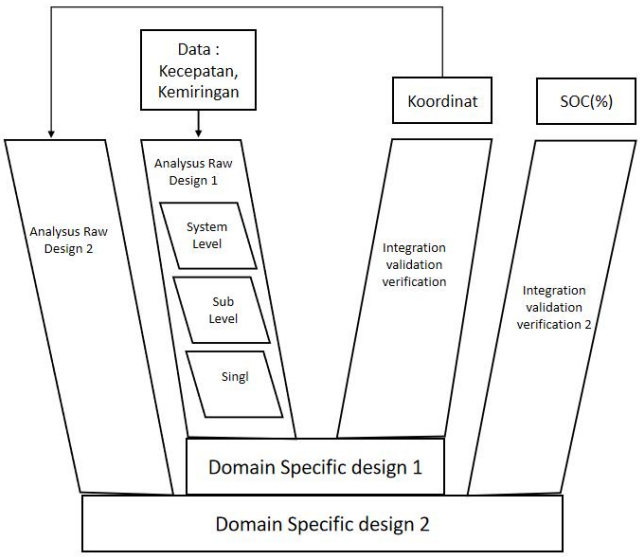
BAB III
METODE PENELITIAN

Metode penelitian mengenai perancangan dan implementasi DAQ berbasis *realtime operating* untuk masukan pemodelan dan simulasi konsumsi baterai kendaraan listrik. Berdasarkan landasan teori dari bab sebelumnya dan latar belakang dari tujuan penelian ini maka metode yang digunakan pada tugas akhir. Tahapan awal mengumpulkan sumber literatur yang terkait dengan pembahasan yang sedang diteliti. Setelah sumber sudah terkumpul maka dibuatkan untuk sistem pemodelan baik secara matematis maupun secara langsung melalui simulink pada matlab. Setelah sistem tergambar dengan rinci selanjutnya perancangan terkait kebutuhan untuk software dan integrasi pada peralatan. Kedua sistem sudah berjalan dengan baik maka melakukan kolaborasi antara kedua sistem tersebut. Setelah itu monioring sistem dan melakukan *trial and error.*

Apabila metodologi sistem diimplementasikan dengan menggunakan VD12206 makan akan menjadi seperti diawah ini.

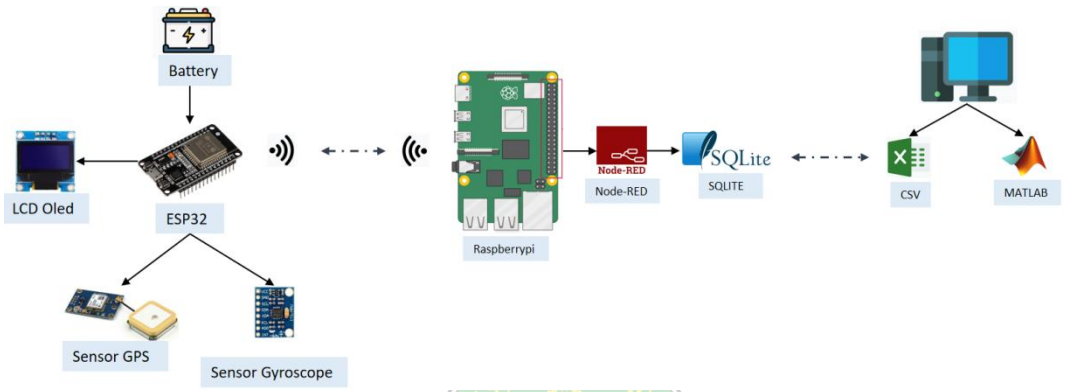


Gambar 3.2 Metodologi VDI2206

3.1 Arsitektur Umum Sistem Perancangan

Secara umum bagian utama arsitektur perancangan terbagi menjadi tiga bagian diantaranya bagian pengambilan data dengan mikrokontroler ESP32 sebagai

kendali untuk sistem. Lalu bagian akuisisi data pada bagian raspberrypi dengan menggunakan database SQLITE melaui sebuah aplikasi Node-RED. Setelah bagian pengambilan data melalui sensor dan data berhasil diambil oleh raspberrypi kedalam database maka data tersebut akan dijadikan sebagai masukan untuk sistem simulasi yang dijalankan pada *software* MATLAB. Simulasi pada MATLAB menggunakan metode simulink untuk mendapatkan pemodelan yang dapat divisualkan. Blok diagram untuk sistem ditunjukan pada gambar 3.xx.



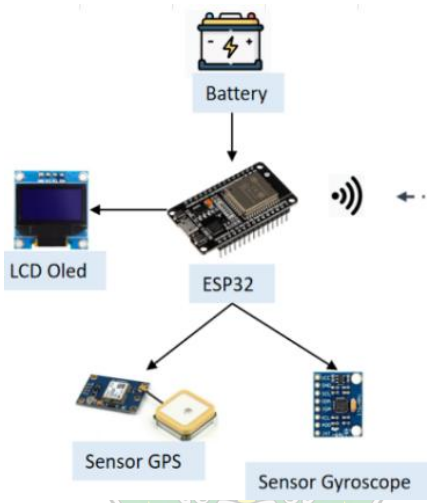
Gambar 3.3 Arsitektur umum sistem perancangan

Sistem yang dijalankan membutuhkan koneksi internet untuk dapat mengakses link MQTT Broker (broker.mqtt-dashboard.com) baik ESP32 maupun raspberrypi membutuhkan koneksi internet. Sensor yang digunakan hanya dapat berfungsi apabila berada diluar ruangan karena keterbatasan pada sensor GPS yang digunakan. Raspberrypi menjadi server utama ketika alat tersebut memberikan atau mempublish sebuah data. Data yang nantinya akan dikirimkan antara lain seperti :

1. Latitude
2. Longitude
3. Altitude
4. Kecepatan
5. Time
6. *Gradien* (Kemiringan jalan)
7. Akselerasi x,y,z
8. Rotasi x,y,z

3.1.1 Blok Pengirim Signal

Pada bagian blok pengirim signal menggunakan esp32 sebagai kontroler utama untuk mengatur segala dari perubahan yang terjadi pada sensor Gyroscope dan sensor GPS. Sensor dan kontroler mendapatkan tegangan sumber sebesar 5 Volt dari tegangan baterai luar menggunakan *powerbank.* Dengna menggunakan fitur wifi yang sudah tersedia secara buildin didalam esp32 mengirimkan data secara *realtime* kedalam server yang terhubung dengan internet. Dengna interface menggunakan LCD Oled sebagai indikator keberhasilan dari sensor yang mengirimkan data kedalam server.

