Penelitian ini berdasar pada *Capstone Design Project* berjudul *Sistem Pelacakan dan Deteksi Sepeda Motor Dari Pencurian, Great-Circle Distance* dibutuhkan dalam penelitian ini untuk menghitung tingkat akurasi GPS Ublox Neo-6M dengan menggunakan perantara dan basis data yang diunggah ke *server* internet.

1.2 Rumusan Masalah

Sistem pelacakan digunakan di masa modern ini karena adanya kebutuhan pada tingkat keamanan yang kurang mumpuni atau tidak mencukupi, maka dibutuhkan suatu hal untuk meningkatkan keamanan sepeda motor.

Berdasarkan latar belakang yang telah dipaparkan diatas maka dapat dirumuskan permasalahan sebagai berikut:

1. Bagaimana menganalisis *error* koordinat “Pada Sistem Pelacakan Dan Deteksi Sepeda Motor Dari Pencurian” dengan mikrokontroler Node MCU ESP8266 dan menggunakan GPS Ublox Neo-6M?
2. Apakah koordinat keluaran GPS Ublox Neo-6M mendekati lokasi *real* (mendekati) yang diinginkan dengan patokan *Google Maps* dari *smartphones*?
3. Bagaimana tingkat akurasi GPS Ublox Neo-6M menggunakan metode *Great Circle Distance*?

1.3 Batasan Penelitian

Adapun batasan-batasan masalah pada penelitian ini adalah sebagai berikut:

1. Kendaraan bermotor yang digunakan dalam penelitian ini adalah sepeda motor merk Yamaha Aerox
2. Mikrokontroler yang digunakan adalah Node Mcu ESP8266 dan sensor ultrasonic digunakan sebagai data masukan
3. Modul GPS yang digunakan adalah Ublox Neo-6M
4. Indikasi gejala pencurian motor dibatasi pada adanya gerakan dan jarak manusia di depan kunci sepeda motor
5. Arduino diprogram dengan Arduino IDE.
6. Pengujian dilakukan dengan melihat keberhasilan alat mendeteksi gejala pencurian motor dan pengiriman notifikasi serta posisi motor ke ponsel pengguna.
7. Alat tidak bekerja jika tidak dinyalakan terlebih dahulu dari aplikasi pada ponsel pengguna.
8. Alat diletakkan di dalam bagasi motor.

1.4 Tujuan

Tujuan penelitian ini adalah menentukan tingkat akurasi sistem deteksi dan pelacakan sepeda motor dari pencurian dengan menggunakan GPS Ublox Neo-6M.

1.5 Manfaat

Manfaat dari perancangan sistem ini adalah sebagai berikut:

1. Mengurangi angka pencurian motor.
2. Mengetahui tingkat akurasi GPS.
3. Melakukan analisis perbandingan akurasi GPS Ublox Neo-6M dan GPS Google Maps terhadap pengujian sistem.
4. Mengetahui perhitungan metode *Great Circle Distance* antara dua titik koordinat.
5. Memberikan spesifikasi khusus mengenai tingkat akurasi GPS Ublox Neo-6M.

1.6 Sistematika Penulisan Skripsi

Dalam proses penyelesaian masalah akan dilakukan beberapa metode, antara lain:

1. Studi literatur

Pada tahap ini dilakukan pencarian dan pengumpulan literatur-literatur dan kajian-kajian yang berkaitan dengan masalah-masalah yang ada pada tugas akhir ini, baik berupa artikel, buku referensi, jurnal, dan sumber-sumber lain yang terpercaya. Hal-hal yang dipelajari meliputi:

- Konsep GPS
- Metode *Great Circle Distance*

2. Simulasi dan Perancangan Sistem

Tahap ini merupakan tahap lanjutan setelah mempelajari literatur, tahap selanjutnya adalah membuat rancangan sistem dan penentuan spesifikasi perangkat yang digunakan serta melakukan analisis *error* keluaran GPS dengan cara menganalisis pengujian akurasi dari GPS dan pemberian notifikasi kepada pengguna.

3. Pengujian

Pengujian dilakukan untuk memastikan bahwa sistem berjalan dengan baik dan untuk mencari kekurangan dari sistem agar dapat disempurnakan lagi.

4. Penyusunan laporan dan Pengambilan Kesimpulan

Bertujuan untuk melaporkan hasil dari setiap pelaksanaan dari penelitian dan mengambil kesimpulan dari setiap analisis yang telah dilakukan.

BAB II TINJAUAN PUSTAKA

Bab ini memaparkan literatur, jurnal, dan teori dasar yang digunakan sebagai acuan dan pedoman dalam proses penyelesaian masalah yang ada pada pelaksanaan penelitian ini.

2.1 Mikrokontroler

Mikrokontroler adalah sebuah sistem komputer yang seluruh atau sebagian besar elemennya dikemas dalam satu chip IC, sehingga sering disebut single chip microcomputer.

Mikrokontroler merupakan sistem komputer yang mempunyai salah satu atau beberapa tugas yang sangat spesifik. (Chamim, (2012)). Dalam tugas akhir ini akan digunakan mikrokontroler Node MCU ESP8266.

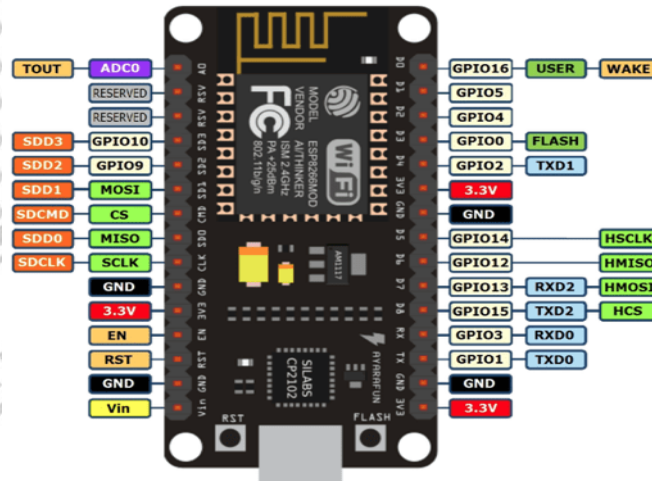
2.1.1 Node MCU ESP8266

NodeMCU merupakan sebuah open source platform IoT dan pengembangan kit yang menggunakan bahasa pemrograman Lua untuk membantu dalam membuat prototype produk IoT atau bisa dengan memakai sketch dengan adruino IDE. Pengembangan kit ini didasarkan pada modul ESP8266, yang mengintegrasikan GPIO, PWM (Pulse Width Modulation), IIC, 1-Wire dan ADC (*Analog to Digital Converter*) semua dalam satu board. Node MCU berukuran panjang 4.83cm, lebar 2.54cm, dan berat 7 gram. Board ini sudah dilengkapi dengan fitur WiFi dan Firmwarena yang bersifat opensource.

Spesifikasi yang dimiliki oleh NodeMCU sebagai berikut :

1. Board ini berbasis ESP8266 serial WiFi SoC (Single on Chip) dengan onboard USB to TTL. Wireless yang digunakan adalah IEEE 802.11b/g/n.
2. Tantalum capacitor 100 micro farad dan 10 micro farad.
3. 3.3v LDO regulator.
4. Blue led sebagai indikator.
5. Cp2102 usb to UART bridge.
6. Tombol reset, port usb, dan tombol flash.
7. Terdapat 9 GPIO yang di dalamnya ada 3 pin PWM, 1 x ADC Channel, dan pin RX TX
8. 3 pin ground.
9. S3 dan S2 sebagai pin GPIO
10. S1 MOSI (*Master Output Slave Input*) yaitu jalur data dari master dan masuk ke dalam slave, sc cmd/sc.

11. S0 MISO (*Master Input Slave Input*) yaitu jalur data keluar dari slave dan masuk ke dalam master.
12. SK yang merupakan SCLK dari master ke slave yang berfungsi sebagai clock.
13. Pin Vin sebagai masukan tegangan.
14. Built in 32-bit MCU

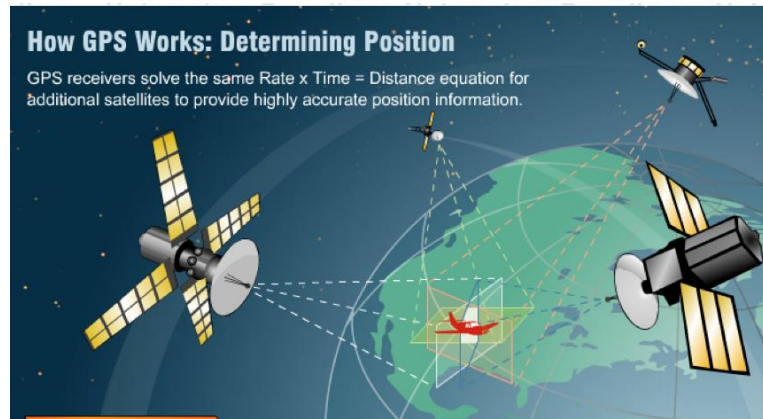


Gambar 2.1 NodeMcu ESP8266

(Sumber: <https://www.nyebarilmu.com/apa-itu-module-nodemcu-esp8266/>)

2.2 Global Positioning System (GPS)

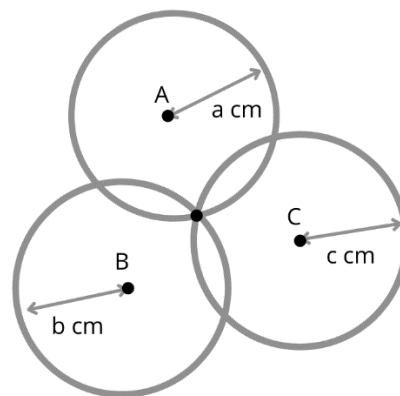
GPS adalah singkatan dari *Global Positioning System*. GPS adalah sistem navigasi yang berbasis satelit yang mengorbit bumi. GPS *receiver* mengambil informasi itu dengan menggunakan perhitungan “*triangulation*” menghitung lokasi *user* dengan tepat. GPS *receiver* membandingkan waktu sinyal di kirim dengan waktu sinyal tersebut di terima. Dari informasi itu didapat diketahui berapa jarak satelit. Dengan perhitungan jarak, GPS *receiver* dapat melakukan perhitungan dan menentukan posisi *user* dan menampilkan dalam *maps*. Sebuah GPS *receiver* harus mengunci sinyal minimal tiga satelit untuk menghitung posisi 2D (*latitude* dan *longitude*) dari pergerakan *user* (*tracking*). Jika GPS *receiver* dapat menerima empat atau lebih satelit, maka hal itu dapat menghitung posisi 3D (*latitude*, *longitude* dan *altitude*). Jika sudah dapat menentukan posisi *user*, selanjutnya GPS dapat menghitung informasi lain, seperti kecepatan, arah yang dituju, jalur, tujuan perjalanan, jarak tujuan, matahari terbit dan matahari terbenam dan masih banyak lagi. Ilustrasi ditunjukkan oleh Gambar 2.2.



Gambar 2.2 *How GPS Works*

(Sumber: <http://avionicswest.com/Articles/howGPSworks.html>)

Jumlah minimal yang dibutuhkan untuk dapat menentukan lokasi (koordinat) obyek yang diamati adalah 3 satelit. Hal ini berhubungan dengan konsep Triangulasi. Triangulasi dapat dianalogikan sebagai berikut: Suatu titik A berada pada jarak a cm dari pengamat X. Dari informasi ini dapat diketahui bahwa X dapat terletak di mana saja sepanjang keliling lingkaran dengan radius a cm (ditunjukkan oleh **Gambar 2.3** kiri). Titik B diketahui berada pada jarak b cm dari X (disajikan pada **Gambar 2.3** tengah). Dari data kedua ini dapat ditentukan dua kemungkinan posisi X (titik merah), yaitu di kedua titik perpotongan kedua lingkaran. Kemudian titik C diketahui berada pada jarak c cm (disajikan pada **Gambar 2.3** kanan) dari posisi X. Dengan data terakhir ini bisa dengan tepat dipastikan letak X.



Gambar 2.3 Konsep Triangulasi

Faktor-faktor yang dapat memengaruhi sinyal dan akurasi GPS meliputi hal-hal berikut:

1. Penundaan ionosfer dan troposfer

Sinyal satelit lambat saat melewati atmosfer. Sistem GPS menggunakan model built-in untuk sebagian besar untuk jenis kesalahan ini.

2. *Multipath Signal*

Sinyal GPS dapat memantulkan objek seperti gedung tinggi atau permukaan batu besar sebelum mencapai penerima, yang akan meningkatkan waktu perjalanan sinyal dan menyebabkan kesalahan.

3. Kesalahan *clock receiver*

Jam built-in receiver mungkin memiliki sedikit kesalahan pengaturan waktu karena kurang akurat dibandingkan jam atom pada satelit GPS.

4. Kesalahan orbital

Lokasi yang dilaporkan satelit mungkin tidak akurat.

5. Jumlah satelit yang terlihat

Semakin banyak satelit, penerima GPS dapat "melihat", semakin baik akurasi.

Ketika sinyal diblokir, Anda mungkin mendapatkan kesalahan posisi atau mungkin tidak ada posisi membaca sama sekali. Unit GPS biasanya tidak akan bekerja di bawah air atau di bawah tanah, tetapi penerima sensitivitas tinggi baru dapat melacak beberapa sinyal ketika berada di dalam gedung atau di bawah penutup pohon.

6. Geometri satelit / bayangan

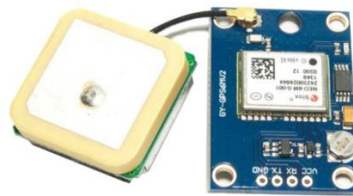
Sinyal satelit lebih efektif ketika satelit berada pada sudut lebar relatif satu sama lain, bukan dalam garis atau pengelompokan yang ketat.

2.2.1 Modul GPS Ublox Neo 6M

Modul GPS ublox Neo 6M merupakan modul GPS yang cocok digunakan untuk mobilitas tinggi. Penggunaan modul GPS yang digunakan mempunyai kelebihan dibandingkan kamera. Kamera tidak mungkin ditempatkan ditempat yang tersembunyi sehingga kamera akan ditempatkan yang bisa menjangkau cakupan luas. Hal ini dapat menjadi celah untuk oknum yang mengetahui cara kerja sistem dan mengambil keuntungan darinya terutama malam hari. Adapun modul GPS yang digunakan adalah modul GPS Module Neo Ublox Neo 6M berikut fitur utama Ublox Neo 6M:

- Indoor GPS: -162 dBm sensitivity tracking
- Voltase: 3,3V – 5V
- Temperatur operasi: - 40-85 C

Modul GPS berfungsi untuk mendapatkan lokasi sepeda motor berupa koordinat *Longitude* dan *Latitude*. Bentuk fisik modul GPS Ublox Neo-6M ditunjukkan oleh **Gambar 2.4**.



