

### BAB III

#### METODE PENELITIAN

Kajian yang dilakukan dalam penelitian ini bersifat aplikatif, yaitu perencanaan dan perealisasi alat agar dapat bekerja sesuai dengan yang direncanakan dengan mengacu pada rumusan masalah. Langkah-langkah yang perlu dilakukan untuk merealisasikan alat yang dirancang adalah penentuan spesifikasi alat, studi literatur, perancangan sistem, pembuatan alat, pengujian alat, dan pengambilan kesimpulan.

##### 3.1 Penentuan Spesifikasi Alat

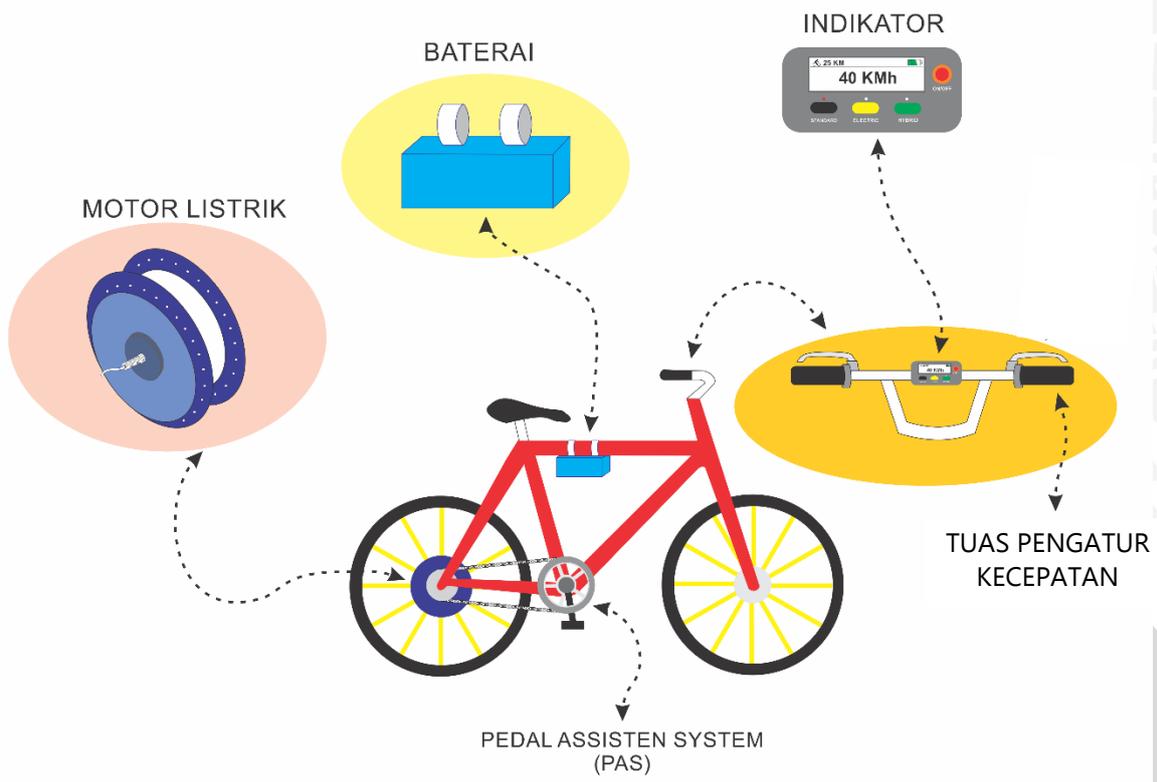
Spesifikasi alat secara global ditetapkan terlebih dahulu sebagai acuan dalam perancangan selanjutnya. Spesifikasi alat yang direncanakan yaitu:

- 1) Dapat mengukur arus hingga 20A.
- 2) Diakses melalui *smarthphone* Android via *Bluetooth*.
- 3) Komunikasi alat dengan *smarthphone* tanpa kabel.
- 4) Catu daya berasal dari baterai sebesar 36V.
- 5) Catu daya sistem sebesar 5V.