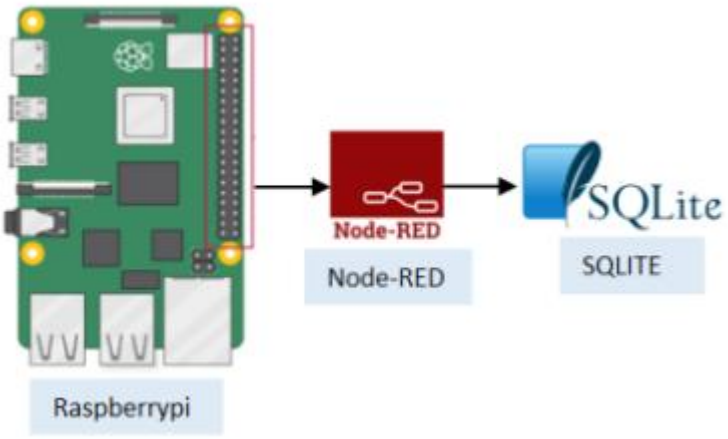


Gambar 3.4 Blok pengirim signal

Besasran-besaran tambahan lain masih dapat diambil oleh sensor dengan bantuan sebuah library pada mikrokontroler ESP32. Semua *sample* data yang diambil memiliki interval waktu masing-masing, tetapi setiap besaran memiliki momen yang sama ketika waktu pengambilan datanya. Metode pengambilan data dengan ESP32 dengan mengambil data dari sensor yang terdiri dari sensor *Gyro* untuk mendapatkan nilai kemiringan dari jalanan, sensor GPS Ublox NEO-6M untuk mendapatkan beberapa parameter koordinat untuk medapatkan posisi latitude dan longitude. Besaran ini tersebut yang nantinya akan digunakan untuk menentukan beberapa parameter. Sensor *gyro* digunakan untuk menentukan kemiringan dari jalan yang dilalui oleh kendaraan listrik.

3.1.2 Blok Penerima data

Pada bagian penerima data menggunakan raspberrypi untuk mengolah data yang sudah dikirimkan dengan protocol MQTT. Data yang dikirimkan diolah menggunakan node-red dan database sqlite sebagai penampung dari datanya. Masing - masing data yang diterima oleh raspi disimpan dalam table dan semua masuk dalam kolom yang sama baik sensor GPS maupun sensor Gyroscope.

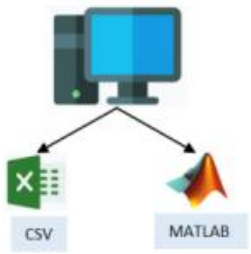


Gambar 3.4 Blok pengirim signal

Rasppberrypi sebagai kontrol untama mengolah data dengan bantuan aplikasi yang bernama Node-RED untuk program data akuisisinya dan pemilahan data apa saja yang akan dimasukan kedalam database SQLite.

3.1.3 Blok Simulasi Dengan MATLAB

Data yang sudah disimpan dalam database lalu dimasukan kedalam simulasi matlab dengan format dasar csv. Data yang dimasukan menghubungkan antara pengaruh dari jarak dan kemiringan yang ditempuh oleh kendaraan mobil listrik terhadap konsumsi energi listrik.



Data yang diperlukan untuk simulasi ini dikumpulkan dari berbagai sensor dan perangkat pengukuran yang dipasang pada kendaraan mobil listrik. Data ini mencakup informasi tentang jarak yang ditempuh dan kemiringan jalan yang dilalui. Setelah data dikumpulkan, data tersebut disimpan dalam database untuk memudahkan pengelolaan dan aksesibilitas. Format penyimpanan yang umum digunakan adalah CSV (Comma-Separated Values), karena format ini sederhana dan mudah diintegrasikan dengan berbagai aplikasi analisis data.

Setelah data disimpan dalam database, langkah selanjutnya adalah mengekspor data tersebut ke dalam format CSV. Proses ini melibatkan pemilihan data yang relevan dari database dan menyimpannya dalam file CSV. File CSV ini akan berisi kolom-kolom yang mewakili berbagai parameter seperti jarak, kemiringan, dan konsumsi energi listrik. Setiap baris dalam file CSV mewakili satu set data yang diukur pada waktu tertentu.

Dengan file CSV yang sudah siap, langkah berikutnya adalah mengimpor data tersebut ke dalam MATLAB. MATLAB memiliki fungsi bawaan seperti readtable atau csvread yang memungkinkan pengguna untuk membaca data dari file CSV dan menyimpannya dalam bentuk tabel atau array. Data yang diimpor ini kemudian dapat digunakan dalam berbagai analisis dan simulasi.

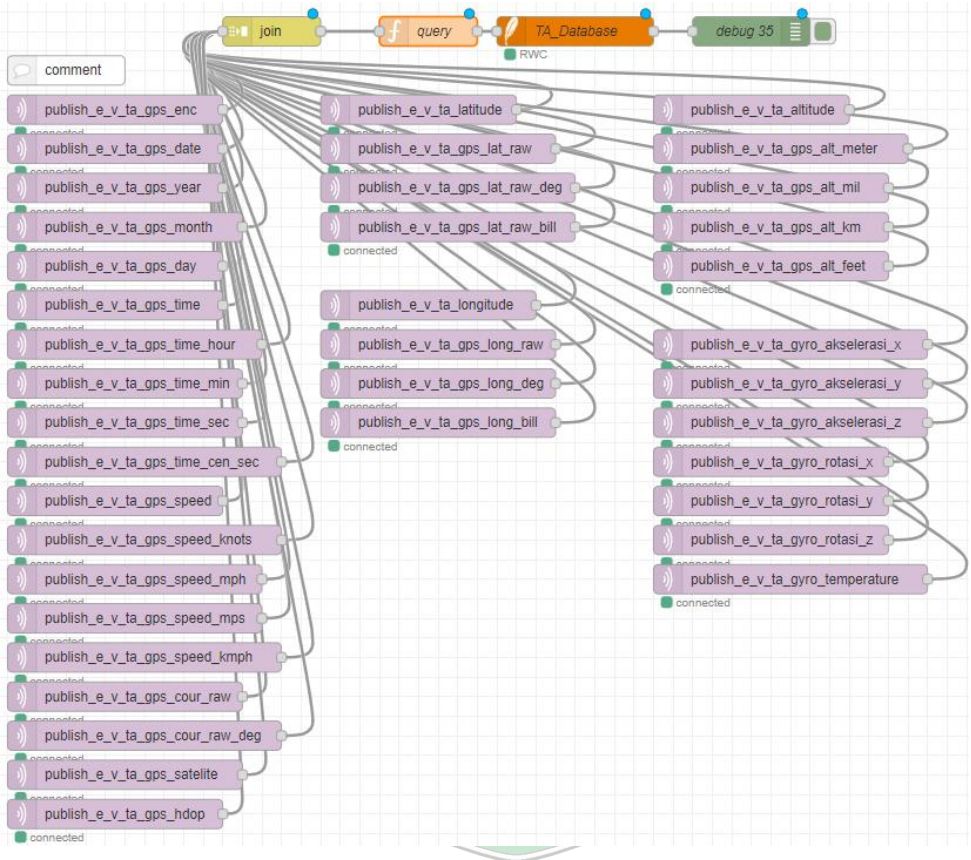
Dalam simulasi MATLAB, data yang diimpor digunakan untuk menganalisis bagaimana jarak dan kemiringan jalan mempengaruhi konsumsi energi listrik pada kendaraan mobil listrik. Simulasi ini dapat melibatkan pembuatan model matematis yang menggambarkan hubungan antara variabel-variabel tersebut. Misalnya, model dapat menunjukkan bahwa konsumsi energi meningkat seiring dengan peningkatan jarak dan kemiringan jalan.

Setelah simulasi dijalankan, hasilnya dianalisis untuk mendapatkan wawasan tentang efisiensi energi kendaraan. Analisis ini dapat mencakup pembuatan grafik yang menunjukkan hubungan antara jarak, kemiringan, dan konsumsi energi.

Hasil analisis ini dapat digunakan untuk mengoptimalkan desain kendaraan dan strategi pengemudian, sehingga konsumsi energi dapat diminimalkan.