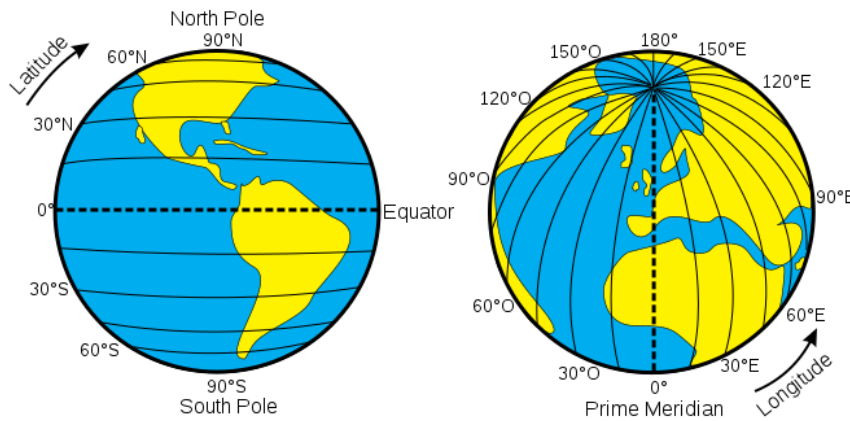
Gambar 2.4 Modul Fisik GPS Ublox Neo 6M

(Sumber: <https://www.cronyos.com/cara-menggunakan-modul-GPS-ublox-neo-6m-dengan-arduino/>)

2.2.2 Koordinat GPS (*Latitude & Longitude*)

Diawali dengan 2 koordinat letak bumi dari angka depan *Latitude* (ketinggian) dan angka belakang *Longitude* (panjang). *Latitude*: garis lintang mengarah dari khatulistiwa (0) ke kutub selatan, atau khatulistiwa ke kutub utara (sudut 0-90 dan 0 -90) – *Longitude*: garis bujur adalah garis horizontal seperti dari khatulistiwa. Sudut 0 (Greenwich) horizon Hawaii adalah 0-180, sedangkan kebalikannya dari 0 ke -180 *Latitude* adalah garis yang horizontal / mendatar. Titik 0 adalah sudut ekuator, tanda + menunjukkan arah ke atas menuju kutub utara, sedangkan tanda minus di koordinat *Latitude* menuju ke kutub selatan. Titik yang

dipakai dari 0 ke 90 derajat horizon kutub utara, dan 0 ke -90 derajat ke kutub selatan. *Longitude* adalah garis lintang. Angka dari sudut bundar bumi horizontal. Titik diawali dari 0 ke 180 derajat, dan 0 ke-180 ke arah sebaliknya. Ilustrasi bisa dilihat pada **Gambar 2.5** di bawah.

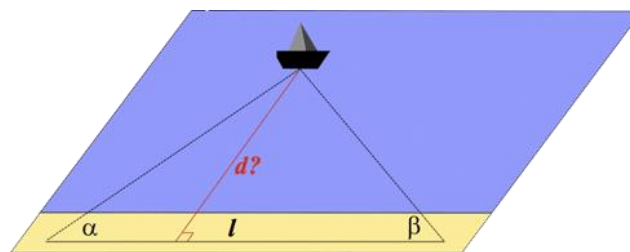


Gambar 2.5 Latitude & Longitude

(Sumber: <http://www.habibullahurl.com/2018/08/pengertian-garis-lintang-dan-garis-bujur.html>)

2.2.3 Konsep Triangulasi

Triangulasi adalah proses mencari koordinat dari sebuah titik dengan cara menghitung panjang sisi segitiga yang berhadapan dengan titik tersebut, dan ukuran kedua sudut antara garis tersebut ke titik yang dicari sudah diketahui. Proses ini bisa dijalankan dengan syarat sudah diketahui dengan pasti berapa besar kedua sudut yang terbentuk antara garis acuan dengan titik yang ingin dicari koordinatnya. **Gambar 2.6** menunjukkan koordinat kapal (obyek) yang telah secara triangulasi.



Gambar 2.6 Triangulasi Koordinat Kapal

(Sumber: <https://ryosaeba.wordpress.com/2007/11/27/triangulasi-trilaterasi-dan-multilaterasi/>)

Penentuan jarak kapal dari garis pantai dengan cara menetapkan dahulu dua titik acuan A dan B yang berada di garis pantai. Jarak dua titik diukur dengan teliti dan misalkan kita dapatkan l . Misalkan sudut yang terbentuk di titik A adalah α dan sudut yang dibentuk di titik B adalah β . Misalkan jarak kapal ke garis pantai yang ingin kita tentukan adalah d .

2.2.4 Geometri Satelit GPS

Dimisalkan melakukan pengukuran jarak kita dari satelit dan diperoleh hasil 11.000 mil. Perlu diketahui dengan jarak 11000 mil dari satelit yang sangat jauh, semua akan terlihat cakupan yang luas dan seolah-olah bumi yang dilihat dari satelit terpusat menjadi satu titik.

Untuk mengukur jarak dari sesuatu yang bergerak yaitu dengan menghitung berapa lama waktu yang diperlukan untuk mengirim sinyal dari satelit GPS ke tempat *receiver* GPS itu berada. Jadi dimanapun *receiver* GPS itu, di pesawat udara, kapal laut atau kendaraan di darat tidak menjadi masalah.

Perlu diingat pula bahwa mengukur perjalanan waktu sinyal radio GPS itu merupakan suatu hal yang penting, sehingga settingan waktu pada *receiver* GPS harus benar-benar tepat. Seandainya setting waktu *receiver* GPS berhenti ataupun berbeda seperseribu detik pun itu berpengaruh. Begitupula jam atau setting waktu GPS Satelit, jamnya mati atau telat seperseribu detik, maka dengan konversi kecepatan cahaya, kesalahan sinyal yang ditransmisikan *error*nya sekitar 200 mil. Bagian satelit GPS telah menggunakan jam yang sangat akurat, karena satelit GPS dilengkapi dengan jam atom di dalamnya.

2.2.5 GPS Clock Receiver

Satelit GPS dengan *receiver* GPS harus benar-benar sinkron terhadap masing-masing *code pseudo random*, untuk membuat sistem GPS berjalan benar.

Untuk meningkatkan presisi dapat dilakukan pengukuran atau pengamatan satelit GPS lebih banyak. Menggunakan pengukuran 3 satelit GPS, maka dapat menentukan posisi suatu titik dalam suatu bentuk ruang 3 dimensi, begitu juga dengan mengukur menggunakan 4 satelit GPS dapat mendapatkan hasil lebih baik.

2.3 Wifi Modul ESP 8266

ESP8266 adalah chip yang menawarkan solusi jaringan Wifi lengkap dan independen. Penggunaan modul ini sebagai media komunikasi melalui jaringan Wifi untuk mengirimkan data ke ponsel. Penggunaan modul Wifi mempunyai kelebihan dibandingkan dengan modul GSM, modul GSM/GPRS juga merupakan platform yang biayanya sangat mahal karena dalam pengambilan data yang diperlukan pada sistem ini sangatlah banyak agar dapat

menghasilkan data yang lebih akurat sehingga memerlukan pulsa yang lebih besar. Sedangkan modul wifi merupakan platform yang sangat murah sehingga lebih efektif untuk pengambilan data yang banyak. Modul Wifi berfungsi pengiriman data melalui internet ke server.



Gambar 2.7 Modul Wifi ESP8266

(Sumber: <https://beetrona.com/pengertian-esp8266-modul-wifi-lengkap/>)

2.4 Google Maps

Google Maps adalah layanan pemetaan web yang dikembangkan oleh Google. Layanan ini memberikan citra satelit, peta jalan, panorama 360°, kondisi lalu lintas, dan perencanaan rute untuk bepergian dengan berjalan kaki, mobil, sepeda (versi beta), atau angkutan umum. *Google Maps* memiliki fitur search sehingga pengguna tidak perlu mencari secara manual lokasi yang diinginkan. Serta memiliki fitur *Google Street View* dapat digunakan untuk melihat langsung visual daerah yang diinginkan melalui jalan yang telah direkam oleh Google.



Google Maps

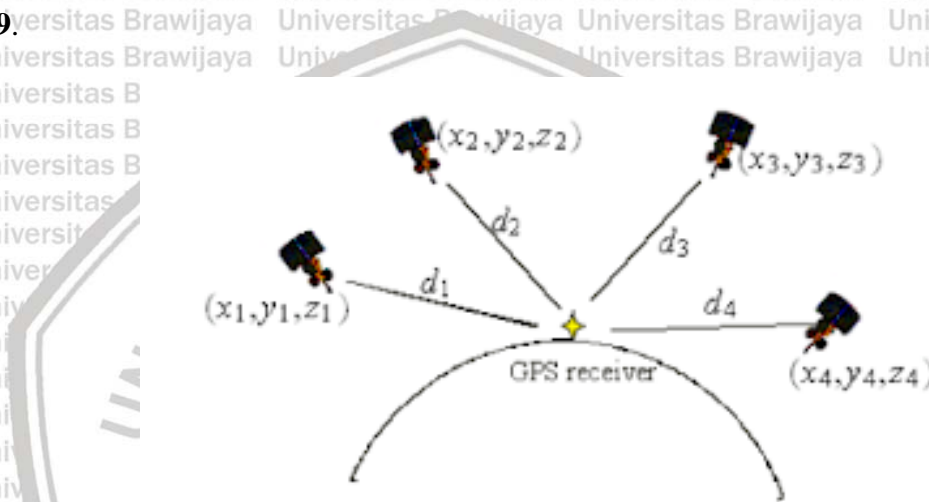
Gambar 2.8 Logo Google Maps

(sumber: <https://terralogiq.com/cara-menggunakan-google-maps/>)

2.5 Perhitungan Triangulasi

Perhatikan **Gambar 2.9** alat GPS menerima sinyal dari 3 satelit. Sinyal tersebut memberi informasi tentang koordinat 3 satelit berupa sudut lintang dan bujur. Dengan rumus yang telah dipasang dalam alat GPS maka koordinat 3 satelit dapat ditentukan. Misalkan koordinat 3 satelit tersebut adalah (x_1, y_1, z_1) , (x_2, y_2, z_2) , dan (x_3, y_3, z_3) . Berdasarkan waktu yang ada di sinyal satelit maka alat GPS dapat menghitung jarak masing-masing satelit. Misalkan jarak masing-masing satelit yang dihitung adalah d_1, d_2 , dan d_3 . Yang akan kita tentukan adalah koordinat alat GPS yang berada di bumi. Misalkan GPS tersebut berada pada koordinat (x, y, z) dan koordinat tersebut yang akan kita tentukan seperti pada **Gambar**

2.9.



Gambar 2.9 Ilustrasi GPS Receiver dan Satelit

Persamaan pythagoras:

$$(X - X_1)^2 + (Y - Y_1)^2 = D_1^2$$

$$(X - X_2)^2 + (Y - Y_2)^2 = D_2^2$$

$$(X - X_3)^2 + (Y - Y_3)^2 = D_3^2$$

2.6 Analisis

Analisis yang dimaksudkan dalam penelitian ini adalah kajian tentang data koordinat yang berkaitan dengan lokasi obyek yang terdeteksi. Analisis yang dipakai dalam penelitian ini adalah analisis komparatif.

Menurut Nazir (2005: 58) penelitian komparatif adalah sejenis penelitian deskriptif yang ingin mencari jawaban secara mendasar tentang sebab-akibat, dengan menganalisis faktor-faktor penyebab terjadinya ataupun munculnya suatu fenomena tertentu.

Penelitian yang bersifat membandingkan. Penelitian ini dilakukan untuk membandingkan persamaan dan perbedaan dua atau lebih fakta-fakta dan sifat-sifat objek yang di teliti berdasarkan kerangka pemikiran tertentu. Pada penelitian ini variabelnya masih mandiri tetapi untuk sampel yang lebih dari satu atau dalam waktu yang berbeda. Analisis komparatif sendiri memiliki tujuan umum yaitu:

1. Membandingkan persamaan dan perbedaan dua atau lebih fakta-fakta dan sifat-sifat objek yang di teliti berdasarkan kerangka pemikiran tertentu.
2. Membuat generalisasi tingkat perbandingan berdasarkan cara pandang atau kerangka berpikir tentu.
3. Menentukan mana yang lebih baik atau mana yang sebaiknya dipilih.
4. Menyelidiki kemungkinan hubungan sebab-akibat dengan cara berdasar atas pengamatan terhadap akibat yang ada dan mencari kembali faktor yang mungkin menjadi penyebab melalui data tertentu.

Analisis komparatif memiliki tujuan khusus dalam penelitian ini yaitu membandingkan keluaran GPS *smartphones* MI A2 dengan GPS Ublox Neo-6M di tempat dan waktu yang sama.

2.7 Akurasi dan Presisi

Akurasi dalam pengukuran merupakan tingkat kedekatan pengukuran kuantitas terhadap nilai yang sebenarnya, sedangkan kepresisian dari suatu sistem pengukuran diartikan sejauh mana pengulangan pengukuran dalam kondisi yang tidak berubah mendapatkan hasil yang sama. Algoritme PPP yang telah terbukti di industri u-blox memberikan tingkat akurasi posisi yang sangat tinggi dalam aplikasi bergerak dengan keadaan statis dan lambat, dan menjadikan NEO-6M solusi ideal untuk berbagai aplikasi presisi tinggi seperti survei, pemetaan, kelautan, pertanian atau kegiatan rekreasi. Koreksi ionosfer seperti yang diterima dari satelit geostasioner SBAS12 lokal (WAAS, EGNOS, MSAS) atau dari GPS memungkinkan akurasi posisi tertinggi dengan algoritma PPP. Peningkatan maksimum dari akurasi pemosisian dicapai dengan PPP+SBAS dan hanya dapat diharapkan di lingkungan tanpa halangan pemandangan langit selama periode dalam urutan menit.

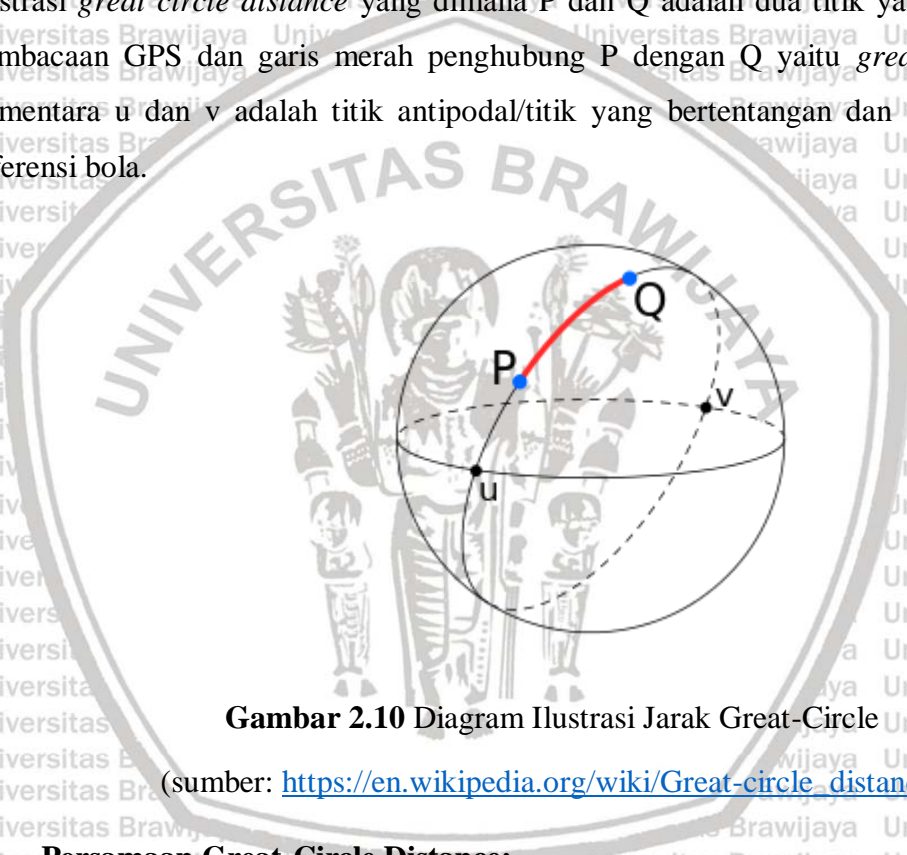
2.8 Haversine Formula

Rumus Haversine / *Haversine Formula* merupakan persamaan yang umumnya digunakan dalam navigasi, karena dapat memberikan radius dengan “*Great Circle*” antara dua titik pada suatu bentuk bola berdasarkan data dari garis bujur dan garis lintang.