##### 3.2 Perancangan Alat

Perancangan dan pembuatan alat dalam penelitian ini dibagi menjadi dua bagian, yaitu *hardware* dan *software*. Untuk sepeda listriknya sendiri, memiliki beberapa bagian utama (ditunjukkan dalam Gambar 3.1) seperti :

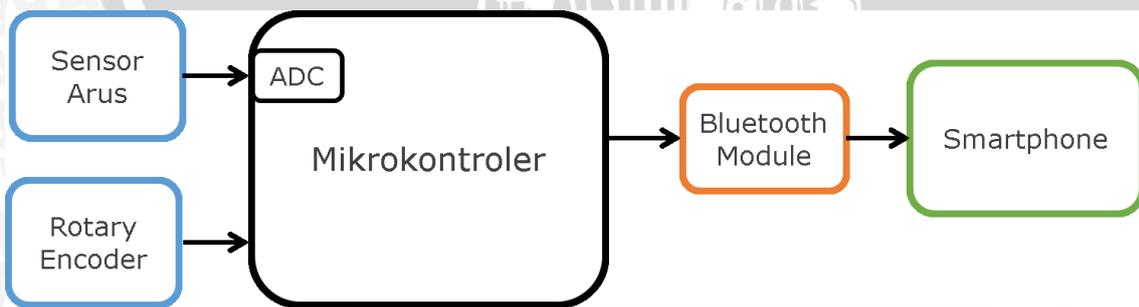
- Baterai : Sebagai sumber energi listrik. Jenis baterai yang digunakan adalah baterai lithium-ion yang dapat diisi ulang.
- Motor listrik : Sebagai penggerak roda sepeda.
- *Controller* : Mengatur *supply* listrik ke motor listrik agar dapat bekerja.
- Tuas pengatur kecepatan : Berfungsi untuk mengatur kecepatan motor listrik.
- Indikator : Indikator ini berupa *smarthphone* yang digunakan untuk menampilkan informasi kondisi sepeda listrik.



Gambar 3.1 Bagian utama sepeda listrik

### 3.2.1 Diagram Blok

Pembuatan diagram blok merupakan dasar dari perancangan sistem agar sistem dapat digambarkan dengan jelas dan sederhana. Diagram blok secara umum akan ditunjukkan dalam Gambar 3.2.



Gambar 3.2 Diagram blok keseluruhan sistem

Diagram blok pada Gambar 3.2 memaparkan keseluruhan sistem yang ada dalam penelitian ini.

1. Baterai berfungsi sebagai penyimpan daya dan sebagai catu daya terhadap beban.
2. Sensor arus berfungsi untuk mendeteksi arus yang mengalir melalui baterai, sehingga setiap perubahan nilai arus dapat diamati.
3. *Rotary encoder* bertugas sebagai pendeteksi perputaran roda, yang kemudian digunakan oleh mikrokontroler untuk menghitung kecepatan sepeda.
4. Data hasil pembacaan sensor arus dan *rotary encoder* akan diolah oleh mikrokontroler dan dikonversi menjadi kapasitas baterai dan prediksi jarak tempuh. Kemudian dikirim menuju display (*smartphone*) master melalui modul *bluetooth* HC-05.
5. *Smartphone* digunakan untuk menampilkan kecepatan dan prediksi jarak tempuh sepeda listrik.

### 3.2.2 Perancangan Perangkat Keras

Perancangan perangkat keras alat perekam penggunaan daya listrik terbagi menjadi beberapa bagian, antara lain:

1. Perancangan sistem minimum ATmega328p.
2. Perancangan *power supply*.

- **Perancangan sistem minimum ATmega328p**

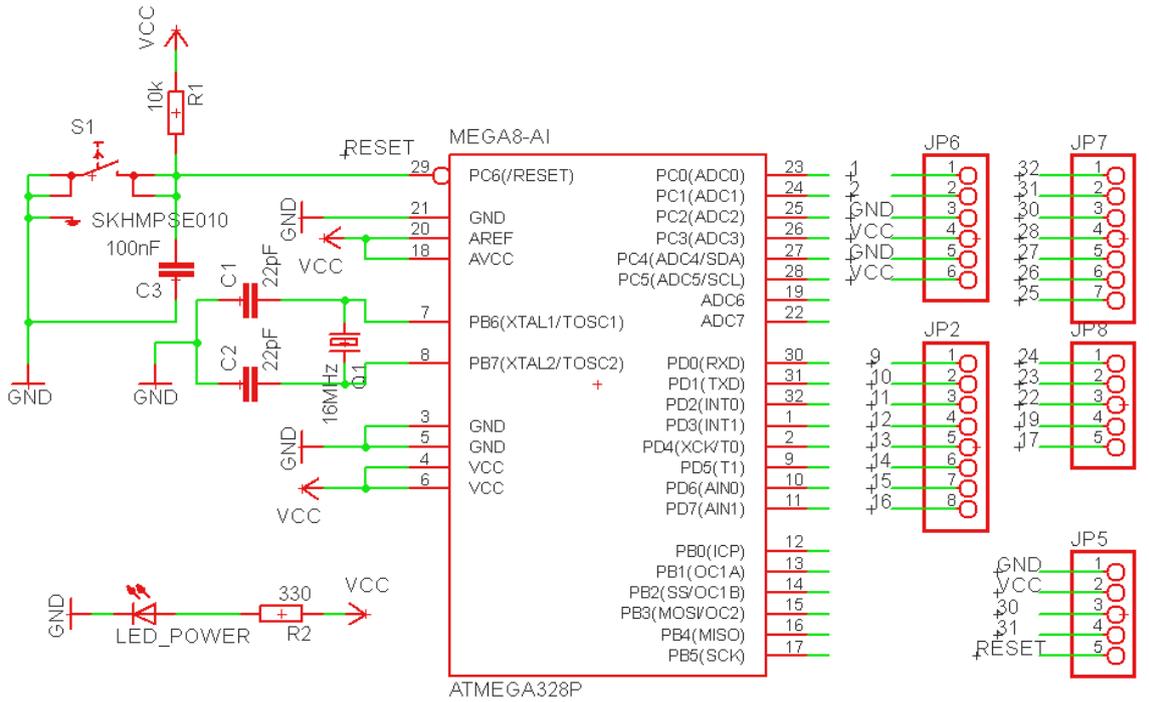
Untuk menggunakan mikrokontroler ATmega328p, diperlukan tambahan rangkaian sistem minimum seperti pada Gambar 3.3. Yang perlu diperhatikan berikutnya adalah koneksi pin ke perangkat lain. Pin ATmega328p yang dipakai yaitu :

Pin ADC0 = Dihubungkan dengan pin keluaran sensor arus

Pin INT0 = Dihubungkan ke keluaran *rotary encoder*

Pin RXD = Dihubungkan ke pin TX modul *bluetooth*

Pin TXD = Dihubungkan ke pin RX modul *bluetooth*

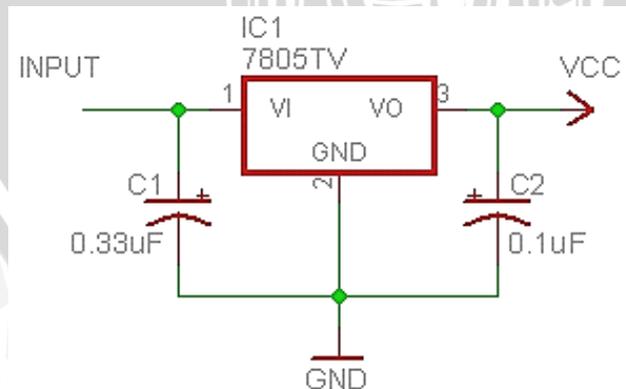


Gambar 3.3 Rangkaian sistem minimum ATmega328p

- **Perancangan power supply**

Salah satu perangkat penting yang bertugas memberi supply listrik ke perangkat lain, *power supply*. Perangkat-perangkat yang ada di sistem ini membutuhkan *supply* tegangan +5V, sehingga *power supply* harus mampu memberikan tegangan sebesar itu.