3.2 Perancangan Software Akuisisi Data

3.2.1 Node-red

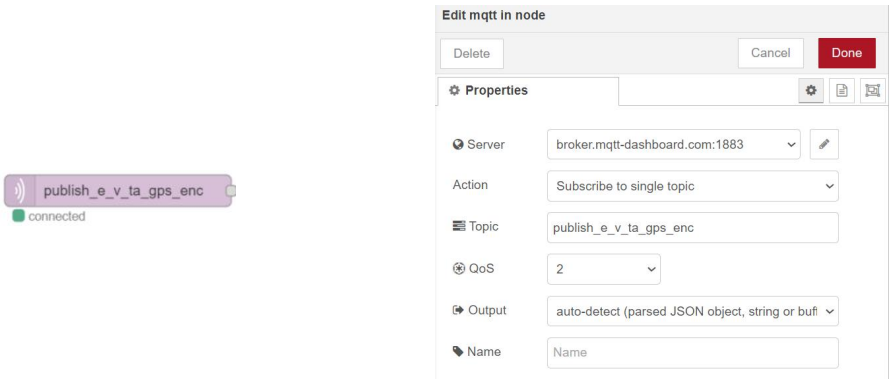


Gambar 3.5 *Node* akuisisi data Node-RED

Node dalam Node-RED memiliki keunikan dan fungsinya masing-masing. *Node* MQTT *broker* memiliki fungsi untuk menghubungkan beberapa sistem yang ada di plant kedalam sebuah server. Plant dalam hal ini bisa berupa *publisher* data yang dikirimkan oleh ESP32 atau *subscriber* yang dikendalikan oleh ESP32. *Node* join memiliki fungsi untuk mengumpulkan semua data dari MQTT *broker* menjadi satu dalam tipe data JSON(JavaScript Object Notation). Node SQLITE memiliki fungsi untuk mengolah JSON yang diberikan dan menjalankan query agar dapat memasukan data kedalam database.

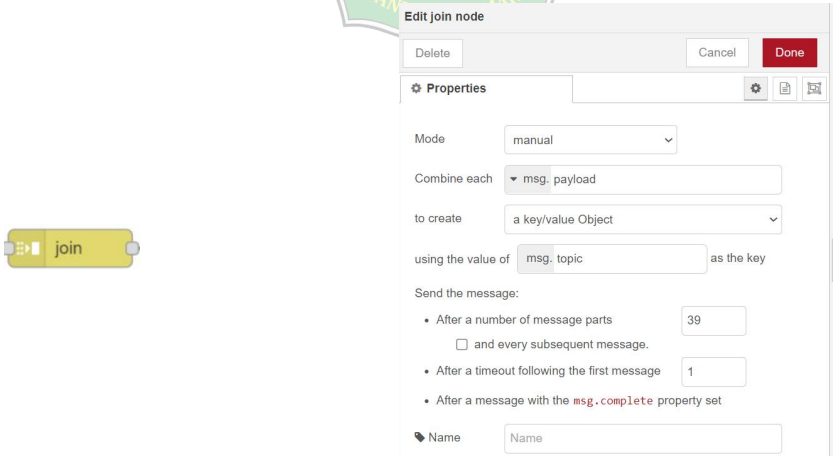
Gambar 3.6 Flow process dari MQTT sampai database

Node-RED diinstal di dalam Raspberrypi dan dijalankan didalam sebuah web browser. Untuk menjalankannya melalui sebuah perintah node-red didalam terminal lalu membuka nya didalam link localhost:1880.



Gambar 3.7 Flow process dari node MQTT

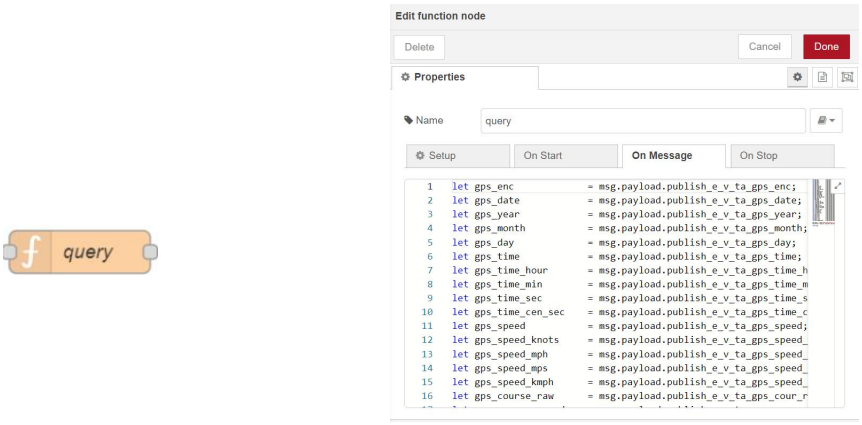
Proses pertama pada node MQTT dilakukan proses pembacaan untuk *subscribe* sebuah topic yang sudah di *publish* oleh ESP32 dan setiap topic memiliki penyimpanan yang unik agar tidak tertukar dengan proses *publish* yang lain. Karena node MQTT yang digunakan tidak hanya satu. Setiap node yang digunakan menampung *value* dari masing-masing besaran.



Gambar 3.8 Proses node Joint

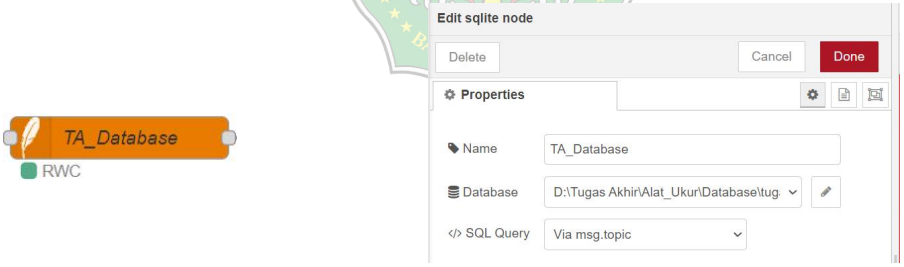
Node join berfungsi untuk menggabungkan *value* yang dimasukan sebagai inputnya. Input yang dapat dimasukan memiliki topic sebagai key untuk identitas dari masing-masing variabel. Variabel yang ada sudah didefinisikan sebelumnya

oleh mikrokontroler ESP32. Data hasil olahan node ini bisa berupa JSON atau sebuah objek untuk memudahkan ketika proses query.



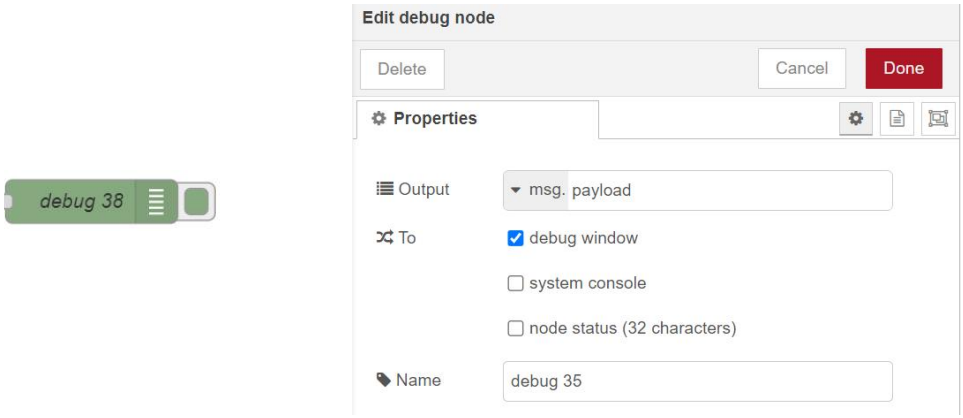
Gambar 3.9 Proses node function query

Node function memiliki fungsi untuk membuat sebuah sintak yang memiliki bahasa program Javasctipt. Bahasa ini tempat algoritma pemilahan dan query untuk memasukan nilai pada database. Proses masukan data pada database dilakukan dengan sebuah sintak pada node *function* tersebut. Dalam sintak mengikuti kaidan algoritma secara umum diantaranya ada bagian deklarasi, bagian *setup,* bagian body sintak.