Persamaan ini merupakan pengembangan dari rumus yang lebih umum dalam trigonometri bidang bola yaitu hukum haversines yang mana menghubungkan sisi dan sudut dari segitiga dalam bidang bola. Persamaan haversine ditunjukkan dengan persamaan:

$$2r(\sqrt{\sin^2\left(\frac{\phi_2-\phi_1}{2}\right) + \cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\lambda_2-\lambda_1}{2}\right)}). \dots\dots\dots(1)$$

Great Circle Distance adalah jarak lingkaran besar, jarak ortodromik, atau jarak bola adalah jarak terpendek antara dua titik pada permukaan bola, diukur di sepanjang permukaan bola (berlawanan dengan garis lurus melalui interior bola). Bisa dilihat di **Gambar 2.10** ilustrasi *great circle distance* yang dimana P dan Q adalah dua titik yang diketahui/hasil pembacaan GPS dan garis merah penghubung P dengan Q yaitu *great circle distance*. Sementara u dan v adalah titik antipodal/titik yang bertentangan dan berfungsi sebagai referensi bola.



Gambar 2.10 Diagram Ilustrasi Jarak Great-Circle

(sumber: https://en.wikipedia.org/wiki/Great-circle_distance)

Persamaan Great-Circle Distance:

R = jari-jari dari bumi (kurang lebih = 6371 km)

$$\Delta lat = lat_2 - lat_1 \dots\dots\dots(2)$$

$$\Delta long = long_2 - long_1 \dots\dots\dots(3)$$

Dari persamaan (1) dan (2). Didapatkan perbedaan antara *longitude* dan *latitude* dari titik keluaran alat dan titik referensi/patokan.

$$a = \sin^2\left(\frac{\Delta lat}{2}\right) + \cos(lat_1) \cdot \cos(lat_2) \cdot \sin^2\left(\frac{\Delta long}{2}\right) \dots \dots \dots (4)$$

$$c = 2 \cdot \text{atan2}(\sqrt{a} \cdot \sqrt{1-a}) \dots \dots \dots (5)$$

$$d = R \cdot c \dots \dots \dots (5)$$

Sudut harus dalam radian untuk menggunakan rumus tersebut.

Keterangan:

lat_1 = garis lintang di titik 1

lat_2 = garis lintang di titik 2

Δlat = selisih garis lintang 2 & 1

$long_1$ = garis bujur di titik 1

$long_2$ = garis bujur di titik 2

$\Delta long$ = selisih garis bujur 2 & 1

a = rumus haversine

c = rumus *great-circle*

d = *great-circle distance*

Pengimplementasian rumus *Great-Circle Distance* pada Microsoft Excel:

$$= \text{ACOS}(\text{COS}(\text{RADIANS}(90 - \text{Latitude}_1)) * \text{COS}(\text{RADIANS}(90 - \text{Latitude}_2)) + \text{SIN}(\text{RADIANS}(90 - \text{Latitude}_1)) * \text{SIN}(\text{RADIANS}(90 - \text{Latitude}_2)) * \text{COS}(\text{RADIANS}(\text{Longitude}_1 - \text{Longitude}_2))) * 6371 * 1000$$

Catatan:

Longitude dan *Latitude* harus dalam bentuk sudut derajat.

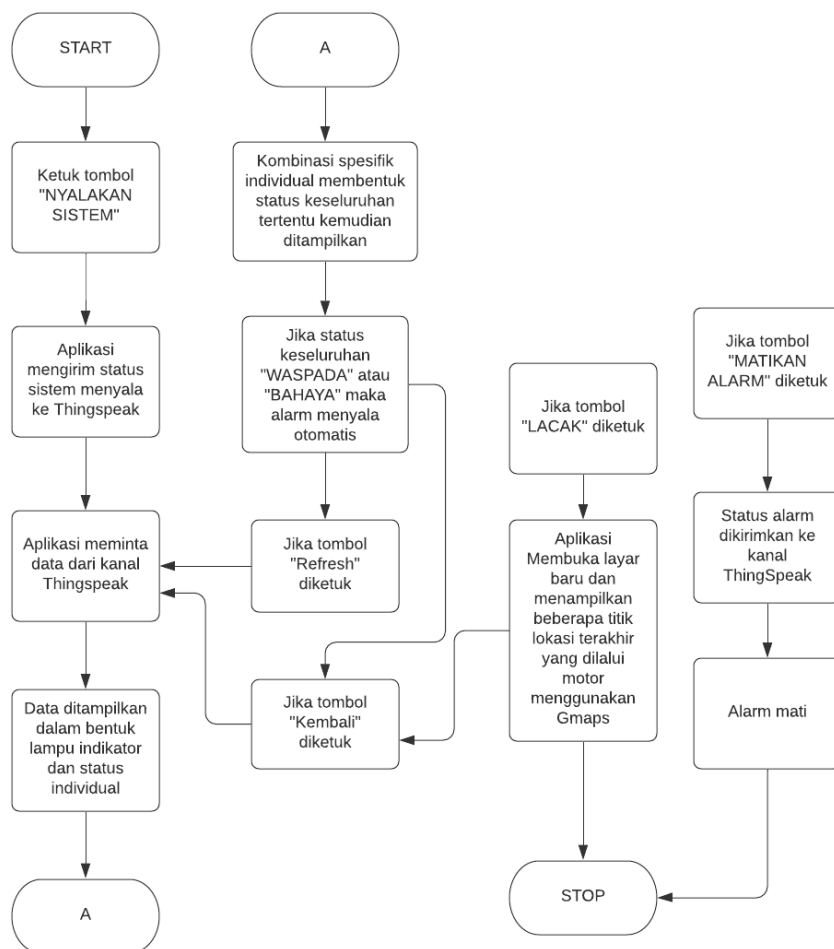
BAB III METODE PENELITIAN

Penelitian dalam skripsi ini adalah penelitian yang bersifat eksperimen yaitu menguji dan mengamati . Tahapan yang dilakukan dalam penelitian ini adalah penentuan jenis dan cara pengambilan data, variabel dan cara analisis yang digunakan, serta kerangka solusi masalah yang disajikan dalam bentuk diagram alir dan pembahasannya.

3.1 Skenario Pengujian

3.1.1 Rancangan Proses Kerja Aplikasi Android

Berikut merupakan blok diagram proses kerja aplikasi android secara keseluruhan:



Gambar 3.1 Blok diagram proses kerja aplikasi

Dari blok diagram di atas, proses yang terjadi pada sistem adalah sebagai berikut:

Ketika tombol “NYALAKAN SISTEM” diketuk, maka aplikasi akan mengirimkan status sistem “menyala” ke ThingSpeak. Aplikasi kemudian meminta data dari kanal ThingSpeak.

Data yang dimaksud adalah data sensor Ultrasonik HC-SR04. Data tersebut kemudian diproses aplikasi dan ditampilkan dalam bentuk lampu indikator dan status individual.

3.1.2 Pengujian Sistem GPS

Pada Tugas Akhir ini dilakukan pengujian yaitu, pengujian sistem GPS, pengujian sistem notifikasi dan lokasi. Pengujian Sistem GPS pengujian yang dilakukan dengan menghubungkan GPS dengan mikrokontroler dimana sistem GPS akan mengambil data koordinat titik baca motor. Pengujian akan dilakukan pada lokasi terbuka. Berikut pengujian Sistem GPS:

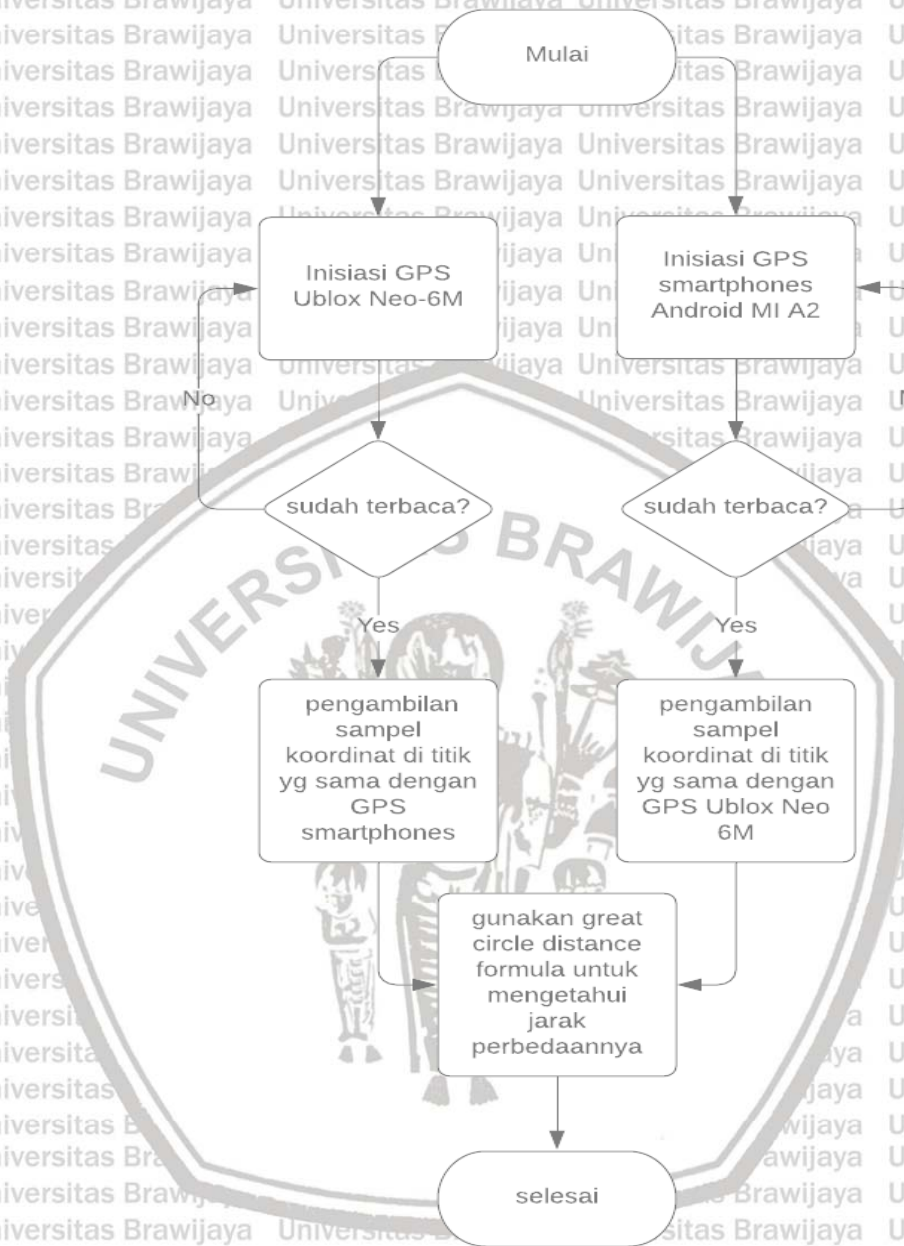
1. Pengujian Koordinat GPS Ublox Neo 6M

Pengujian hasil lokasi yang menjadi keluaran GPS (*Longitude* dan *Latitude*).

Pengujian akan dilakukan di beberapa titik, untuk menjadi pembandingan titik tersebut akurat yaitu dilakukan pengujian bersamaan dengan *smartphone* dengan fitur google maps sebagai pembandingan, karena GPS paling lumrah dan keakurasiannya sudah sering digunakan oleh orang-orang yaitu GPS *smartphone*.

2. Pengujian *Error GPS* Ublox Neo 6M dengan *GPS Smartphone*

Pengujian dilakukan dengan membandingkan akurasi koordinat GPS Ublox Neo 6M dengan koordinat GPS *Smartphone*. Kemudian melihat lebih akurat mana antara GPS dengan cara dilakukan perhitungan selisih jarak dari koordinat yang didapat dengan menggunakan metode *Great Circle Distance/Haversine Formula* yang dikali dengan radius 6371000 untuk mendapatkan *error* jarak dalam satuan meter.

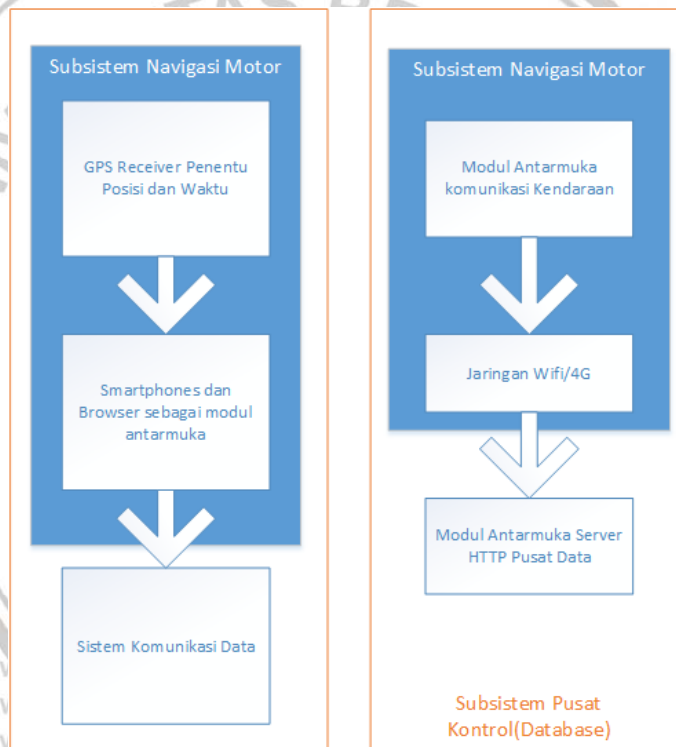


Gambar 3.2 Flowchart Pengambilan Data

Setelah dilakukan pengambilan data oleh masing-masing GPS dan di masing-masing tempat sebagai variabel bebas maka dilanjutkan analisis komparatif yaitu jarak hasil perhitungan dengan *great circle distance* dirata-ratakan dan dibandingkan antara rerata hasil variabel bebas lainnya

3.1.3 Sistem Pelacakan Operasional Menggunakan *Great Circle Distance*

Arsitektur sistem yang akan dibangun terbagi menjadi tiga sub sistem yaitu sub sistem navigasi kendaraan, subsistem komunikasi data dan sub sistem pusat kendali. Sub sistem navigasi kendaraan berfungsi sebagai penentu posisi dan waktu kendaraan kemudian dikirim ke sub sistem pusat kendali melalui data sub-sistem komunikasi. Sub-sistem ini terdiri dari penerima GPS sebagai alat penentuan posisi dan waktu, serta data antarmuka komunikasi berfungsi untuk menyaring data yang diperoleh dari *GPS receiver* dan kemudian dikirim ke pusat kendali. Untuk menjalankan antarmuka komunikasi kendaraan digunakan *smartphones* dengan program yang sudah diunggah ke mikrokontroler. Arsitektur sistem bisa dilihat pada **Gambar 3.3**.



Gambar 3.3 Arsitektur Sistem

Sub sistem komunikasi data berfungsi sebagai penghubung antara sub sistem navigasi kendaraan dan sub-sistem pusat kendali. Data navigasi kendaraan (posisi, tanggal, dan waktu) diperoleh pada navigasi kendaraan sub-sistem dikirimkan melalui sub-sistem ini ke pusat kendali. Sub-sistem ini terkait dengan kendaraan sub sistem navigasi dan sub sistem pusat kendali melalui modul antarmuka komunikasi yang ada pada keduanya sub-sistem. Sub-sistem ini terdiri dari kendaraan / armada dipantau dilengkapi dengan alat yang disebut

Alat Pelacak GPS. Perangkat ini akan menentukan posisi kendaraan yang dilalui sinyal dikirim dari satelit GPS dan secara otomatis mengirim posisi informasi ke sistem melalui MIFI atau modul wifi. Data tentang posisi kendaraan dikirim oleh perangkat melalui media komunikasi link browser di *smartphone* android dengan perantara modul wifi di jaringan 4G. Di dalam penelitian menggunakan Mifi dan kartu GSM telkomsel. Di pusat kendali, data dikirim dan diterima diteruskan ke server HTTP untuk diproses. Sub sistem pusat kendali merupakan pusat pengolahan data, visualisasi data, dan pengendalian kendaraan. Data posisi yang dikirimkan oleh kendaraan (motor) diolah dan divisualisasikan sehingga pengelola memiliki kemudahan untuk melacak pergerakan kendaraan.