*Power supply* menggunakan sumber listrik dari baterai +40V yang diturunkan tegangannya hingga mencapai +9V, kemudian diturunkan lagi oleh regulator LM7805 menjadi +5V. Rangkaian power supply ditunjukkan dalam Gambar 3.4.



Gambar 3.4 Rangkaian power supply  
Sumber : Fairchild, 2014

### 3.2.3 Perancangan Perangkat Lunak

Perancangan perangkat lunak dibutuhkan untuk mengendalikan perangkat keras. Perancangan perangkat lunak dilakukan dengan pembuatan *flowchart* sistem secara keseluruhan dan pembuatan *flowchart* untuk fungsi-fungsi yang dibutuhkan. Bahasa pemrograman yang dipakai mikrokontroler adalah bahasa pemrograman C dengan menggunakan *software* Arduino. Sedangkan untuk aplikasi *smartphone* Android adalah Basic4Android.

- **Pengiriman data**

Setelah diolah oleh mikrokontroler, data prediksi jarak, kecepatan, jarak tempuh, dan kapasitas baterai akan dikirim ke *smartphone* melalui koneksi *bluetooth* menggunakan komunikasi serial.

Semua data tersebut digabung menjadi satu yang nantinya disebut sebagai “paket data”. Paket data memiliki format yang menyesuaikan aplikasi *smartphone* untuk menghindari kesalahan penerimaan data. Format yang dimaksud adalah :



Keterangan :

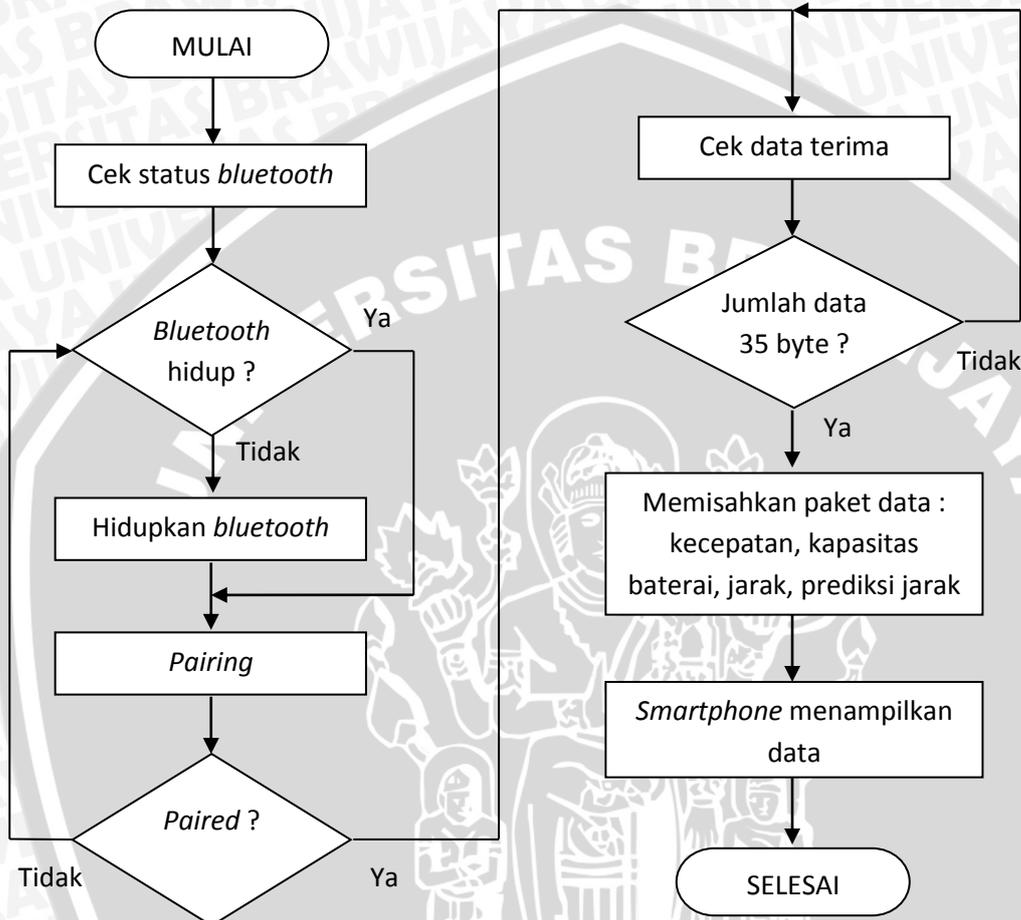
- Jumlah data = Besarnya bagian isi paket data dalam satuan byte.
- Isi = Berisi informasi yang akan ditampilkan *smartphone*.
- “:” = Sebagai pemisah antar data (*delimiter*).