Gambar 3.10 Proses node database SQLITE

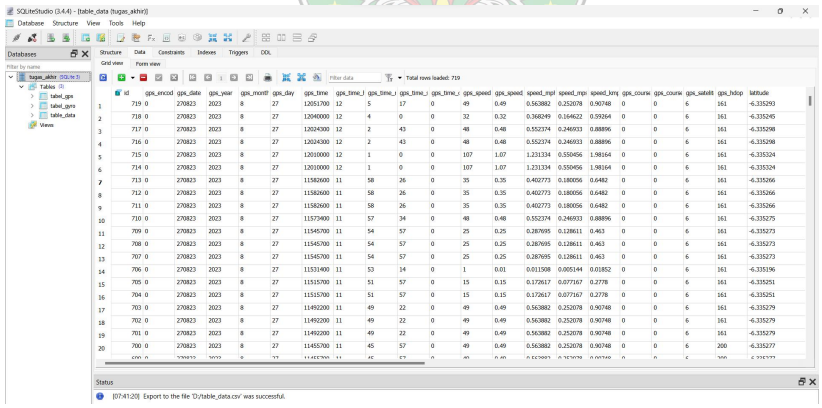
Node database SQLITE untuk menghubungkan antara sintak pada node function Node-RED dengan database. Dalam hal ini harus didefinisikan dimana letak folder database tersebut disimpan. Pada bagian database alamat yang disetting seperti (/home/pi/database/tugas_akhir.db) database SQLITE harus menempati sebuah folder yang menjadi acuan untuk *environment* pada pemrograman.



Gambar 3.11 Proses node debug

Bagian node debug berfungsi untuk menampilkan pesan dari msg.payload. Apabila ada error maka pesan ini akna ditampilkan sebagai indikator dan dimana lokasi node *error* tersebut berada. Sistem debug menampilkan data yang bisa dilihat pada jendela *sidebar.* Node debug ini memudahkan untuk melacak letak error dengan menyusuri sumber kesalahan nya, pesan yang ditmapilkan akan berwarna biru apabila pesan yang disampaikan benar, sedangkan pesan akan berwarna merah apabila pesan yang disampaikan keliru.

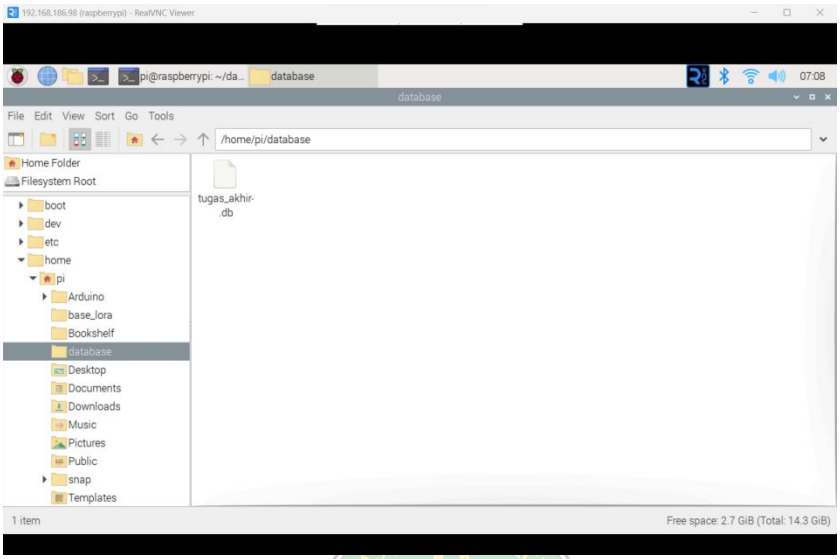
3.2.2 Databse SQLITE



Gambar 3.12 Database SQLite

Database SQLITE memiliki fungsi untuk menampung data yang diterima oleh MQTT *broker.* Beberapa data yang digunakan seperti kecepatan, longitude dan latitdu. Data yang diterima akan dimasukan kedalam sistem simulasi dan dilakukan pengolahan data untuk mecari hubungan antara konsumsi batrai terhadap kecepatan dan kemiringan jalan. CRUD(*Create Read Update Delete*)

dapat diimplementasikan untuk database SQLITE karena bahasa *query* yang digunakan sama. Variabel yang akan dimasukan kedalam sistem CRUD seperti latitude, longitude, altitude dari sensor GPS dan beberapa besaran akselerasi koordinat kartesius. Nilai dari setiap variabel tersebut akan tersimpan didalam folder raspi.

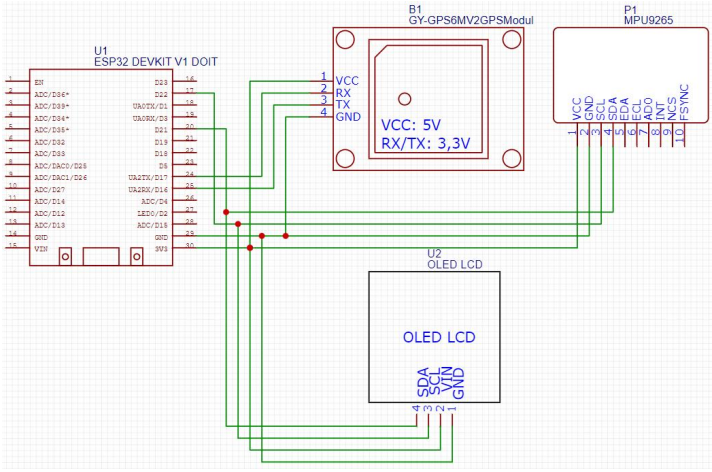


Gambar 3.13 lokasi penyimpanan database

Untuk dapat mengakses database yang sudah tersimpan harus menggunakan sebuah bahasa program yang dapat menjalankan instruksi *query leangue.* Semua sintak CRUD dijalankan menggunakan bahasa program JavaScript melalui Node-RED.