3.2 Metode Pengumpulan Data

Pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam rangka mencapai tujuan penelitian. Metode pengumpulan data berupa suatu perntaan tentang sifat, keadaan, kegiatan tertentu dan sejenisnya. Pengumpulan data dilakukan untuk mendapatkan suatu informasi yang dibutuhkan dalam mencapai tujuan penelitian. Dalam penyusunan tugas akhir ini penulis mengambil objek penelitian pada Universitas Brawijaya yang berlokasi di Jl. Veteran, Ketawanggede, Kec. Lowokwaru, Kota Malang, Jawa Timur. Pengumpulan data dalam penelitian di Universitas Brawijaya Malang menggunakan 3 cara berikut merupakan uraian yang digunakan:

3.2.1 Observasi

Suatu metode pengumpulan data yang dilakukan dengan mengamati langsung, melihat dan mengambil suatu data yang dibutuhkan di tempat Penelitian itu dilakukan. Observasi juga bisa diartikan sebagai proses yang kompleks. Pengumpulan data yang dilakukan di Universitas Brawijaya Malang.

3.2.2 Penentuan Variabel Tetap

Pemberian perlakuan yang sama pada suatu pengujian sistem GPS, variable tetap tersebut adalah Sistem Pelacakan dan Deteksi Sepeda Motor Dari Pencurian, jumlah satelit yang terhubung yaitu tiga satelit.

3.2.3 Pengujian Variabel Bebas

Memberi perlakuan berbeda pada objek yang diuji (GPS), variable bebas tersebut adalah ruang terbuka dan ruang tertutup. Ruang terbuka sendiri terdiri dari ruang terbuka

padat gedung dan ruang terbuka minim gedung (lapangan rektorat), untuk ruang tertutup dikarenakan pengujian menggunakan objek bergerak yaitu sepeda motor maka penulis mengujinya di bawah kanopi/ruang terbuka yang tidak terdapat pintu tapi memiliki atap.

3.3 Dokumentasi

Suatu pengumpulan data dengan cara melihat langsung sumber-sumber Dokumen yang terkait. Dengan arti lain bahwa dokumentasi sebagai pengambilan data melalui dokumen tertulis maupun elektronik. Digunakan Sebagai mendukung kelengkapan data yang lain. Dokumentasi lainnya juga dibutuhkan sebagai pembacaan waktu ketika mengambil data koordinat GPS alat untuk dibandingkan dengan GPS *smartphones*.



BAB IV DESAIN DAN RANCANGAN SISTEM

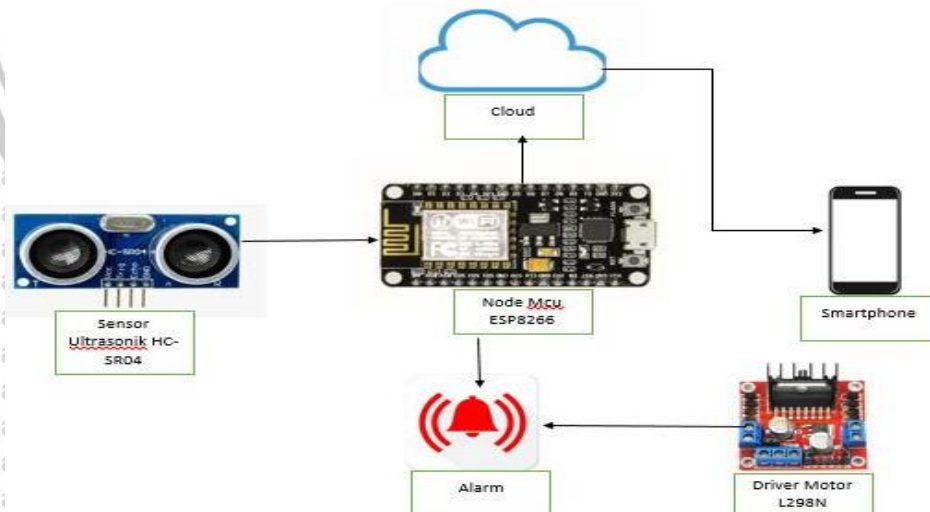
Pada bab ini akan dibahas mengenai gambaran umum sistem, spesifikasi sistem, perancangan mekanika pendeteksian sistem, perancangan komunikasi sistem, perancangan aplikasi android, proses kerja sistem, cara pengujiannya, serta pengambilan kesimpulan dan saran.

4.1 Pembuatan Perangkat Keras

Pembuatan perangkat keras dilakukan sebagai langkah awal sebelum terbentuknya suatu sistem beserta pemrogramannya. Hal ini dimaksudkan agar sistem dapat bekerja sesuai dengan deskripsi awal yang telah direncanakan. Pembuatan perangkat keras yang dilakukan pada penelitian ini meliputi:

4.1.1 Skema Pembuatan Perangkat Keras

Skema pembuatan perangkatan keras ditujukan untuk penempatan dalam menyambungkan antar komponen agar mengurangi *error* pemasangan. Gambar skema pembuatan perangkat keras dapat dilihat dalam **Gambar 4.1**



Gambar 4.1 Diagram Blok Umum Dari Sistem yang Dirancang

Skema kerja alat dimulai dengan pengambilan data menggunakan sensor *ultrasonic*. Data-data tersebut selanjutnya diolah oleh mikrokontroler untuk menentukan apakah motor sedang mengalami gejala pencurian motor. Apabila sistem mendeteksi positif terjadi gejala pencurian motor, yakni, sensor akan mengirimkan sinyal positif, maka mikrokontroler akan

mengirimkan notifikasi ke *smartphone* pengguna melalui IoT Node Mcu ESP8266 menggunakan modul wifi. Alarm pada sistem yang diletakkan di bagasi motor akan menyala, durasi 24ank eras suara dari alarm ditentukan dari jarak seseorang dengan sensor ultrasonic. Fungsi dari driver motor L298N disini adalah mengatur frekuensi dari buzzer. Semakin dekat seseorang dengan sensor makana suara alarm akan semakin besar dan begitupun sebaliknya, semakin jauh dari sensor maka suara alarm akan semakin kecil.

4.1.2 Penentuan Modul Elektrik Yang Digunakan

1. Mikrokontroller Node Mcu ESP8266

Mikrokontroller Node Mcu ESP8266 digunakan sebagai sistem utama atau main system dari kontrol logika fuzzy. Sistem minimum ini sebagai otak dari alat yang akan menerjemahkan data sensor ultrasonik yang kemudian digunakan untuk menggerakkan buzzer



Gambar 4.2 Node Mcu ESP8266

(Sumber: <https://www.jakartanotebook.com/v3-wireless-module-nodemcu-4m-bytes-lua-esp8266-esp-12e-for-arduino>)

2. Sensor Ultrasonik

Sensor ultrasonik digunakan sebagai data masukan alat untuk menggerakkan buzzer. Saat sensor ultrasonik mendeteksi seseorang, maka akan mengirimkan data yang didapat ke mikrokontroller node mcu ESP8266 untuk diolah sebagai data masukan.



Gambar 4.3 Sensor Ultrasonik

(Sumber: <https://www.tokopedia.com/arduinomakassar/sensor-ultrasonic-hc-sr04>)

3. Buzzer

Buzzer di alat ini digunakan sebagai output. Keras nya suara dari buzzer akan diatur oleh fuzzy logic. Semakin dekat seseorang maka suara dari buzzer akan semakin keras, begitupun sebaliknya.

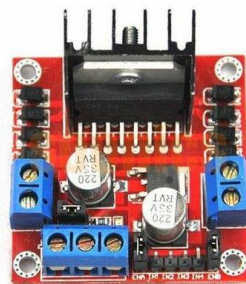


Gambar 4.4 Buzzer

(Sumber: <https://www.bukalapak.com/p/elektronik/komponen-elektronik/1lkx0tu-jual-buzzer-high-quality-3-24v-white-12v-continuous-sound-putih>)

4. Driver Motor L298N

Driver Motor L298N di alat ini digunakan untuk menaikkan tegangan PWM dari 5V ke 12V.



Gambar 4.5 Driver Motor L298N

(Sumber: <https://www.electrodragon.com/product/30183/>)

5. *Smartphone*

Disaat data sudah diterima oleh Mikrokontroler node mcu ESP8266 oleh sensor ultrasonik, data akan diolah oleh mikrokontroler terlebih dahulu, setelah itu akan dikirimkan ke *smartphone* berupa notifikasi



Gambar 4.6 *Smartphone*

(Sumber: <https://pngtree.com/so/mobile-mockup>)

Sistem yang dirancang diharapkan mampu:

1. Mendeteksi adanya seseorang di depan lubang kunci motor
2. Menyalakan alarm ketika sistem mendeteksi positif terjadi gejala pencurian motor.
3. Menghubungi pengguna melalui kiriman notifikasi pada aplikasi Android.
4. Keras suara dari alarm ditentukan dari jarak seseorang terhadap sensor

4.2 Spesifikasi Sistem

Spesifikasi sistem ditentukan dengan tujuan agar dapat membuat sistem sesuai dengan yang diharapkan. Sistem yang dirancang akan memiliki spesifikasi sebagai berikut.

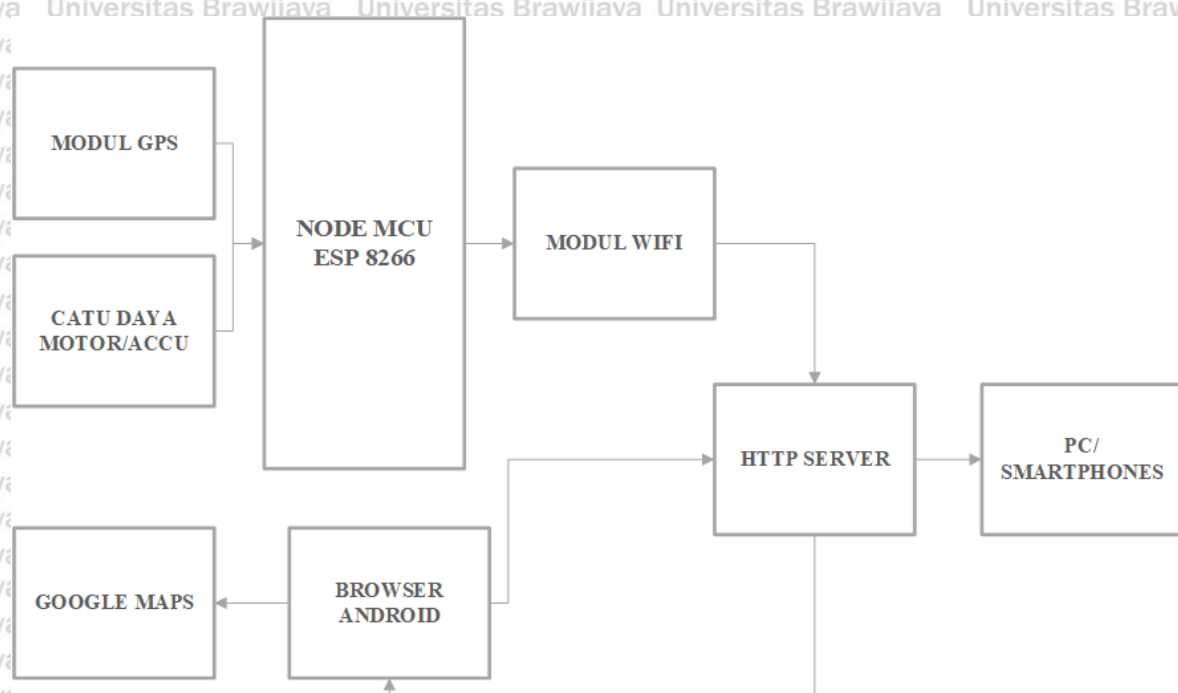
1. Sensor Ultrasonic HC-SR04 digunakan dengan tujuan mampu mendeteksi gerakan manusia di depan lubang kunci motor.
2. Board Node Mcu ESP8266 mampu memproses data yang diambil dari kedua sensor dan akselerometer dan melakukan proses klarifikasi.

3. Modul Wifi digunakan dengan tujuan untuk menyambungkan mikrokontroler dengan *cloud Thingspeak*.
4. Buzzer digunakan untuk output alarm sederhana. Untuk pengaplikasiannya bisa diganti dengan alarm lain yang lebih cocok dan mumpuni.
5. *Thingspeak* digunakan sebagai perantara mikrokontroler dengan aplikasi sederhana yang akan dibuat untuk Android user.
6. Android mampu menampilkan data dari input sensor ultrasonic
7. Notepad++ digunakan sebagai alat untuk membuat aplikasi sederhana yang telah disebutkan



4.3 Perancangan Blok Diagram Sistem

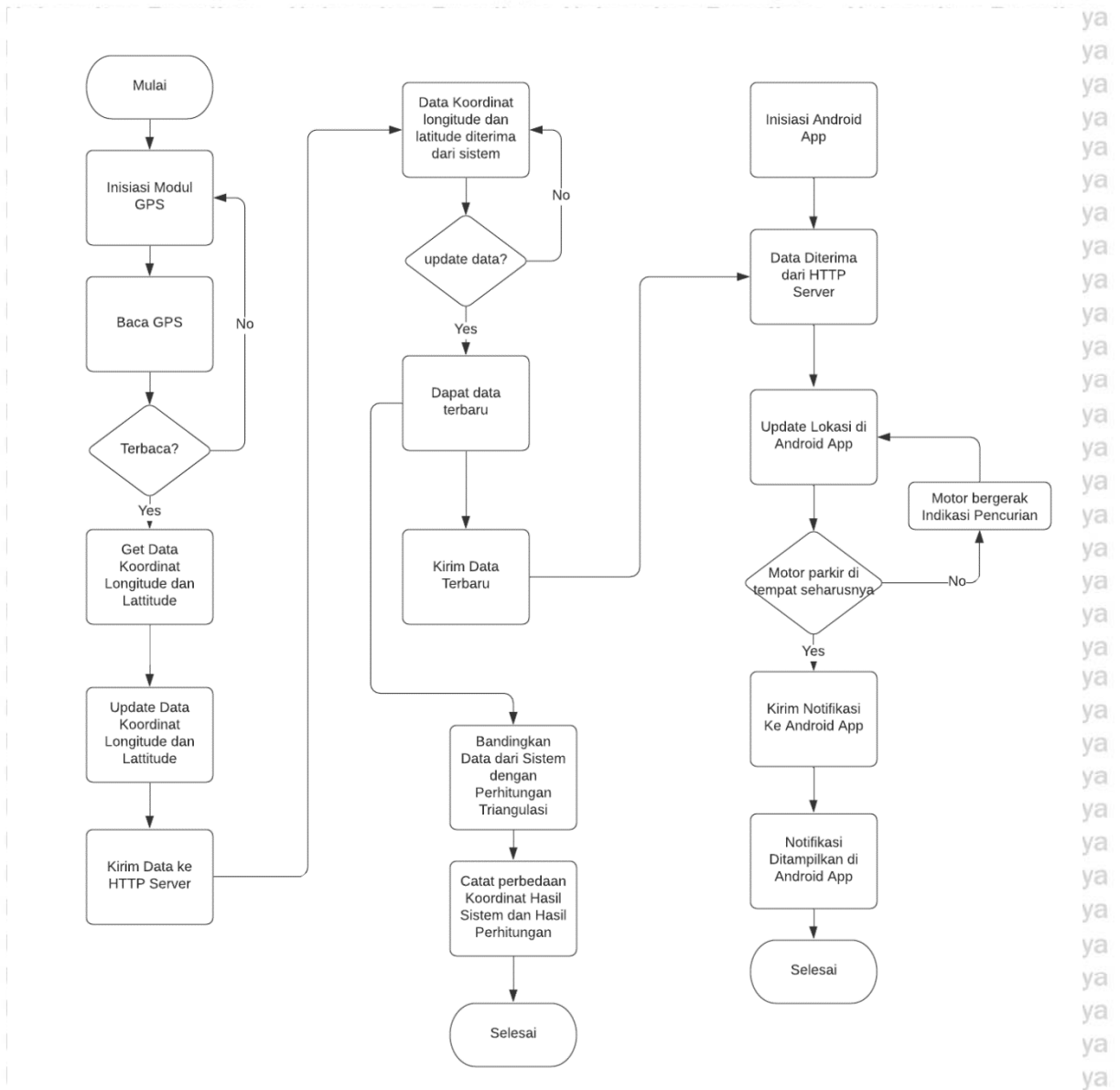
Tahapan awal perancangan alat yaitu diperlukan gambaran awal tentang bagaimana sistem kerja dari alat, seperti yang ditunjukkan oleh **Gambar 4.7** di bawah ini:



Gambar 4.7 Diagram Blok Sistem

4.3.1 Rancangan Perangkat Lunak

Perancangan perangkat lunak pada alat ini menggunakan software Arduino IDE sebagai *programmer* mikrokontroler Node MCU ESP8266. Untuk membuat program yang sesuai dengan tujuan penelitian, maka terlebih dahulu dibuat rancangan program dalam suatu diagram alir (*flowchart*) dari program yang dibuat agar alat bekerja berdasarkan program yang diinginkan. Secara umum diagram alir program utama dapat dilihat pada **Gambar 4.8** dimana program dimulai dengan melakukan inisialisasi semua variabel-variabel utama dari proses pengontrolan.