Pada sistem ini, masing – masing data berjumlah 8 byte ditambah 3 byte pemisah ( : ), sehingga total isi paket data adalah 35 byte.

Dengan adanya bagian awalan, isi paket data dapat dicocokkan jumlahnya. Jika jumlah data dan awalan cocok maka paket data dapat diterima, dan akan ditolak kalau tidak cocok.

- **Aplikasi *smartphone***

Aplikasi *smartphone* berfungsi mengolah data yang diterima dari mikrokontroler dan menampilkannya. Gambar 3.5 menunjukkan proses kerja pengolahan data aplikasi *smartphone*.



Gambar 3.5 Diagram alir pengolahan data pada *smartphone*

- **Perhitungan Kapasitas Baterai**

Baterai yang digunakan pada sepeda listrik ini adalah baterai Lithium-Ion yang bisa diisi ulang. Untuk mengetahui kapasitas baterai, arus yang keluar masuk baterai perlu dihitung jumlahnya. Berikut merupakan perhitungan kapasitas baterai :

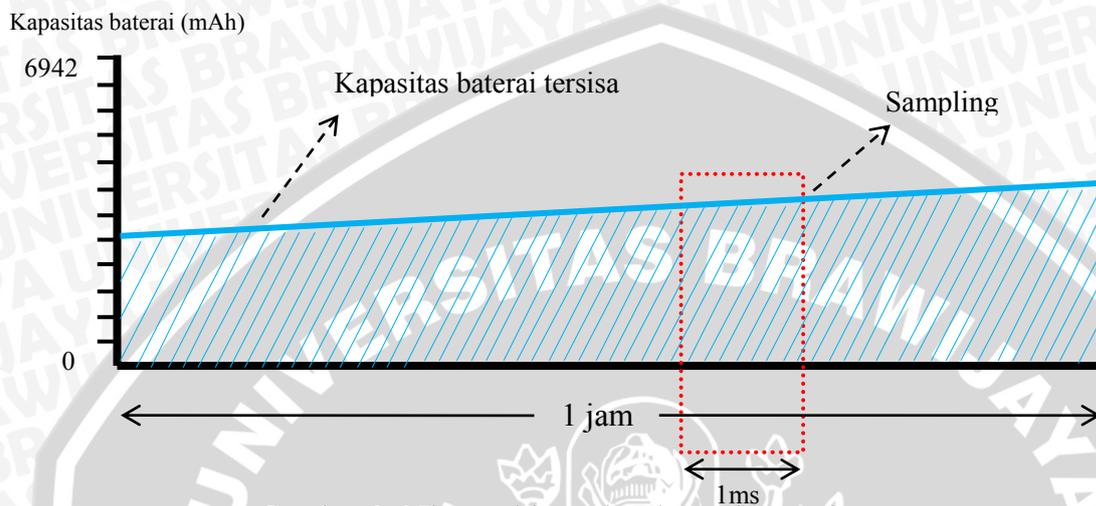
Saat proses *discharging* :

$$\text{Kapasitas baterai} = \text{Kapasitas baterai tersisa} - \text{Arus discharge}$$

Saat proses *charging* :

$$\text{Kapasitas baterai} = \text{Kapasitas baterai tersisa} + \text{Arus charge}$$

Baterai Li-Ion yang digunakan memiliki kapasitas maksimal 6942 mAh. Kapasitas tersebut disimpan di dalam *memory* EEPROM mikrokontroler. Kemudian jumlah arus didapat dari hasil pengolahan data sensor arus ACS712. Jumlah arus yang mengalir merupakan akumulasi besarnya arus yang mengalir selama waktu tertentu. Gambar 3.6 menunjukkan ilustrasi kapasitas baterai.