3.3 Perancangan Alat Akuisisi Data

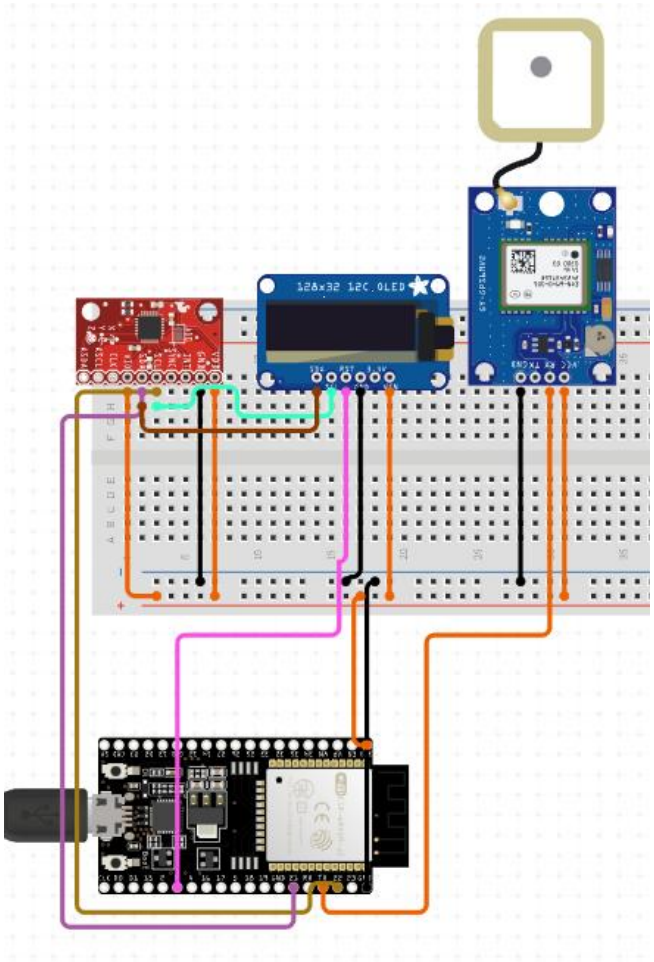
3.3.1 Skematik Rangkaian



Gambar 3.14 Skematik Rangkaian DAQ

Skematik berupa modul modul komponen elktronika yang sudah tertanam, sehingga dalam penggunaannya relatif lebih mudah karena sudah tersedia pin-pin untuk catu daya, ground, dan sinyal data. Perlu dibuatkan rangkaian skematik dengan tujuan membuat gambaran dan sketsa awal wiring untuk sistem elektrik. Setelah proses wiring skematik selesai maka selanjutnya tahapan pembuatan board PCB . PCB memudahkan dalam menyimpan dan menjaga ke stabilan sistem karena wiring yang sudah kokoh tidak ada jumper kabel untuk koneksinya. Sumber tegangan yang digunakan pada sistem skematik

3.3.2 Perancangan Hardware



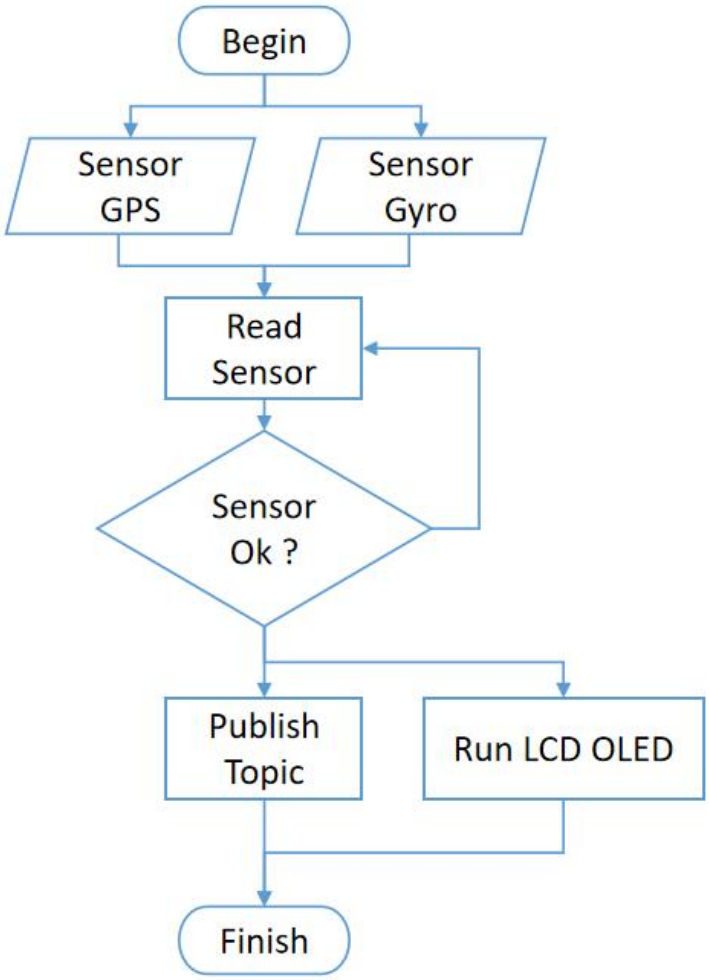
Gambar 3.15 Rancangan Komponen Hardware

Rangkaian elektrik yang dihubungkan satu sama lain dengan tampilan 2D untuk memudahkan pemahamana terkait metode pengambilan data melalui sensor GPS. Hardware yang digunakan meliputi mikrokontroler ESP32 dan beberapa sensor dengan satu interface yaitu LCD OLED. Senor Gyro menggunakan pin SDA, SCL untuk metode pengambilan data dengan mikrokontrolernya dan LCD OLED juga sama menggunakan SDA, SCL untuk dapat menampilkan grafik yang sudah diprogram seusai dengan kebutuhan. Sensor GPS menggunakan metode serial komunikasi dalam hal ini menggunakan pin RX dan pin TX.

3.3.4 Perancangan Sensor Gyroscope

Gambar 3.16 Perancangan Sensor Gyroscope

Dalam perancangan pengambilan datanya sensor Gyroscope menggunakan pin 5Vdc dan GND. Sedangkan dalam pengambilan datanya menggunakan pin RX TX yang dihubungkan dengan pin ESP32. Sensor *Gyroscope* memiliki fungsi untuk mendapatkan signal kemiringan dari kendaraan. Dengan bantuan sensor ini dapat memudahkan untuk mendapatkan signal kemiringan jalan. Diagram alir untuk sistem tersbut seperti ditunjukan pada gambar dibawah.

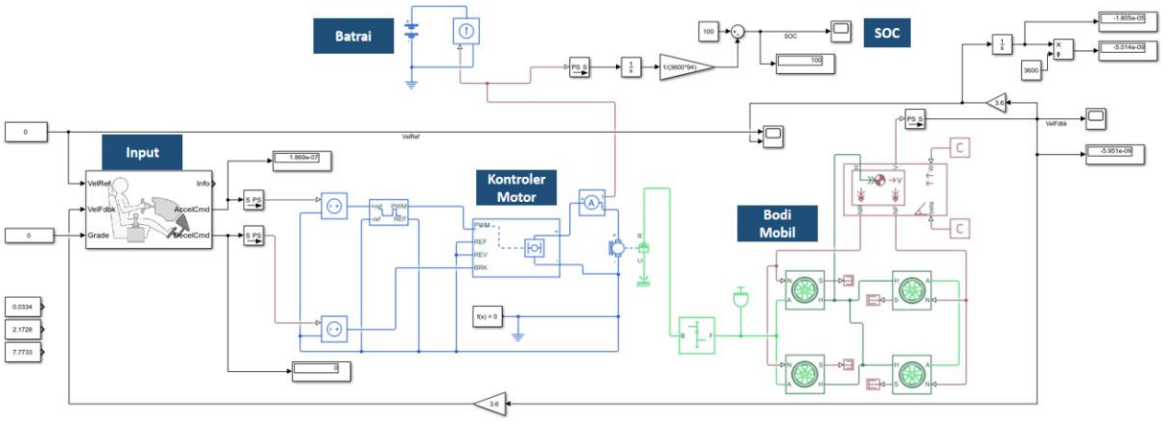


Gambar 3.16 Diagram alir pengambilan data

Peroses pertama ketika sistem berjakan akan membaca masukan dari sensor GPS dan sensor *gyroscope.* Sensor GPS dan *gyroscope* berjalan bersamaan dan ketika proses pembacaan berhasil maka sistem akan mempublish variabel-variabel yang telah di deklarasikan. Proses deklarasi bersamaan dengan ditampilkan nya besaran tersebut pada interface LCD OLED.