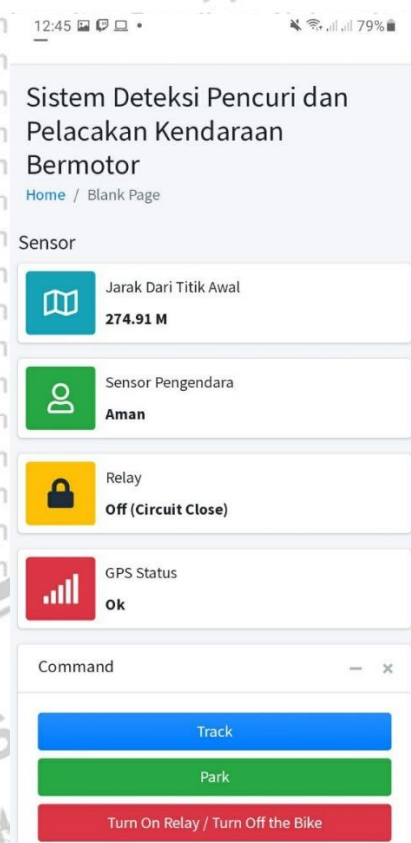
Gambar 4.8 Diagram Alir Perangkat Lunak

Ketika sistem diinisiasi semua modul akan menyala termasuk GPS, GPS akan mengirimkan koordinat ke server yg akan ditampilkan ke link browser *smartphones* jika ada indikasi pencurian yaitu motor bergerak sejauh 50 meter dari posisi awal, lalu pengguna bisa mengamati melalui link browser tadi yang memiliki fitur on/off relay untuk mematikan mesin motor.

4.4 Perancangan Aplikasi Android

4.4.1 Rancangan Antarmuka Aplikasi Android

Aplikasi android akan diberi nama “Sistem Deteksi Maling dan Pelacakan Kendaraan Bermotor” Rancangan antarmuka aplikasi android dapat dilihat pada **Gambar 4.9**, dan **4.10**.

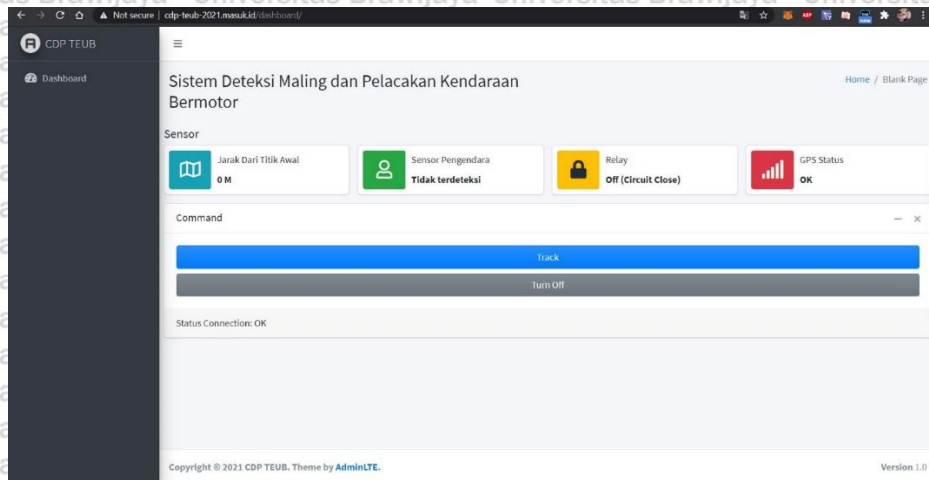


Gambar 4.9 Tampilan Link Antarmuka di *Browser Smartphone*

Berikut penjelasan untuk setiap bagian dari antarmuka aplikasi:

1. Judul halaman: menunjukkan judul halaman. Dalam aplikasi ini hanya akan ada 3 halaman, yaitu halaman “HOME” (halaman utama), halaman “Lokasi Motor”, dan halaman “Alert” yang hanya akan muncul ketika pengguna mengetuk notifikasi dari aplikasi android ini.
2. Nama aplikasi: menampilkan nama aplikasi. Dalam hal ini, nama aplikasi adalah “Sistem Deteksi Maling dan Pelacakan Kendaraan Bermotor”
3. Jarak Dari Titik Awal: Memberitahukan posisi letak motor sejauh mana dari titik awal.
4. Sensor Pengendara: Memberitahukan ada atau tidaknya seseorang diatas jok motor, yang dibaca oleh sensor ultrasonik
5. *Relay* : Memberitahukan kondisi relay yang on/off
6. *GPS* : menunjukkan status lokasi dari kendaraan pengguna secara real time
7. *Track* : Berfungsi untuk melacak kendaraan pengguna secara real time, disaat tombol ini ditekan, lokasi motor pengguna akan langsung diberitahu dilokasi mana secara real time

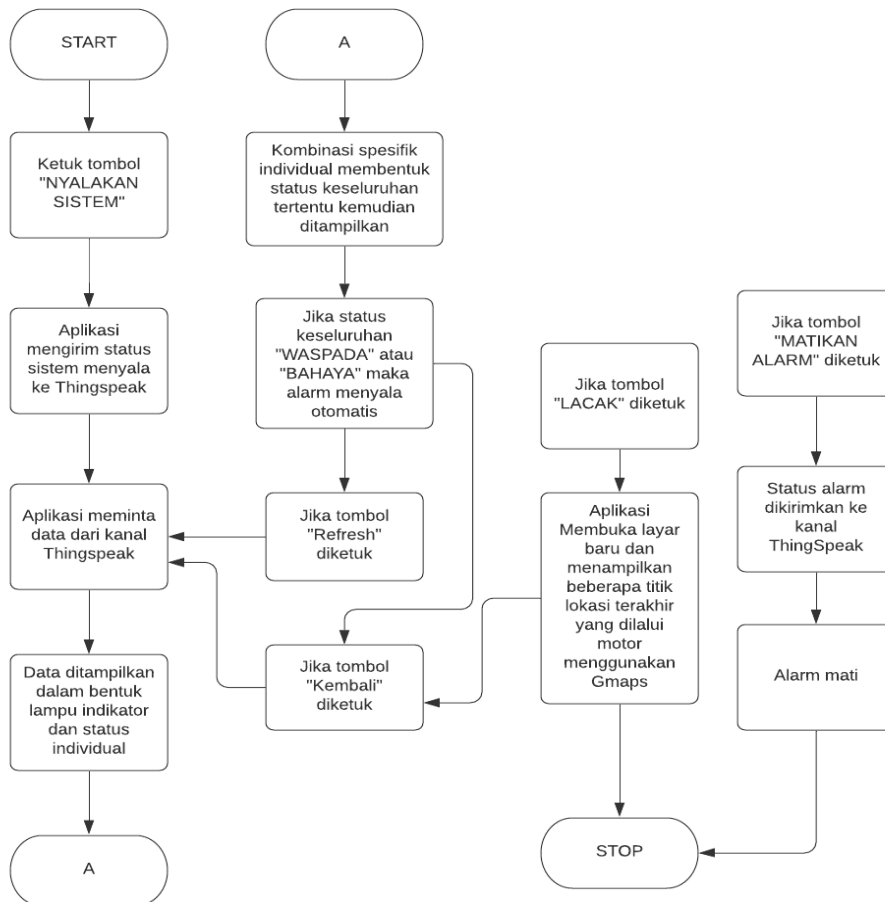
8. *Turn off* : Berfungsi untuk mematikan mesin dari kendaraan bermotor pengguna, disaat tombol ini ditekan motor pengguna akan langsung secara otomatis akan mati



Gambar 4.10 Halaman aplikasi disaat dibuka dengan laptop

4.4.2 Rancangan Proses Kerja Aplikasi Android

Berikut merupakan blok diagram proses kerja aplikasi android secara keseluruhan:



Gambar 4.11 Blok diagram proses kerja aplikasi

Dari blok diagram di atas, proses yang terjadi pada sistem adalah sebagai berikut: Ketika tombol “NYALAKAN SISTEM” diketuk, maka aplikasi akan mengirimkan status sistem “menyala” ke *server*. Aplikasi kemudian meminta data dari kanal *server*. Data yang dimaksud adalah data sensor Ultrasonik HC-SR04. Data tersebut kemudian diproses aplikasi dan ditampilkan dalam bentuk lampu indikator dan status individual.



BAB V
HASIL PENELITIAN DAN PEMBAHASAN

5.1 Hasil Penelitian

Hasil penelitian dan pembahasan dalam skripsi ini berupa analisis mengenai mekanisme pengambilan data koordinat GPS, pembahasan tentang tingkat akurasi serta faktor-faktor yang mempengaruhi ketidakakuratan GPS yang diambil dari pemberian variabel bebas sebelumnya.

5.1.1 Data Koordinat Pada Ruang Terbuka Padat Gedung

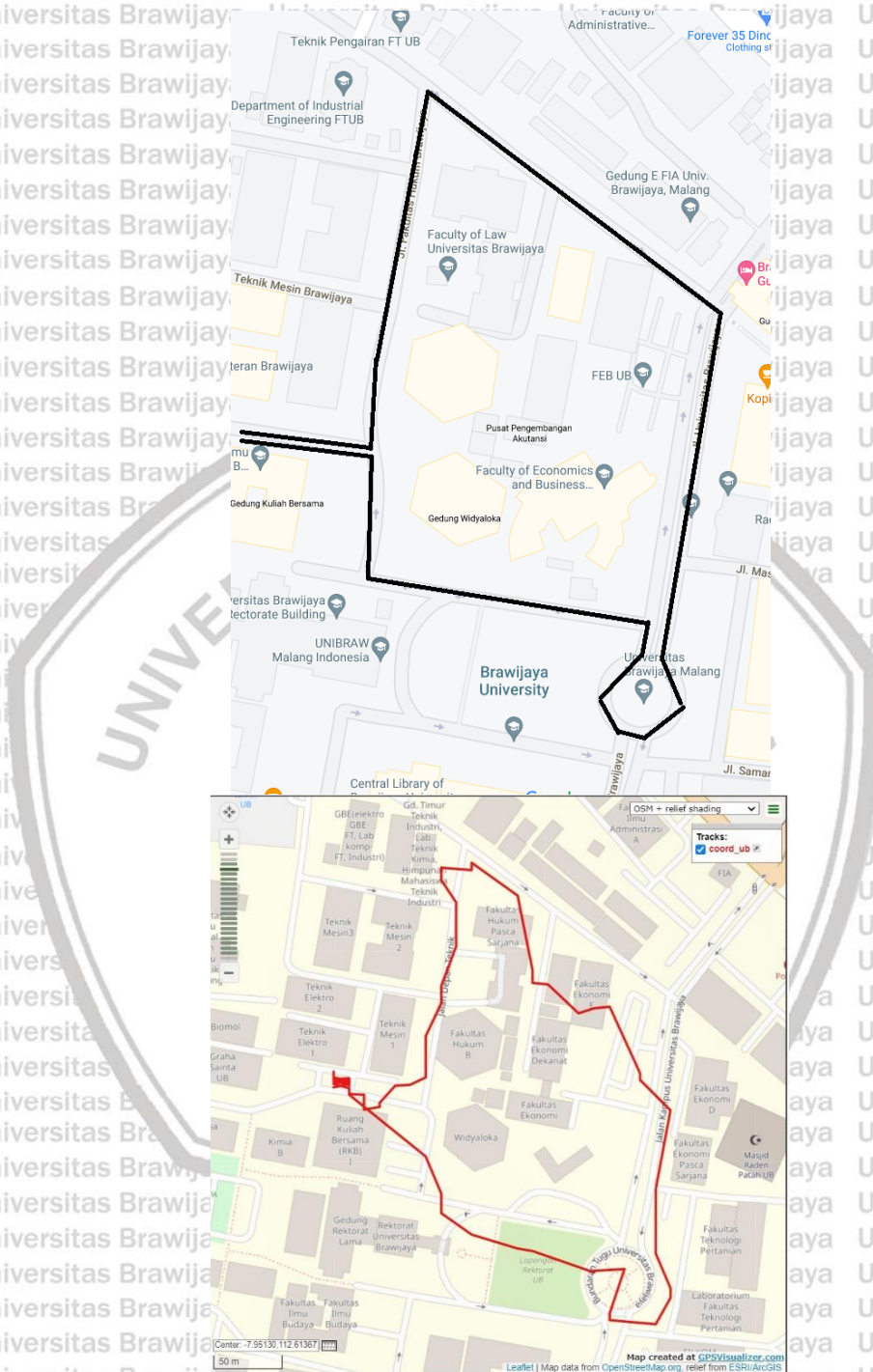
Waktu	Latitude Gmaps	Longitude Gmaps	Latitude Alat	Longitude Alat	Perbedaan jarak (m)
13:55	-7,951284	112,612653	-7,95127	112,6126	6,041645367
13:58	-7,951379	112,612904	-7,9513	112,6126	34,61156608
13:58	-7,951224	112,613165	-7,951421	112,6129	36,48987333
13:59	-7,950649	112,613286	-7,951253	112,6132	67,82620456
13:59	-7,949901	112,613405	-7,950663	112,6133	85,51594373
14:00	-7,949942	112,613733	-7,94998	112,6133	47,87149466
14:00	-7,950522	112,614510	-7,94988	112,6135	132,1653472
14:01	-7,951571	112,614742	-7,951229	112,6146	41,11838344
14:02	-7,952680	112,614639	-7,952654	112,6148	17,96435737
14:02	-7,952202	112,614077	-7,952505	112,6143	41,69237933
14:03	-7,951764	112,613192	-7,951766	112,6132	0,910587389

Tabel 5.1 Data koordinat Ruang Terbuka Padat Gedung

- Rata-Rata Jarak Lingkaran Besar Pada Pengujian Ruang Terbuka Padat Gedung Tinggi

$$\frac{\sum \text{Perbedaan Jarak}}{\text{Jumlah Pengambilan}} = 46,56 \text{ meter}$$

Akurasi GPS Ublox Neo 6M akurat sampai 46,56 meter jika digunakan di ruang terbuka padat gedung tinggi.