Gambar 3.6 Ilustrasi kapasitas baterai saat *charging*

Jumlah arus *charge* dan *discharge* yang memiliki satuan mAh dihitung dengan menjumlahkan arus yang mengalir dalam 1 jam. Namun jika harus menunggu selama 1 jam untuk menghitung jumlah arus, hasil pengukuran pun akan dihasilkan setiap 1 jam sekali. Hal tersebut tidak cocok untuk sistem yang membutuhkan informasi *real time*. Maka untuk mengatasinya, arus yang mengalir tersebut dibaca secara berkala oleh mikrokontroler setiap 1ms (*sampling*) kemudian dikonversi ke satuan mAh seperti berikut :

$$C = I \times t_h$$

$$C = I \times t_{ms}$$

$$C = I \times \frac{t_{ms}}{3.600.000}$$

$$\text{dimana } 1ms = \frac{1}{3.600.000} h$$

keterangan :

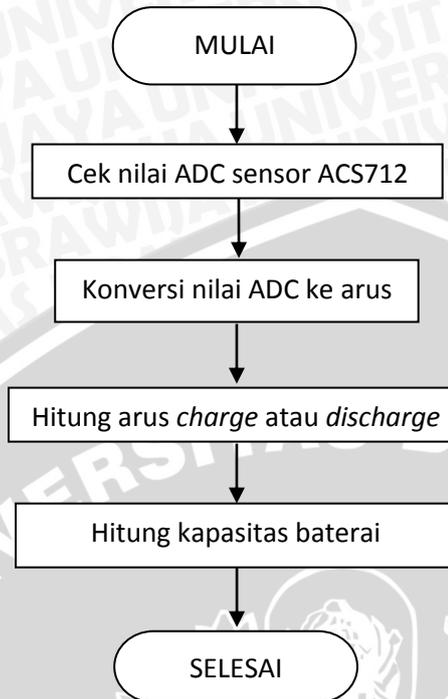
C = Kapasitas baterai (mAh)

I = Arus (mA)

$t_h$  = Waktu (hour)

$t_{ms}$  = Waktu (millisecond)

Proses perhitungan kapasitas baterai diperlihatkan dalam Gambar 3.7.



Gambar 3.7 Diagram alir perhitungan kapasitas baterai

- **Perhitungan jarak tempuh**

Hasil pengolahan data keluaran sensor *rotary encoder* yang berupa jumlah putaran roda digunakan untuk menghitung jarak tempuh menggunakan hasil perhitungan berikut :

$$s = R \times \text{Keliling}$$

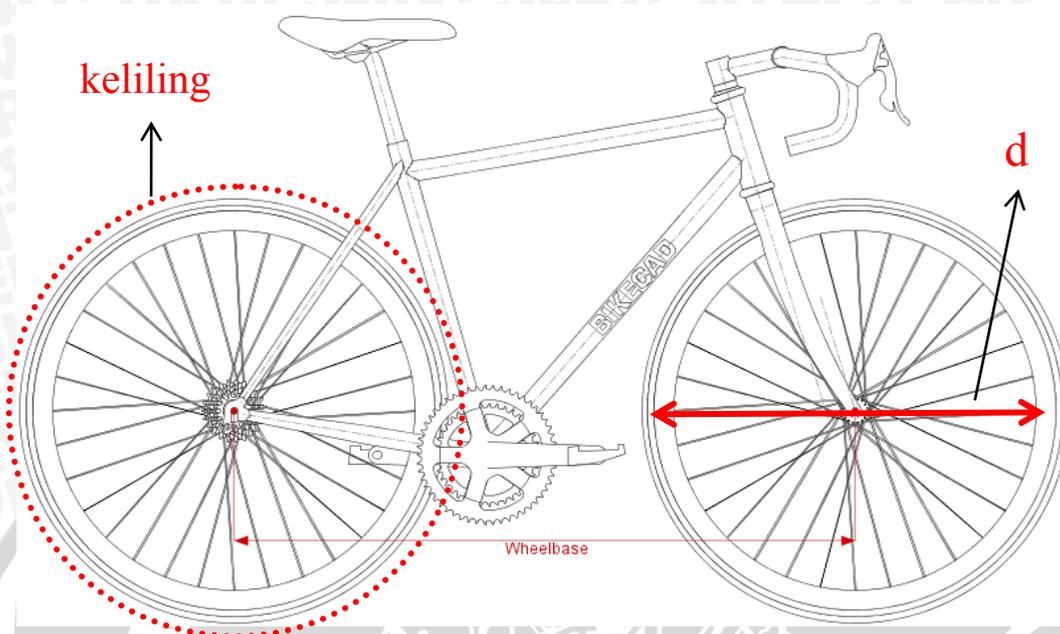
keterangan :