3.4 Perencanaan Simulasi Kendaraan Listrik

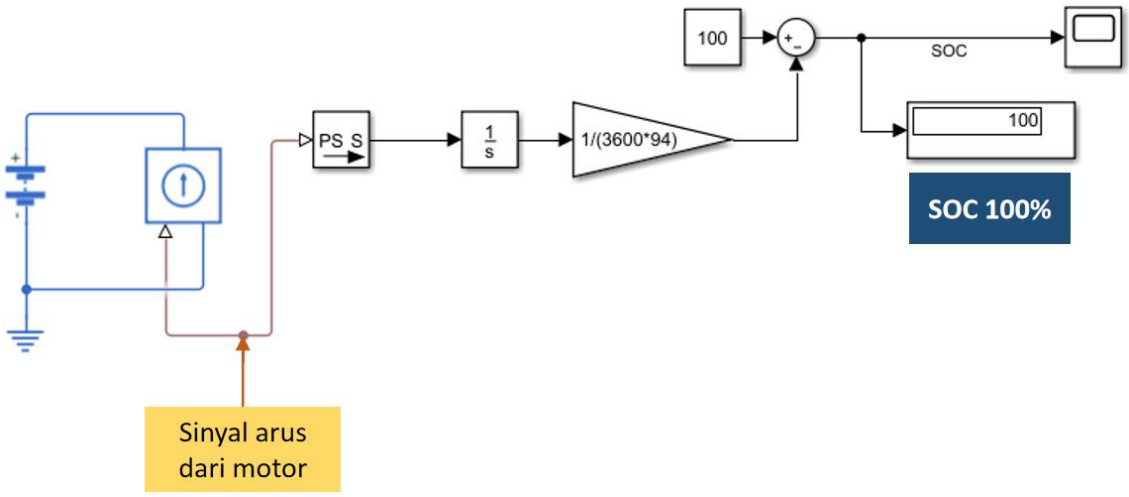
Simulasi kendaraan listrik menggunakan metode simulink dan terdiri dari beberapa bagian sub sistem seperti bagian sistem batrai, *longitudinal driver,* kontroler motor, bodi mobil dan bagian wheel. Keseluruhan sistem simulasi ditunjukan pada Gambar 3.17.



Gambar 3.17 Simulasi mobil listrik dengan simulink

Nilai masukan yang dijadikan parameter menggunakan blok constant dengan nilai yang sebelumnya sudah didapatkan pada proses DAQ. Semua sistem saling terhubung menjadi satu bagian untuk mendapatkan nilai SOC dengan inputan dari profile jalan.

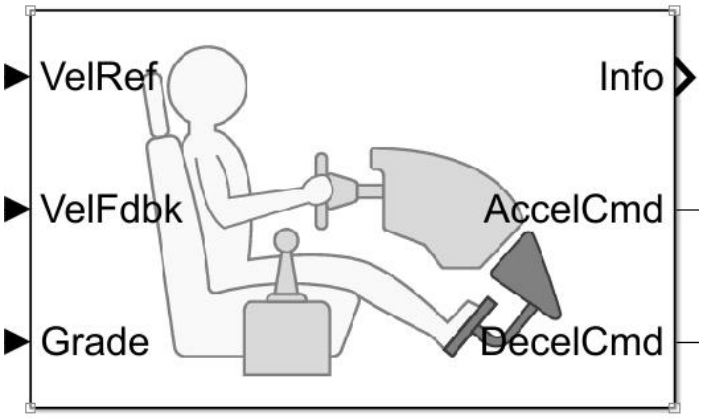
3.4.1 Perancangan Model Batrai



Gambar 3.18 Perancangan Model Batrai

Pada blok perancangan model batrai ketika beroperasi membutuhkan sinyal dari arus motor penggerak utama. Sinyal dari simulasi batrai harus dikonversi dari PS-Simulink agar dapat dilakukan proses aritmatika dan konversi nilai SOC. Sinyal arus akan melalui proses integrasi dan operasi aritmatika unutk mendapatkan nilai sebenarnya dari SOC. Dalam gambar di atas apabila sinyal arus yang dibawa oleh moto sebesar 0A maka nilaia dari SOC menjadi 100%. nilai SOC ketika 100% menunjukan kondisi batrai dalam keadaan penuh.

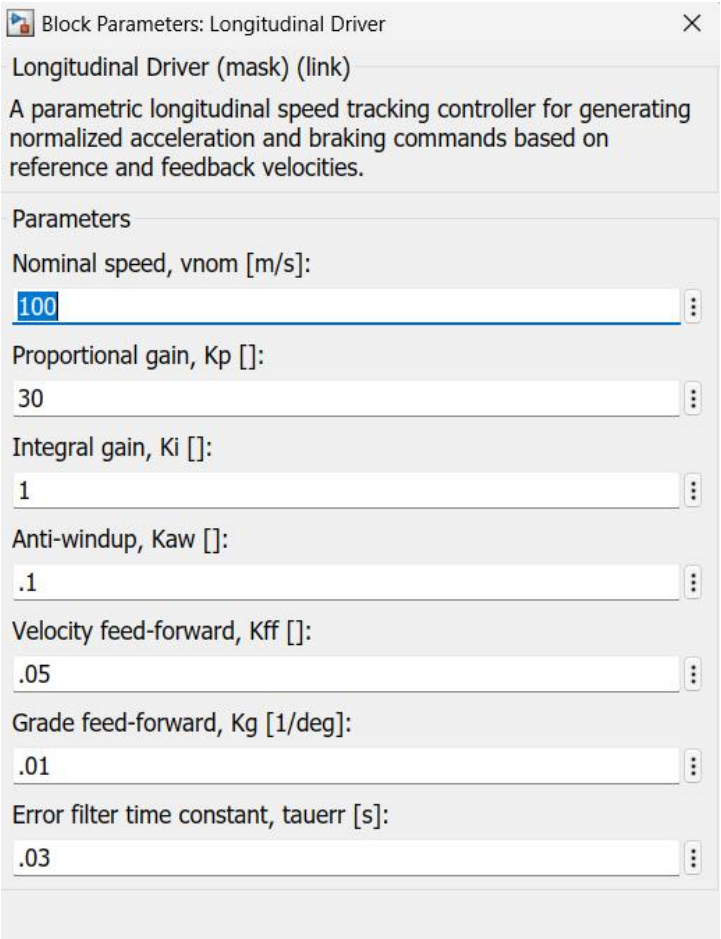
3.4.2 Perancangan Model Input



Gambar 3.19 Perancangan Model Input Longitudinal Driver

Parameter yang dijadikan inputan untuk simulasi kendaraan listrik terdiri dari 3 bagian diantaranya bagian VelRef untuk menentukan setpoint dari kecepatan yang