Gambar 5.1 Pengujian Ruang Terbuka Padat Gedung Tinggi

5.1.2 Data Koordinat Pada Ruang Terbuka Minim Gedung (Lapangan)

Waktu	Latitude Gmaps	Longitude Gmaps	Latitude Alat	Longitude Alat	Perbedaan jarak (m)
13:06	-7,952685	112,613430	-7,9531000	112,6132	52,64023163
13:07	-7,952768	112,613786	-7,9528510	112,6137	13,22363849
13:07	-7,952850	112,614186	-7,9528160	112,6133	97,64440378
13:08	-7,952737	112,614336	-7,9527640	112,6143	4,973921103
13:08	-7,952567	112,614267	-7,9526280	112,6143	7,694445869
13:09	-7,952314	112,614462	-7,9523230	112,6142	28,87032204
13:09	-7,952223	112,614227	-7,9522910	112,6140	26,11687129
13:09	-7,952189	112,613944	-7,9521960	112,6137	26,88194844
13:10	-7,952131	112,613568	-7,9521430	112,6136	3,768823717
13:10	-7,952066	112,613195	-7,9520410	112,6134	22,74646569

Tabel 5.2 Data Koordinat Ruang Terbuka Minim Gedung (Lapangan)

- Rata-Rata Jarak Lingkaran Besar Pada Pengujian Ruang Terbuka Minim Gedung Tinggi (Lapangan Rektorat)

$$\frac{\sum \text{Perbedaan Jarak}}{\text{Jumlah Pengambilan}} = 28,453 \text{ meter}$$

Akurasi GPS Ublox Neo 6M akurat sampai 28,453 meter jika digunakan di ruang terbuka minim gedung tinggi (Lapangan Rektorat).



Gambar 5.2 Pengujian Ruang Terbuka Minim Gedung (Lapangan)

5.1.3 Data Koordinat Pada Ruang Terbuka Beratap/Kanopi

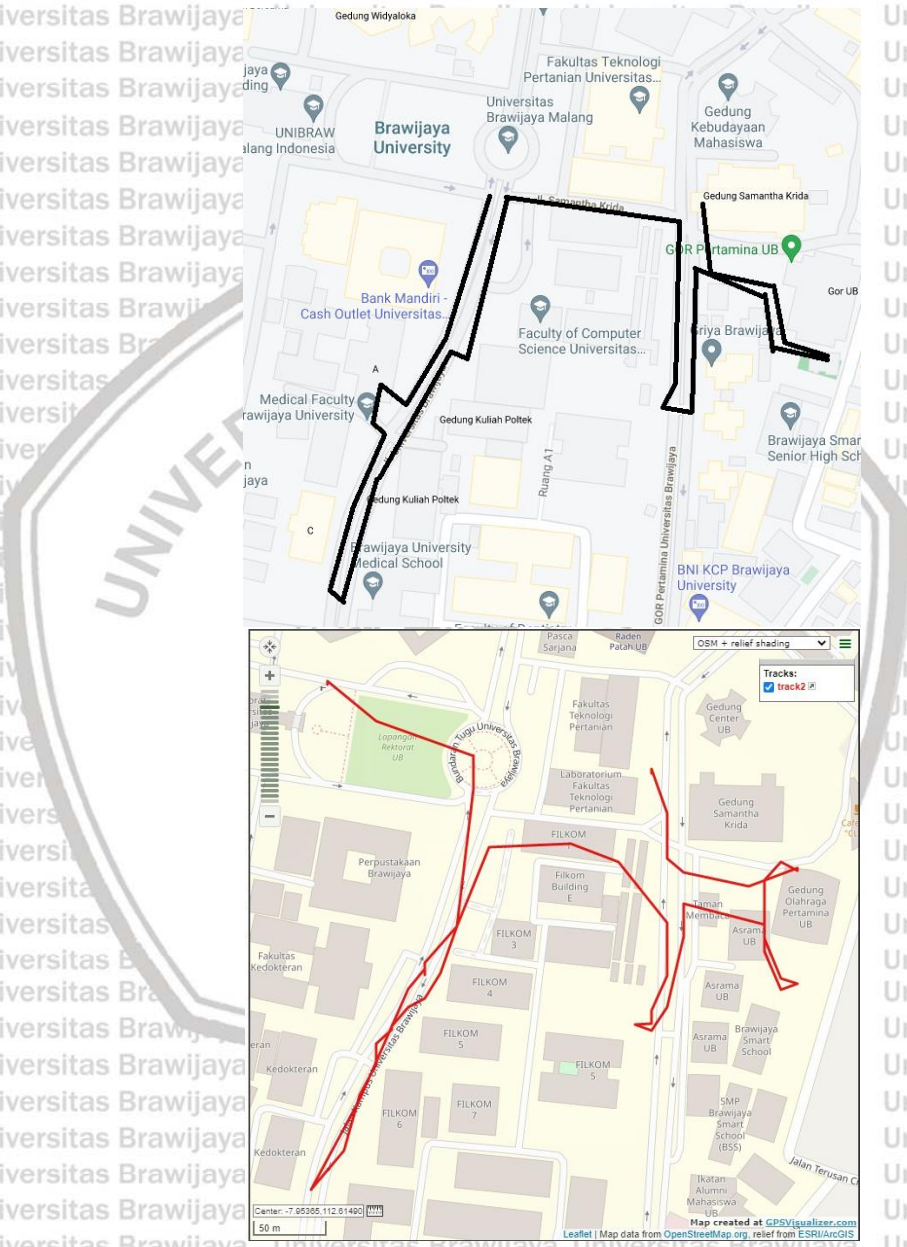
Waktu	Latitude Gmaps	Longitude Gmaps	Latitude Alat	Longitude Alat	Perbedaan Jarak (m)
13:17	-7,952938	112,615706	-7,9526590	112,6155	38,43320437
13:19	-7,953948	112,616485	-7,9539830	112,6163	20,74157027
13:20	-7,954132	112,615351	-7,9542290	112,6155	19,63600725
13:21	-7,954041	112,614230	-7,9538890	112,6141	22,15025847
13:23	-7,953985	112,613733	-7,9542270	112,6139	32,5932387
13:24	-7,951971	112,613510	-7,9523410	112,6138	52,08280579

Tabel 5.3 Data Koordinat Pada Ruang Terbuka Beratap/Kanopi

- Rata-Rata Jarak Lingkaran Besar Pada Pengujian Ruang Tertutup (beratap) atau Kanopi

$$\frac{\sum \text{Perbedaan Jarak}}{\text{Jumlah Pengambilan}} = 30,936 \text{ meter}$$

Akurasi GPS Ublox Neo 6M akurat sampai 30,936 meter jika digunakan di ruang tertutup beratap (kanopi).



Gambar 5.3 Pengujian Ruang Terbuka Beratap/Kanopi

BAB VI KESIMPULAN DAN SARAN

6.1 Kesimpulan

Setelah dilakukan berbagai macam pengujian dan analisis data, maka dapat disimpulkan sebagai berikut:

1. Error Koordinat

Dengan menggunakan metode *Great-Circle Distance*, error atau ketidakakuratan pembacaan koordinat oleh GPS Ublox Neo-6M bisa terlihat dalam satuan meter dan cara menganalisisnya pun memiliki beberapa syarat yaitu memiliki GPS/device lain yang tingkat akurasinya lebih tinggi sehingga bisa dijadikan tolok ukur, data koordinat dari GPS referensi dan GPS Ublox Neo-6M dibandingkan sehingga terlihat perbedaannya.

2. Lokasi

Hasil keluaran dari GPS Ublox Neo-6M tidak terlalu akurat atau tidak mendekati tempat aslinya dikarenakan rata-rata perbedaan data koordinat kedua GPS tersebut 35,316 meter.

3. Tingkat Akurasi GPS Ublox Neo-6M

- a. Akurasi GPS Ublox Neo 6M akurat sampai 46,56 meter jika digunakan di ruang terbuka padat gedung tinggi
- b. Akurasi GPS Ublox Neo 6M akurat sampai 28,453 meter jika digunakan di ruang terbuka minim gedung tinggi (lapangan rektorat).
- c. Akurasi GPS Ublox Neo 6M akurat sampai 30,936 meter jika digunakan di ruang tertutup beratap (kanopi).

4. *Multipath Signal*

Pemberian perlakuan pengujian yang berbeda-beda didasarkan pada sinyal radio/GPS yang dipakai akan terpantul jika bertemu dengan objek seperti gedung atau batu besar. Akurasi semakin tinggi jika tempat pengujian tidak terhalang gedung-gedung tinggi dan berlaku sebaliknya, seperti tempat pengujian bisa diurutkan dari yang paling akurat yaitu ruang terbuka minim gedung, ruang terbuka beratap (kanopi), dan ruang terbuka padat gedung tinggi.

6.2 Saran

Saran yang dapat diberikan untuk penelitian selanjutnya yaitu

- a. memberikan komponen tambahan pada sistem yang berupa *amplifier* GPS untuk mengurangi ketidakakuratan pembacaan koordinat.
- b. pengujian akurasi harus berdasarkan jumlah satelit yang sama, untuk menampilkan jumlah satelit yang terhubung dibutuhkan program tambahan



DAFTAR PUSTAKA

- BPS Kota Malang. (2017). *Jumlah Kendaraan Bermotor Berdasarkan Plat Nomor di Kota Malang, 2015-2016*. <https://malangkota.bps.go.id/statictable/2017/07/21/601/jumlah-kendaraan-bermotor-berdasarkan-plat-nomor-di-kota-malang-2015-2016.html>. (diakses 3 April 2020).
- BPS Subdirektorat Statistik Politik dan Keamanan. (2019). *Statistik Kriminal 2019*. Jakarta: Badan Pusat Statistik.
- Components 101. (2017). *Active Passive Buzzer*. <https://components101.com/buzzer-pinout-working-datasheet>. (diakses 18 Mei 2020).
- Components 101. (2017). *HC-SR501 PIR Sensor*. <https://components101.com/hc-sr501-pir-sensor>. (diakses 18 Mei 2020).
- Components 101. (2018). *NEO-6MV2 GPS Module*. <https://components101.com/modules/neo-6mv2-GPS-module>. (diakses 18 Mei 2020).
- Electronics For U. (2019). *All You Wanted to Know About GSM Module and GPRS Module*. <https://www.electronicsforu.com/resources/gsm-module>. (diakses 18 Mei 2020).
- Jawa Pos Radar Malang. (2017). *Kota Malang Darurat Curanmor, 9 Bulan 625 Motor Hilang*. <https://radarmalang.id/kota-malang-darurat-curanmor-9-bulan-625-motor-hilang/>. (diakses 3 April 2020).
- Kadir, A. (2012). *Panduan Praktis Mempelajari Aplikasi Mikrokontroler dan Pemrogramannya menggunakan Arduino*. Yogyakarta: Andi.
- Kartha, V. (n.d.). *Interfacing MPU-6050 / GY-521 board with Arduino Uno*. <https://electrosome.com/interfacing-mpu-6050-gy-521-arduino-uno/>. (diakses 18 Mei 2020).
- Last Minute Engineers. (n.d.). *Send Receive SMS & Call with SIM800L GSM Module & Arduino*. <https://lastminuteengineers.com/sim800l-gsm-module-arduino-tutorial/>. (diakses 18 Mei 2020).
- Ruth, E. (2013). Deskripsi Kualitas Layanan Jasa Akses Internet di Indonesia dari Sudut Pandang Penyelenggara. *Buletin Pos dan Telekomunikasi*, Vol.11 No.2 Juni, 137-146.
- Santos, S. (2018). *Guide to NEO-6M GPS Module with Arduino*. <https://randomnerdtutorials.com/guide-to-neo-6m-gps-module-with-arduino/>. (diakses 18 Mei 2020).
- Sulaiman, A. (2012). *Arduino: Mikrocontroller Bagi Pemula Hingga Mahir*. <http://buletin.balaelektronika.com/?p=163>. (diakses 2 Mei 2014).
- Wikipedia. (2020). *GSM*. https://en.wikipedia.org/wiki/GSM#cite_ref-2.

Great Circle Distance Methode for Improving Operational Control System Based on GPS Tracking System, 2012,
https://www.researchgate.net/publication/267844420_Great_Circle_Distance_Method_for_Improving_Operational_Control_System_Based_on_GPS_Tracking_System

The Haversine Equation In Excel, Ken Burnside, <https://www.techwalla.com/articles/the-haversine-equation-in-excel>

Calculate Distance, Bearing and More Between Latitude/Longitude Points,
<http://www.movable-type.co.uk/scripts/latlong.html?from=48.97097,-122.02893&to=48.965496,-122.072989>

PENGUNAAN GLOBAL POSITIONING SYSTEM (GPS) UNTUK DASAR SURVEY PADA MAHASISWA, 2019, <https://ejournal.upr.ac.id/index.php/JBL/article/view/553>

PENTINGNYA AKURASI DAN PRESISI ALAT UKUR DALAM RUMAH TANGGA, 2017,
<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwilzNSvw4ryAhVPVH0KHSJnAkkQFjAIegQIHhAD&url=https%3A%2F%2Ffejurnal.umri.ac.id%2Findex.php%2FPengabdianUMRI%2Farticle%2Fdownload%2F237%2F143%2F&usq=AOvVaw02FDekGJNy6zoyD0PDWBCv>



LAMPIRAN

a. Perhitungan Great-Circle Distance:

A	B	C	D	E	F	G	H	I	J	K	L
Waktu	latitude gmaps	longitude gmaps	latitude alat	longitude alat	Perbedaan Jarak (m)	keterangan	delta latitude	delta longitude			
2	13:55	-7,951284	112,612653	-7,95127	112,6126	6,041645367		-0,000014	0,000053		
3	13:58	-7,951379	112,612904	-7,9513	112,6126	34,61156608		-0,000079	0,000304		
4	13:58	-7,951224	112,613165	-7,951421	112,6129	36,48987335		0,000197	0,000265		
5	13:59	-7,950649	112,613286	-7,951253	112,6132	67,82620456		0,000604	0,000086		
6	13:59	-7,949901	112,613405	-7,950663	112,6133	85,51594373		0,000762	0,000105		
7	14:00	-7,949942	112,613733	-7,94998	112,6133	47,87149466	Outdoor(Padat Gedung Tinggi)	0,000338	0,000433		
8	14:00	-7,950522	112,614510	-7,94988	112,6135	132,1653472		-0,000642	0,001010		
9	14:01	-7,951571	112,614742	-7,951229	112,6146	41,11838344		-0,000342	0,000142		
10	14:02	-7,952680	112,614639	-7,952654	112,6148	17,96435737		-0,000026	-0,000161		
11	14:02	-7,952202	112,614077	-7,952505	112,6143	41,69237933		0,000303	-0,000223		
12	14:03	-7,951764	112,613192	-7,951766	112,6132	0,910587389		0,000002	-0,000008		
13	13:06	-7,952685	112,613430	-7,953100	112,6132	52,64023163		0,000415	0,000230		
14	13:07	-7,952768	112,613786	-7,9528510	112,6137	13,22363849		0,000083	0,000086		
15	13:07	-7,952850	112,614186	-7,9528160	112,6133	97,64440378		-0,000034	0,000086		
16	13:08	-7,952737	112,614336	-7,9527640	112,6143	4,973921103		0,000027	0,000036		
17	13:08	-7,952567	112,614267	-7,9526280	112,6143	7,694445869		0,000061	-0,000033		
18	13:09	-7,952314	112,614462	-7,9523230	112,6142	28,87032204	Outdoor(Lapangan Rektorat)	0,000009	0,000262		
19	13:09	-7,952223	112,614227	-7,9522910	112,6140	26,11687129		0,000068	0,000227		
20	13:09	-7,952189	112,613944	-7,9521960	112,6137	26,88194844		0,000007	0,000244		
21	13:10	-7,952131	112,613568	-7,9521430	112,6136	3,768823717		0,000012	-0,000032		
22	13:10	-7,952066	112,613195	-7,9520410	112,6134	22,74646569		-0,000025	-0,000205		
23	13:17	-7,952938	112,615706	-7,9526590	112,6155	38,43320437		-0,000279	0,000206		
24	13:19	-7,953948	112,616485	-7,9539830	112,6163	20,74157027		0,000035	0,000185		
25	13:20	-7,954132	112,615351	-7,9542290	112,6155	19,63600725		0,000097	-0,000149		
26	13:21	-7,954041	112,614230	-7,9538890	112,6141	22,15025847	Indoor(Di bawah Kanopi)	-0,000152	0,000130		
27	13:23	-7,953985	112,613733	-7,9542270	112,6139	32,5932387		0,000242	-0,000167		
28	13:24	-7,951971	112,613510	-7,9523410	112,6138	52,08280579		0,000370	-0,000290		

b. Listing Program GPS:

```

1. while (ss.available() > 0)
2. {
3.   if (gps.encode(ss.read()))
4.   {
5.     if (gps.location.isValid())
6.     {
7.       latitude = gps.location.lat();
8.       lat_str = String(latitude , 6);
9.       longitude = gps.location.lng();
10.      lng_str = String(longitude , 6);
11.    }
12.  }
13. }