s = Jarak tempuh (km)

R = Jumlah putaran roda

Keliling = Keliling roda sepeda (km)

Roda sepeda yang digunakan berdiameter 26" (66,04 cm). Sehingga keliling roda adalah :



$$\begin{aligned} \text{Keliling} &= \pi \times d \\ &= 3,14 \times 66,04 \text{ cm} \\ &= 207,3656 \text{ cm} \end{aligned}$$

$$\begin{aligned} \text{Keliling} &= 207,3656 \text{ cm} / 100000 \longrightarrow \text{Konversi cm ke km} \\ &= 0,0021 \text{ km} \end{aligned}$$

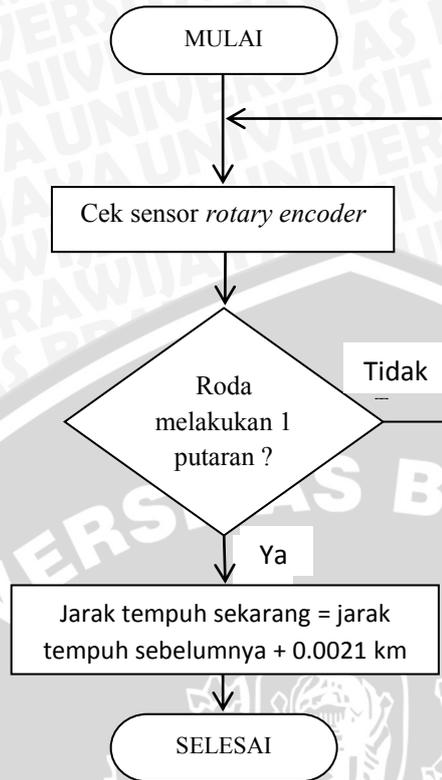
keterangan :

d = diameter (cm)

Dengan keliling roda sebesar 0,0021 km, maka setiap roda berputar akan menempuh jarak sebesar kelilingnya atau **0,0021 km/putaran**. Sebagai contoh, jika roda berputar sebanyak 1.000 kali, maka jarak tempuh sepeda adalah :

$$\begin{aligned} \text{Jarak tempuh} &= 1000 \text{ putaran} \times 0,0021 \text{ km/putaran} \\ &= 2,1 \text{ km} \end{aligned}$$

Diagram alir perhitungan jarak tempuh sepeda ditunjukkan dalam Gambar 3.8.