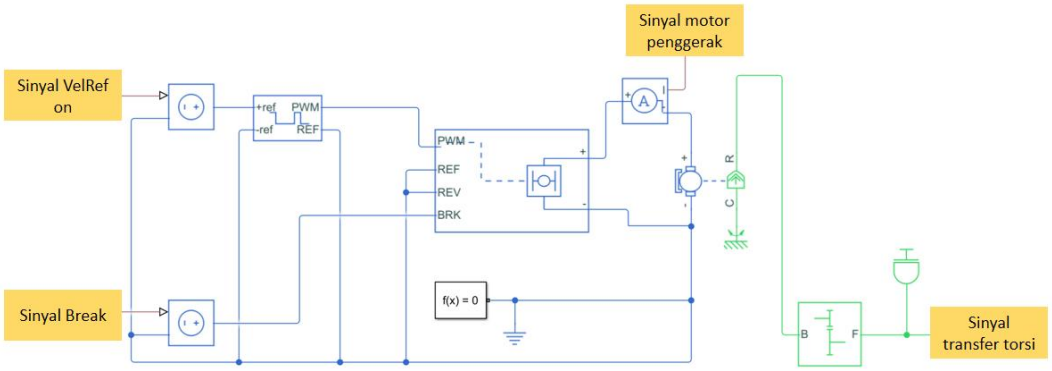
ditentukan. Variabel VelFdbk merupaakn variabel kecepatan dari umpan balik sistem untuk variabel pembanding nilai *error*. Kemringna jalan dinotasikan dengna Grade sebagai variabel yang menampung daata dari profile kemringan jalan dengan satuan derajat.



Gambar 3.20 Spesifik data longitudinal driver

Parameter kecepatan menggunakan satuan m/s karena dalam pengambilan data relatif tidak terlalu cepat, sehingga besaran kecepatan akan lebih mudah untuk dianalisa ketika menggunakan satuan m/s.

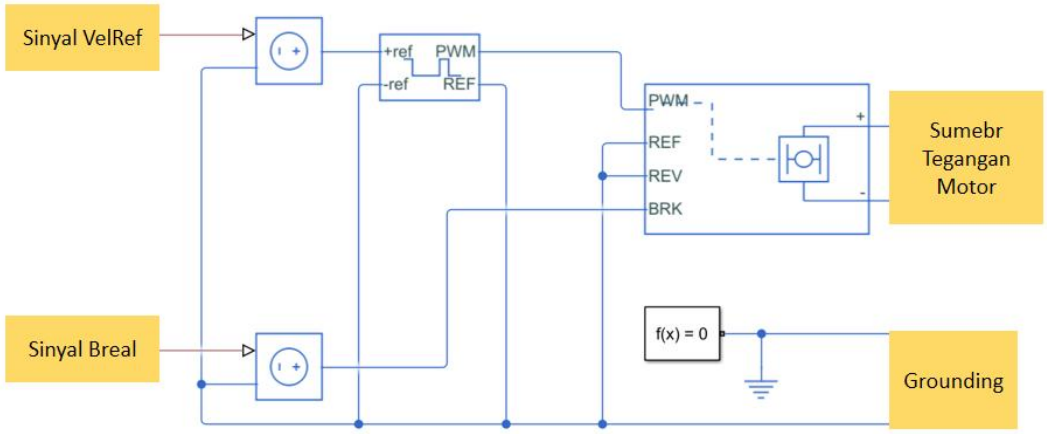
3.4.3 Perancangan Model Penggerak



Gambar 3.21 Model sistem penggerak

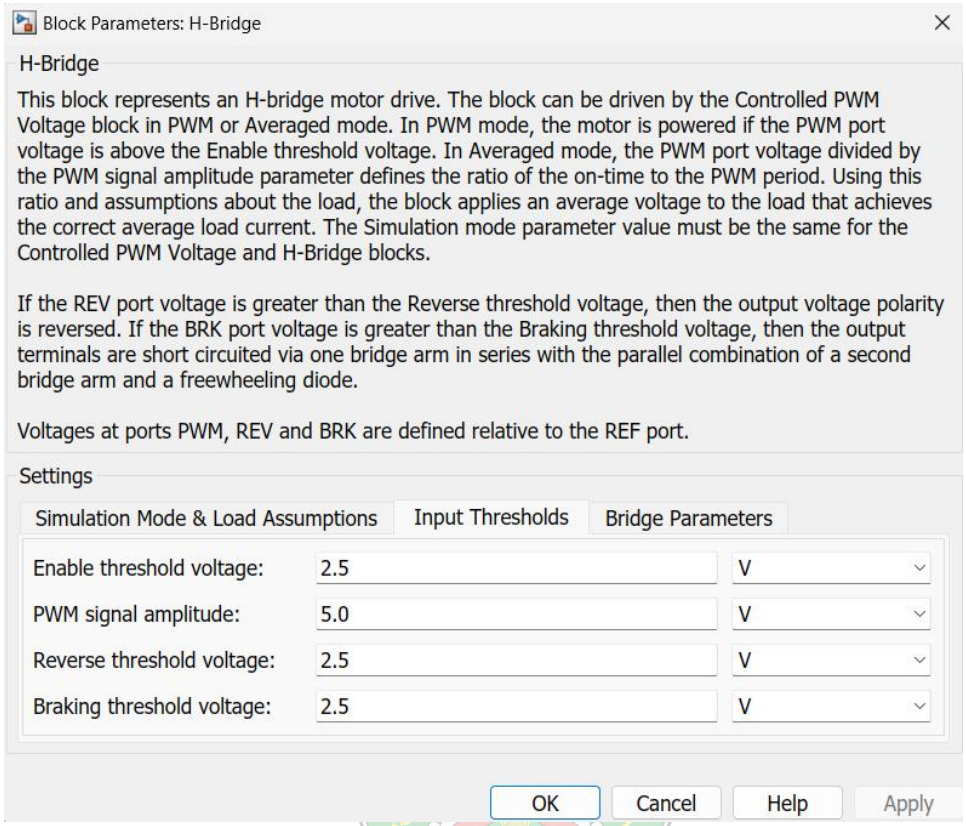
Tiga bagian utama dalam sistem simulasi ada pada bagian penerima sinyal dari input sebagai kondisi untuk PWM. Bagian kontroler utama *duty cycle* untuk kontrol kecepatan pada bagian blok PWM dan yang terakhir pada bagian motor untuk menggerakan roda mobi. Dalam transfer pergerakan menggunakan blok gear yang dihubungkan dengan motor DC dan roda kendaraan listrik.

3.4.4 Perancangan Model Kontroler Motor



Gambar 3.22 Model sistem penggerak

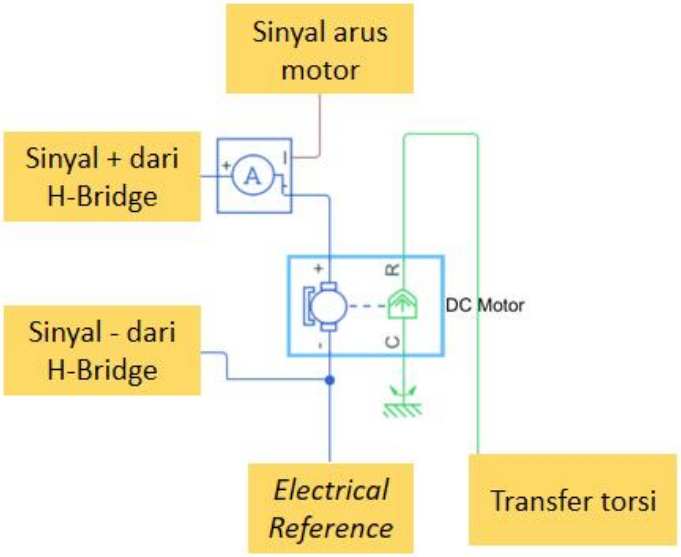
Sinyal VelRef memiliki dua kondisi yaitu kondisi on dan kondisi off. Masing-masing kondisi ditentukan dari nilai inptu yang sebelumnya diberikan, sehingga apabila nilai masukan bernilai logika atau off sistem kontroler motor tidak akan membukakan sumber tegangan untuk motor.