```

c. Listing Program Penunjuk Jumlah Satelit:

```

1. #include <TinyGPS++.h>
2. #include <SoftwareSerial.h>
3. float lat, lon;
4. int year, month, date, hour, minute, second;
5. String date_str, time_str, lat_str, lon_str, sat_str;
6. float park_lat, park_lon;

```



```
7. int Sat;
8. TinyGPSPlus gps;
9. int park = 0;
10. unsigned long get_distance;
11. SoftwareSerial GpsSerial(4, 5);
12. void setup() {
13.   Serial.begin(9600);
14.   if(GpsSerial.available() > 0)
15.   {
16.     Serial.println("GPS Serial OK");
17.   }
18.   GpsSerial.begin(9600);
19. }
20. void loop() {
21.   // Ambil Data GPS
22.   while(GpsSerial.available() > 0)
23.   {
24.     if(gps.encode(GpsSerial.read()))
25.     {
26.       if(gps.location.isValid())
27.       {
28.         lat = gps.location.lat();
29.         lat_str = String(lat, 6);
30.         lon = gps.location.lng();
31.         lon_str = String(lon, 6);
32.         Sat = gps.satellites.value();
33.         sat_str = String(Sat);
34.       }
35.     }
36.   }
37.   Serial.print("Latitude: ");
38.   Serial.print(lat_str);
39.   Serial.print(" - Longitude: ");
40.   Serial.print(lon_str);
41.   Serial.print(" - Jumlah Satelit: ");
42.   Serial.println(sat_str);
43.   delay(1000);
44. }
```



d. Listing Program Server:

```

1. <?php
2. $servername = "localhost";
3. $username = "cdpteubm_motor";
4. $password = "Malang123@";
5. $dbname = "cdpteubm_motor";
6.
7. try {
8.     $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);
9.     $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
10.    $stmt = $conn->prepare("SELECT id, x, y FROM coord WHERE gps_status = '1' ORDER
    BY id DESC LIMIT 1");
11.    $stmt->execute();
12.
13.    // set the resulting array to associative
14.    $stmt->setFetchMode(PDO::FETCH_ASSOC);
15.    /*foreach(new TableRows(new RecursiveArrayIterator($stmt->fetchAll())) as
    $k=>$v) {
16.        echo $v;
17.    }*/
18.    $pog = $stmt->fetchAll();
19.    header('Location: https://www.google.com/maps/search/?api=1&query=' . $pog[0][x]
    . ', ' . $pog[0][y]);
20.    /*echo '<a href="https://www.google.com/maps/search/?api=1&query=' . $pog[0][x]
    . ', ' . $pog[0][y] . '">Test Here</a>';*/
21.    //print_r($stmt->fetchAll());
22. } catch(PDOException $e) {
23.     echo "Error: " . $e->getMessage();
24. }
25. $conn = null;
26. //echo "</table>";
27. ?>

```

```

1. <?php
2. $servername = "localhost";
3. $username = "cdpteubm_motor";
4. $password = "Malang123@";
5. $dbname = "cdpteubm_motor";
6.
7. function distance($lat1, $lon1, $lat2, $lon2, $unit) {
8.     if (($lat1 == $lat2) && ($lon1 == $lon2)) {
9.         return 0;
10.    }
11.    else {
12.        $theta = $lon1 - $lon2;
13.        $dist = sin(deg2rad($lat1)) * sin(deg2rad($lat2)) + cos(deg2rad($lat1))
    * cos(deg2rad($lat2)) * cos(deg2rad($theta));
14.        $dist = acos($dist);
15.        $dist = rad2deg($dist);
16.        $miles = $dist * 60 * 1.1515;
17.        $unit = strtoupper($unit);
18.
19.        if ($unit == "K") {
20.            return ($miles * 1.609344);
21.        } else if ($unit == "N") {
22.            return ($miles * 0.8684);
23.        } else {
24.            return $miles;
25.        }
26.    }
27. }
28.
29. try {

```

```

30.     $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username,
31.     $password);
32.     $stmt = $conn->prepare("SELECT relay,lastupd,x,y,sensor FROM state WHERE
33.     device = '1337' LIMIT 1");
34.     $stmt->execute();
35.     $pog = $stmt->fetchAll();
36.     $relay = $pog[0][relay] == 1? TRUE : FALSE;
37.     $last_command = $pog[0][lastupd];
38.     $park_x = $pog[0][x];
39.     $park_y = $pog[0][y];
40.     $sensor = $pog[0][sensor];
41.     $stmt = $conn->prepare("SELECT waktu, gps_status FROM coord ORDER BY id
42.     DESC LIMIT 1");
43.     $stmt->execute();
44.     $pog = $stmt->fetchAll();
45.     $last_send = $pog[0][waktu];
46.     $status_gps = $pog[0][gps_status] == 1? TRUE : FALSE;
47.     $stmt = $conn->prepare("SELECT waktu,x,y FROM coord WHERE gps_status = '1'
48.     ORDER BY id DESC LIMIT 1");
49.     $stmt->execute();
50.     $pog = $stmt->fetchAll();
51.     $last_valid = $pog[0][waktu];
52.     $last_x = $pog[0][x];
53.     $last_y = $pog[0][y];
54.     } catch(PDOException $e) {
55.     echo "Error: " . $e->getMessage();
56.     }
57. ?>
58. <!DOCTYPE html>
59. <html lang="en">
60. <head>
61.     <meta charset="utf-8">
62.     <meta name="viewport" content="width=device-width, initial-scale=1">
63.     <title>AdminLTE 3 | Blank Page</title>
64.
65.     <!-- Google Font: Source Sans Pro -->
66.     <link rel="stylesheet"
67.     href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700&disp
68.     lay=fallback">
69.     <!-- Font Awesome -->
70.     <link rel="stylesheet" href="raw/plugins/fontawesome-free/css/all.min.css">
71.     <!-- Theme style -->
72.     <link rel="stylesheet" href="raw/dist/css/adminlte.min.css">
73. </head>
74. <body class="hold-transition sidebar-mini">
75. <!-- Site wrapper -->
76. <div class="wrapper">
77.     <!-- Navbar -->
78.     <nav class="main-header navbar navbar-expand navbar-white navbar-light">
79.     <!-- Left navbar links -->
80.     <ul class="navbar-nav">
81.     <li class="nav-item">
82.     <a class="nav-link" data-widget="pushmenu" href="#" role="button"><i
83.     class="fas fa-bars"></i></a>
84.     </li>
85.     </ul>
86. </nav>
87. <!-- /.navbar -->
88. <!-- Main Sidebar Container -->
89. <aside class="main-sidebar sidebar-dark-primary elevation-4">

```

```

awijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya
awijaya Un 89. <!-- Brand Logo --> ya
awijaya Un 90. <a href="#" class="brand-link"> ya
awijaya Un 91.  ya
awijaya Un 92. <span class="brand-text font-weight-light">CDP TEUB</span> ya
awijaya Un 93. </a> ya
awijaya Un 94. ya
awijaya Un 95. <!-- Sidebar --> ya
awijaya Un 96. <div class="sidebar"> ya
awijaya Un 97. <!-- Sidebar user (optional) --> ya
awijaya Un 98. ya
awijaya Un 99. <!-- SidebarSearch Form --> ya
awijaya Un 100. ya
awijaya Un 101. <!-- Sidebar Menu --> ya
awijaya Un 102. <nav class="mt-2"> ya
awijaya Un 103. <ul class="nav nav-pills nav-sidebar flex-column" data-widget="treeview" ya
awijaya Un role="menu" data-accordion="false"> ya
awijaya Un 104. <!-- Add icons to the links using the .nav-icon class ya
awijaya Un 105. with font-awesome or any other icon font library --> ya
awijaya Un 106. <li class="nav-item"> ya
awijaya Un 107. <a href="#" class="nav-link"> ya
awijaya Un 108. <i class="nav-icon fas fa-tachometer-alt"></i> ya
awijaya Un 109. <p> ya
awijaya Un 110. Dashboard ya
awijaya Un 111. </p> ya
awijaya Un 112. </a> ya
awijaya Un 113. </ul> ya
awijaya Un 114. </nav> ya
awijaya Un 115. <!-- /.sidebar-menu --> ya
awijaya Un 116. </div> ya
awijaya Un 117. <!-- /.sidebar --> ya
awijaya Un 118. </aside> ya
awijaya Un 119. ya
awijaya Un 120. <!-- Content Wrapper. Contains page content --> ya
awijaya Un 121. <div class="content-wrapper"> ya
awijaya Un 122. <!-- Content Header (Page header) --> ya
awijaya Un 123. <section class="content-header"> ya
awijaya Un 124. <div class="container-fluid"> ya
awijaya Un 125. <div class="row mb-2"> ya
awijaya Un 126. <div class="col-sm-6"> ya
awijaya Un 127. <h1>Sistem Deteksi Pencuri dan Pelacakan Kendaraan Bermotor</h1> ya
awijaya Un 128. </div> ya
awijaya Un 129. <div class="col-sm-6"> ya
awijaya Un 130. <ol class="breadcrumb float-sm-right"> ya
awijaya Un 131. <li class="breadcrumb-item"><a href="#">Home</a></li> ya
awijaya Un 132. <li class="breadcrumb-item active">Blank Page</li> ya
awijaya Un 133. </ol> ya
awijaya Un 134. </div> ya
awijaya Un 135. </div> ya
awijaya Un 136. </div><!-- /.container-fluid --> ya
awijaya Un 137. </section> ya
awijaya Un 138. ya
awijaya Un 139. <!-- Main content --> ya
awijaya Un 140. <section class="content"> ya
awijaya Un 141. <h5 class="mb-2">Sensor</h5> ya
awijaya Un 142. <div class="row"> ya
awijaya Un 143. <div class="col-md-3 col-sm-6 col-12"> ya
awijaya Un 144. <div class="info-box"> ya
awijaya Un 145. <span class="info-box-icon bg-info"><i class="far fa- ya
awijaya Un map"></i></span> ya
awijaya Un 146. ya
awijaya Un 147. <div class="info-box-content"> ya
awijaya Un 148. <span class="info-box-text">Jarak Dari Titik Awal</span> ya
awijaya Un 149. <span class="info-box-number"><?php echo ya
awijaya Un number_format(distance($park_x, $park_y, $last_x, $last_y, "K")*100, 2); ?> ya
awijaya Un M</span> ya
awijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya
awijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya

```



```

150.         </div>
151.         <!-- /.info-box-content -->
152.     </div>
153. <!-- /.info-box -->
154. </div>
155. <!-- /.col -->
156. <div class="col-md-3 col-sm-6 col-12">
157.     <div class="info-box">
158.         <span class="info-box-icon bg-success"><i class="far fa-
user"></i></span>
159.
160.         <div class="info-box-content">
161.             <span class="info-box-text">Sensor Pengendara</span>
162.             <span class="info-box-number"><?php if($sensor == 0){ echo
"Aman"; }else if($sensor == 1){ echo "Waspada"; } else { echo "Bahaya"; } ?></span>
163.         </div>
164.         <!-- /.info-box-content -->
165.     </div>
166. <!-- /.info-box -->
167. </div>
168. <!-- /.col -->
169. <div class="col-md-3 col-sm-6 col-12">
170.     <div class="info-box">
171.         <span class="info-box-icon bg-warning"><i class="fas fa-
lock"></i></span>
172.
173.         <div class="info-box-content">
174.             <span class="info-box-text">Relay</span>
175.             <span class="info-box-number"><?php echo $relay == TRUE ? 'ON
(Circuit Open)':'Off (Circuit Close)'; ?></span>
176.         </div>
177.         <!-- /.info-box-content -->
178.     </div>
179. <!-- /.info-box -->
180. </div>
181. <!-- /.col -->
182. <div class="col-md-3 col-sm-6 col-12">
183.     <div class="info-box">
184.         <span class="info-box-icon bg-danger"><i class="fas fa-
signal"></i></span>
185.
186.         <div class="info-box-content">
187.             <span class="info-box-text">GPS Status</span>
188.             <span class="info-box-number"><?php echo $status_gps == TRUE ?
'Ok':'No GPS Connection'; ?></span>
189.         </div>
190.         <!-- /.info-box-content -->
191.     </div>
192. <!-- /.info-box -->
193. </div>
194. <!-- /.col -->
195. </div>
196. <!-- Default box -->
197. <div class="card">
198.     <div class="card-header">
199.         <h3 class="card-title">Command</h3>
200.
201.     <div class="card-tools">
202.         <button type="button" class="btn btn-tool" data-card-
widget="collapse" title="Collapse">
203.             <i class="fas fa-minus"></i>
204.         </button>
205.         <button type="button" class="btn btn-tool" data-card-widget="remove"
title="Remove">
206.             <i class="fas fa-times"></i>
207.         </button>

```



```

awijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya
awijaya Un 208.         </div> ya
awijaya Un 209.         </div> ya
awijaya Un 210.         <div class="card-body"> ya
awijaya Un 211.         <button type="button" id="trackBtn" class="btn btn-block bg-gradient- ya
awijaya Un primary">Track</button> ya
awijaya Un 212.         <button type="button" id="parkBtn" class="btn btn-block btn- ya
awijaya Un success">Park</button> ya
awijaya Un 213.         <button type="button" id="turnoffBtn" class="btn btn-block <?php echo ya
awijaya Un $relay == TRUE ? 'btn-info' : 'btn-danger'; ?>><?php echo $relay == TRUE ? 'Turn ya
awijaya Un Off Relay / Turn On Bike' : 'Turn On Relay / Turn Off the Bike'; ?></button> ya
awijaya Un 214.         </div> ya
awijaya Un 215.         <!-- /.card-body --> ya
awijaya Un 216.         <div class="card-footer"> ya
awijaya Un 217.         Last Received Data : <?php echo $last_send; ?> | Last Valid GPS Data : ya
awijaya Un <?php echo $last_valid; ?> | Last Command Received : <?php echo $last_command; ?> ya
awijaya Un 218.         </div> ya
awijaya Un 219.         <!-- /.card-footer--> ya
awijaya Un 220.         </div> ya
awijaya Un 221.         <!-- /.card --> ya
awijaya Un 222.         ya
awijaya Un 223.         </section> ya
awijaya Un 224.         <!-- /.content --> ya
awijaya Un 225.         </div> ya
awijaya Un 226.         <!-- /.content-wrapper --> ya
awijaya Un 227.         ya
awijaya Un 228.         <footer class="main-footer"> ya
awijaya Un 229.         <div class="float-right d-none d-sm-block"> ya
awijaya Un 230.         <b>Version</b> 1.0 ya
awijaya Un 231.         </div> ya
awijaya Un 232.         <strong>Copyright &copy; 2021 CDP TEUB. Theme by <a ya
awijaya Un href="https://adminlte.io">AdminLTE</a>.</strong> ya
awijaya Un 233.         </footer> ya
awijaya Un 234.         ya
awijaya Un 235.         <!-- Control Sidebar --> ya
awijaya Un 236.         <aside class="control-sidebar control-sidebar-dark"> ya
awijaya Un 237.         <!-- Control sidebar content goes here --> ya
awijaya Un 238.         </aside> ya
awijaya Un 239.         <!-- /.control-sidebar --> ya
awijaya Un 240.        </div> ya
awijaya Un 241.        <!-- ./wrapper --> ya
awijaya Un 242.         ya
awijaya Un 243.        <!-- jQuery --> ya
awijaya Un 244.        <script type="text/javascript"> ya
awijaya Un 245.         document.getElementById("trackBtn").onclick = function () { ya
awijaya Un 246.             location.href = "fetch.php"; ya
awijaya Un 247.         }; ya
awijaya Un 248.         document.getElementById("parkBtn").onclick = function () { ya
awijaya Un 249.             location.href = "park.php"; ya
awijaya Un 250.         }; ya
awijaya Un 251.         document.getElementById("turnoffBtn").onclick = function () { ya
awijaya Un 252.             location.href = "control.php"; ya
awijaya Un 253.         }; ya
awijaya Un 254.        </script> ya
awijaya Un 255.        <script src="raw/plugins/jquery/jquery.min.js"></script> ya
awijaya Un 256.        <script src="raw/plugins/bootstrap/js/bootstrap.bundle.min.js"></script> ya
awijaya Un 257.        <script src="raw/dist/js/adminlte.min.js"></script> ya
awijaya Un 258.        <script src="raw/dist/js/demo.js"></script> ya
awijaya Un 259.        <script src="notif.js"></script> ya
awijaya Un 260.        </body> ya
awijaya Un 261.        </html> ya
awijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya
awijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya
awijaya Un 1. <?php ya
awijaya Un 2. ya
awijaya Un 3. $servername = "localhost"; ya
awijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya
awijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya
awijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya Universitas Brawijaya

```



```
4. $username = "cdpteubm_motor";
5. $password = "Malang123@";
6. $dbname = "cdpteubm_motor";
7.
8. function isValidJSON($str) {
9.     json_decode($str);
10.    return json_last_error() == JSON_ERROR_NONE;
11. }
12.
13. function distance($lat1, $lon1, $lat2, $lon2, $unit) {
14.    if (($lat1 == $lat2) && ($lon1 == $lon2)) {
15.        return 0;
16.    }
17.    else {
18.        $theta = $lon1 - $lon2;
19.        $dist = sin(deg2rad($lat1)) * sin(deg2rad($lat2)) + cos(deg2rad($lat1)) *
cos(deg2rad($lat2)) * cos(deg2rad($theta));
20.        $dist = acos($dist);
21.        $dist = rad2deg($dist);
22.        $miles = $dist * 60 * 1.1515;
23.        $unit = strtoupper($unit);
24.
25.        if ($unit == "K") {
26.            return ($miles * 1.609344);
27.        } else if ($unit == "N") {
28.            return ($miles * 0.8684);
29.        } else {
30.            return $miles;
31.        }
32.    }
33. }
34.
35. $json_params = file_get_contents("php://input");
36.
37. if (strlen($json_params) > 0 && isValidJSON($json_params))
38. {
39.     $decoded_params = json_decode($json_params, true);
40.
41.     //print_r($decoded_params);
42.     $parsed = $decoded_params;
43.     //print_r($parsed);
44.     if (array_key_exists("pir", $parsed) && array_key_exists("x", $parsed) &&
array_key_exists("y", $parsed) && array_key_exists("gps", $parsed))
45.     {
46.         //echo $parsed[pir] . " " . $parsed[x] . " " . $parsed[y];
47.         //echo "OK";
48.         try {
49.             $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username,
$password);
50.             // set the PDO error mode to exception
51.             //var_dump($parsed);
52.
53.             $thex = $parsed["x"];
54.             $they = $parsed["y"];
55.             $sensor = $parsed["pir"];
56.             $gps_status = intval($parsed["gps"]) == 1 ? 1 : 0;
57.             if($gps_status == 0)
58.             {
59.                 $thex = 0;
60.                 $they = 0;
61.             }
62.             $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
63.             $sql = "INSERT INTO coord (x, y, gps_status)
VALUES ('" . strval($thex) . "', '" . strval($they) . "', " . $gps_status
64.             .")";
65.             $conn->exec($sql);
```



```

66.         $stmt = $conn->prepare("UPDATE state SET sensor = ". $sensor ." WHERE
device = '1337'");
67.         $stmt ->execute();
68.         $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username,
$password);
69.         $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
70.         $stmt = $conn->prepare("SELECT safe,relay,x,y FROM state WHERE device =
'1337' LIMIT 1");
71.         $stmt->execute();
72.
73.         $stmt->setFetchMode(PDO::FETCH_ASSOC);
74.         $state = $stmt->fetchAll();
75.         $send->safety = $state[0][safe];
76.         $send->relay = $state[0][relay];
77.         if((floatval(distance($thex,$they,$state[0][x],$state[0][y], "K")*100) >
floatval(15)) && $gps_status == 1)
78.         {
79.             $apiToken = "1797609426:AAGbReUKs0MDrcgR-mI8CUtAvguG84hT8Mw";
80.             $ch = curl_init();
81.             date_default_timezone_set("Asia/Bangkok");
82.             $t=time();
83.             curl_setopt($ch, CURLOPT_URL,
"https://api.telegram.org/bot".$apiToken."/sendMessage?chat_id=-
1001413117964&text=Warning%0AMotor berpindah sejauh 10km%0A". date("Y-m-d
H:i:s", $t). "%0A" .urlencode('<a href="http://cdp-teub-
2021.masuk.id/dashboard/">Panel</a>')."&parse_mode=HTML");
84.             curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
85.             $output = curl_exec($ch);
86.             curl_close($ch);
87.         }
88.         $sendJSON = json_encode ($send);
89.         echo $sendJSON;
90.     } catch(PDOException $e) {
91.         echo "<br>" . $e->getMessage();
92.     }
93. }
94. /*try {
95.     $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username,
$password);
96.     // set the PDO error mode to exception
97.     $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
98.     $sql = "INSERT INTO coord (x, y)
VALUES ('-7.941686', '112.642788')";
99.     // use exec() because no results are returned
100.    $conn->exec($sql);
101.    echo "New record created successfully";
102. } catch(PDOException $e) {
103.     echo $sql . "<br>" . $e->getMessage();
104. }*/
105. }
106.
107. }
?>

```




1.3 GPS performance

Parameter	Specification			
Receiver type	50 Channels GPS L1 frequency, C/A Code SBAS: WAAS, EGNOS, MSAS			
Time-To-First-Fix ¹		NEO-6G/Q/T	NEO-6MV	NEO-6P
	Cold Start ²	26 s	27 s	32 s
	Warm Start ²	26 s	27 s	32 s
	Hot Start ²	1 s	1 s	1 s
	Aided Starts ³	1 s	<3 s	<3 s
Sensitivity ⁴		NEO-6G/Q/T	NEO-6MV	NEO-6P
	Tracking & Navigation	-162 dBm	-161 dBm	-160 dBm
	Reacquisition ⁵	-160 dBm	-160 dBm	-160 dBm
	Cold Start (without aiding)	-148 dBm	-147 dBm	-146 dBm
	Hot Start	-157 dBm	-156 dBm	-155 dBm
Maximum Navigation update rate		NEO-6G/Q/M/T 5Hz	NEO-6PV 1 Hz	
Horizontal position accuracy ⁶	GPS	2.5 m		
	SBAS	2.0 m		
	SBAS + PPP ⁷	< 1 m (2D, R50) ⁸		
	SBAS + PPP ⁷	< 2 m (3D, R50) ⁸		
Configurable Timepulse frequency range		NEO-6G/Q/M/P/V 0.25 Hz to 1 kHz	NEO-6T 0.25 Hz to 10 MHz	
Accuracy for Timepulse signal	RMS	30 ns		
	99%	<60 ns		
	Granularity	21 ns		
	Compensated ⁹	15 ns		
Velocity accuracy ⁶		0.1m/s		
Heading accuracy ⁶		0.5 degrees		
Operational Limits	Dynamics	≤ 4 g		
	Altitude ¹⁰	50,000 m		
	Velocity ¹⁰	500 m/s		

Table 2: NEO-6 GPS performance



For more details contact the u-blox support representative nearest you to receive dedicated u-blox 6 Receiver Description Including Protocol Specification [3].

1.10 Precise Point Positioning

u-blox' industry proven PPP algorithm provides extremely high levels of position accuracy in static and slow moving applications, and makes the NEO-6P an ideal solution for a variety of high precision applications such as surveying, mapping, marine, agriculture or leisure activities.

Ionospheric corrections such as those received from local SBAS¹² geostationary satellites (WAAS, EGNOS, MSAS) or from GPS enable the highest positioning accuracy with the PPP algorithm. The maximum improvement of positioning accuracy is reached with PPP+SBAS and can only be expected in an environment with unobstructed sky view during a period in the order of minutes.



1.7 Precision Timing

1.7.1 Time mode

NEO-6T provides a special Time Mode to provide higher timing accuracy. The NEO-6T is designed for use with stationary antenna setups. The Time Mode features three different settings described in Table 3: Disabled, Survey-In and Fixed Mode. For optimal performance entering the position of the antenna (when known) is recommended as potential source of errors will be reduced.

Time Mode Settings	Description
Disabled	Standard PVT operation
Survey-In	The GPS receiver computes the average position over an extended time period until a predefined maximum standard deviation has been reached. Afterwards the receiver will be automatically set to Fixed Mode and the timing features will be activated.
Fixed Mode	In this mode, a fixed 3D position and known standard deviation is assumed and the timing features are activated. Fixed Mode can either be activated directly by feeding pre-defined position coordinates (ECEF - Earth Center Earth Fixed format) or by performing a Survey-In. In Fixed mode, the timing errors in the TIMEPULSE signal which otherwise result from positioning errors are eliminated. Single-satellite operation is supported. For details, please refer to the u-blox 6 Receiver Description including Protocol Specification [2].

Table 3: Time mode settings

1.7.2 Timepulse and frequency reference

NEO-6T comes with a timepulse output which can be configured from 0.25 Hz up to 10 MHz. The timepulse can either be used for time synchronization (i.e. 1 pulse per second) or as a reference frequency in the MHz range. A timepulse in the MHz range provides excellent long-term frequency accuracy and stability.

1.7.3 Time mark

NEO-6T can be used for precise time measurements with sub-microsecond resolution using the external interrupt (EXTINT0). Rising and falling edges of these signals are time-stamped to the GPS or UTC time and counted. The Time Mark functionality can be enabled with the UBXC-FGM2 message

For details, please refer to the u-blox 6 Receiver Description including Protocol Specification [2].

1.8 Raw data

Raw data output is supported at an update rate of 5 Hz on the NEO-6T and NEO-6P. The UBXC-RXM-RAW message includes carrier phase with half-cycle ambiguity resolved, code phase and Doppler measurements, which can be used in external applications that offer precision positioning, real-time kinematics (RTK) and attitude sensing.

1.9 Automotive Dead Reckoning

Automotive Dead Reckoning (ADR) is u-blox' industry proven off-the-shelf Dead Reckoning solution for tier-one automotive customers. u-blox' ADR solution combines GPS and sensor digital data using a tightly coupled Kalman filter. This improves position accuracy during periods of no or degraded GPS signal.

The NEO-6V provides ADR functionality over its software sensor interface. A variety of sensors (such as wheel ticks and gyroscope) are supported, with the sensor data received via UBXC messages from the application processor. This allows for easy integration and a simple hardware interface, lowering costs. By using digital sensor data available on the vehicle bus, hardware costs are minimized since no extra sensors are required for Dead Reckoning functionality. ADR is designed for simple integration and easy configuration of different sensor options (e.g. with or without gyroscope) and vehicle variants, and is completely self-calibrating.

				id	waktu	x	y	gps_status	
<input type="checkbox"/>	Edit	Copy	Delete	932	2021-03-31 18:22:46	0.0000000000	0.0000000000	0	
<input type="checkbox"/>	Edit	Copy	Delete	933	2021-04-01 00:24:07	-7.9284500000	112.6114000000	0	
<input type="checkbox"/>	Edit	Copy	Delete	934	2021-04-06 13:49:41	0.0000000000	0.0000000000	0	
<input type="checkbox"/>	Edit	Copy	Delete	935	2021-04-06 13:49:43	0.0000000000	0.0000000000	0	
<input type="checkbox"/>	Edit	Copy	Delete	936	2021-04-06 13:52:49	0.0000000000	0.0000000000	0	
<input type="checkbox"/>	Edit	Copy	Delete	937	2021-04-06 13:52:55	0.0000000000	0.0000000000	0	
<input type="checkbox"/>	Edit	Copy	Delete	938	2021-04-06 13:53:01	0.0000000000	0.0000000000	0	
<input type="checkbox"/>	Edit	Copy	Delete	939	2021-04-06 13:53:07	0.0000000000	0.0000000000	0	
<input type="checkbox"/>	Edit	Copy	Delete	940	2021-04-06 13:53:13	0.0000000000	0.0000000000	0	
<input type="checkbox"/>	Edit	Copy	Delete	941	2021-04-06 13:53:19	0.0000000000	0.0000000000	0	
<input type="checkbox"/>	Edit	Copy	Delete	942	2021-04-06 13:53:25	0.0000000000	0.0000000000	0	
<input type="checkbox"/>	Edit	Copy	Delete	943	2021-04-06 13:53:31	-7.9513200000	112.6126000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	944	2021-04-06 13:53:37	-7.9513180000	112.6126000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	945	2021-04-06 13:53:43	-7.9512990000	112.6127000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	946	2021-04-06 13:53:49	-7.9512780000	112.6127000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	947	2021-04-06 13:53:55	-7.9512730000	112.6127000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	948	2021-04-06 13:54:01	-7.9512550000	112.6126000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	949	2021-04-06 13:54:07	-7.9512510000	112.6126000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	950	2021-04-06 13:54:13	-7.9512180000	112.6126000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	951	2021-04-06 13:54:19	-7.9512140000	112.6126000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	952	2021-04-06 13:54:25	-7.9512140000	112.6126000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	953	2021-04-06 13:54:31	-7.9512160000	112.6126000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	954	2021-04-06 13:54:37	-7.9512160000	112.6126000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	955	2021-04-06 13:54:43	-7.9512320000	112.6126000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	956	2021-04-06 13:54:49	-7.9512490000	112.6126000000	1	

phpMyAdmin

Server: localhost:3306 Database: cdpteubm_motor Table: coord

Browse Structure SQL Search Insert Export Import Operations % Triggers

<< < 59 > >> Number of rows: 25 Filter rows: Search this table Sort by key: None

				id	waktu	x	y	gps_status	
<input type="checkbox"/>	Edit	Copy	Delete	1457	2021-04-23 13:24:29	-7.9523410000	112.6138000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1458	2021-04-23 13:24:35	-7.9523410000	112.6138000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1459	2021-04-23 13:24:41	-7.9520940000	112.6135000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1460	2021-04-23 13:24:47	-7.9520940000	112.6135000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1461	2021-04-23 13:24:54	-7.9521150000	112.6135000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1462	2021-04-23 13:25:00	-7.9521070000	112.6134000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1463	2021-04-23 13:25:05	-7.9521070000	112.6134000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1464	2021-04-23 13:25:11	-7.9522330000	112.6134000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1465	2021-04-23 13:25:17	-7.9521710000	112.6133000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1466	2021-04-23 13:25:23	-7.9521710000	112.6133000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1467	2021-04-23 13:25:29	-7.9518740000	112.6133000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1468	2021-04-23 13:25:35	-7.9518740000	112.6133000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1469	2021-04-23 13:25:41	-7.9515310000	112.6128000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1470	2021-04-23 13:25:50	-7.9515250000	112.6126000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1471	2021-04-23 13:25:54	-7.9514740000	112.6122000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1472	2021-04-23 13:26:03	-7.9514410000	112.6119000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1473	2021-04-23 13:26:08	-7.9516620000	112.6117000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1474	2021-04-23 13:26:12	-7.9516620000	112.6117000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1475	2021-04-23 13:26:18	-7.9522910000	112.6115000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1476	2021-04-23 13:26:24	-7.9524850000	112.6117000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1477	2021-04-23 13:26:30	-7.9527060000	112.6118000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1478	2021-04-23 13:26:36	-7.9527060000	112.6118000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1479	2021-04-23 13:26:42	-7.9529400000	112.6121000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1480	2021-04-23 13:26:48	-7.9529400000	112.6121000000	1	
<input type="checkbox"/>	Edit	Copy	Delete	1481	2021-04-23 13:26:54	-7.9530220000	112.6127000000	1	