Gambar 3.23 Model sistem penggerak

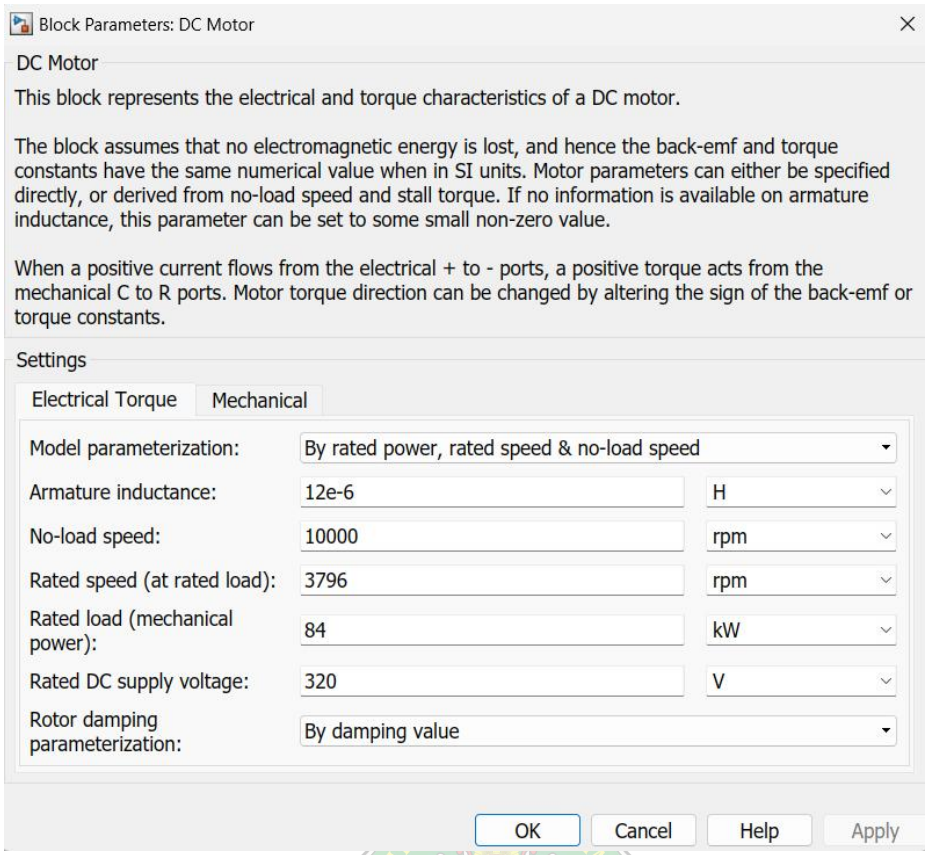
Sinyal H-Brigde beroperasi dalam tegangan 5V didalam kontrolernya. Sistem berjalan dengan menggunakna prinsip pengkondisian switch. Dengan menggunakan blok H-Brigde memungkinkan motor memiliki kondisi dipercepat, dierlambat, dan berhenti. Module signal H-bridge sebagai kontroler utama pada motor berfungsi untuk mengatur segala kondsi dari roda mobil dalam hal ini dapat mengtur kecepatan putaran dan arah putaran dari mobil. Switch yang mengatur perubahan dari putaran roda mobil berjalan kearah kiri secara terus menerus atau mobil berjalan kearah kana secara terus menerus.

3.4.5 Perancangan Model dan Motor



Gambar 3.24 Model sistem penggerak

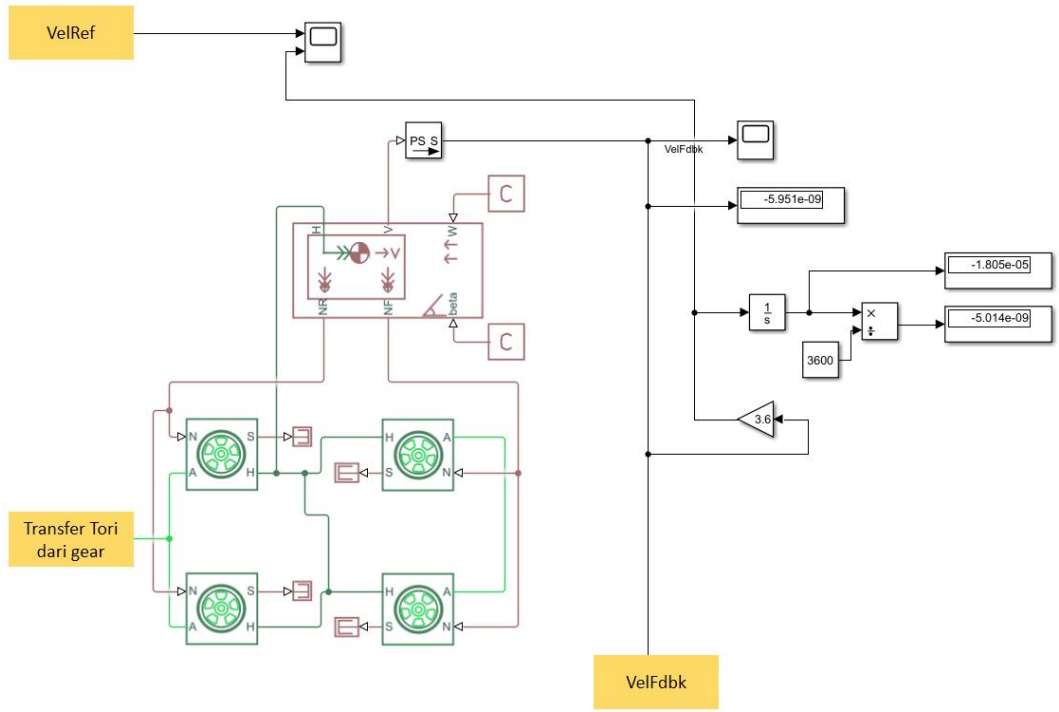
Motor DC memberikan kecepetan RPM sebesar 10.000 rpm kecepatan maksimalny dengan rentang tegangan suplai sebesar 320 V. Spesifikasi tersebut berdasarkan pada sebuah mobil dengan penggerak 4 roda. Lalu rated DC supply voltage sebesar 320 Volt untuk mendapatkan torsi roda yang besar.



Gambar 3.25 Model sistem penggerak spesifikasi

Area blok ini mendapatkan signal dari H-Bridge unuk memberkan arus pada motor roda mobil. Terjadi perubahan energi dari energi listrik menjadi energi mekanik dalam kasus ini rotasi roda moil.

3.4.6 Perancangan Body dan Wheel



Gambar 3.5 Model sistem bodi dan roda

Sistem transfer yang sebelumnya diolah oleh motor DC dijadikan masukan untuk keempat roda yang terhubung dengan bodi mobil. Pengaruh dari berat bodi mobil dan roda menjadi pengaruh terhadap konsumsi baterai. Blok bodi mesin memberikan output kecepatan umpan balik untuk menjadi parameter input pembanding dengan setpoint. Area transfer torsi dari gear pada roda masing-masing keempat bagian memberikan signal untuk seberapa besar torsi yang harus diberikan. Masing-masing roda akan berputar sesuai dengan kondisi yang diperintahkan.