Gambar 3.8 Diagram alir perhitungan jarak tempuh

- **Perhitungan kecepatan sepeda listrik**

Untuk menghitung kecepatan, dapat menggunakan rumus dasar kecepatan yaitu :

$$v = s / t$$

keterangan :

v = Kecepatan (km/h)

s = Jarak tempuh (km)

t = Waktu tempuh (h)

; jarak tempuh = R × Keliling

Rumus dasar tersebut diturunkan hingga didapat RPM seperti berikut :

$$v = \frac{R \times \text{Keliling}}{1 \text{ jam}}$$

dimana 1 jam = 60 menit

$$= \frac{R \times \text{Keliling}}{60 \text{ menit}}$$

$$= \frac{R}{1 \text{ menit}} \times \frac{\text{Keliling}}{60 \text{ menit}}$$

dimana  $\frac{R}{1 \text{ menit}} = \text{RPM}$

$$= \text{RPM} \times \frac{\text{Keliling}}{60 \text{ menit}}$$

Setelah didapat RPM, rumus perlu diturunkan lagi untuk mendapatkan rev/s :

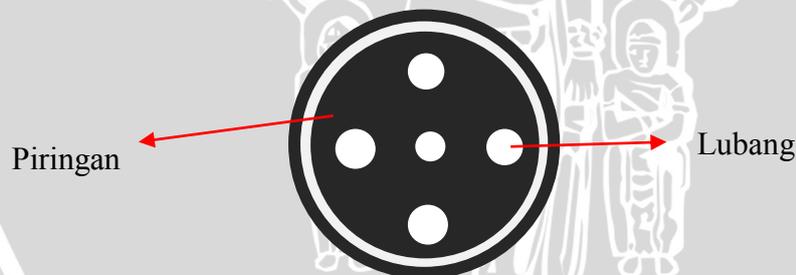
$$\begin{aligned}
 RPM &= \frac{R}{1 \text{ menit}} && \text{dimana } 1 \text{ menit} = 60 \text{ detik} \\
 &= \frac{R}{60 \text{ detik}} \\
 &= \frac{R}{1 \text{ detik}} \times \frac{1}{60 \text{ detik}} && \text{dimana } \frac{R}{1 \text{ detik}} = \text{rev/s} \\
 &= \text{rev/s} \times \frac{1}{60 \text{ detik}}
 \end{aligned}$$

keterangan :

RPM = *Revolutions Per Minute*

rev/s = *Revolutions per second*

Kemudian rev/s dapat diketahui dengan memanfaatkan fitur TIMER pada mikrokontroler untuk menghitung putaran roda tiap 1 detik. Putaran roda didapat dari keluaran sensor *rotary encoder* yang berupa *pulse*. Karena piringan sensor *rotary encoder* yang digunakan memiliki 4 buah lubang atau dengan kata lain menghasilkan 4 *pulse* setiap 1 putaran, maka rev/s perlu dibagi 4 untuk mendapatkan jumlah putaran.



Sebagai contoh, dalam 1 detik sensor *rotary encoder* pada sepeda dengan keliling roda 0,002 km mengirim 12 *pulse* ke mikrokontroler, maka jumlah putarannya adalah  $12 / 4 = 3$  putaran atau 3 rev/s. Kemudian RPM dapat dihitung :

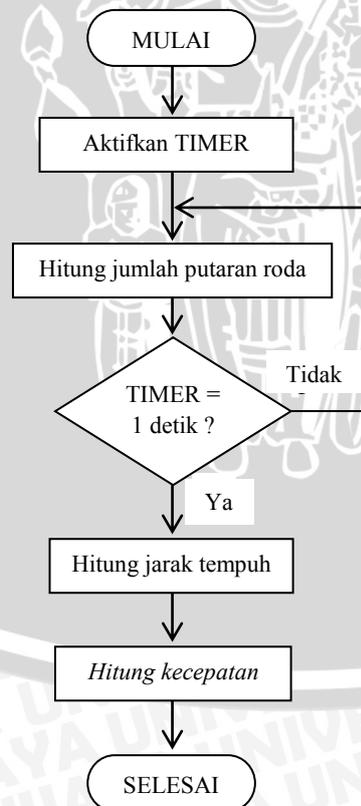
$$\begin{aligned}
 RPM &= \text{rev/s} \times \frac{1}{60 \text{ detik}} \\
 &= 3 \times \frac{1}{60 \text{ detik}} \\
 &= \frac{3 \text{ putaran}}{(1 \text{ detik}/60)} \times \frac{1}{(60 \text{ detik}/60)}
 \end{aligned}$$

$$\begin{aligned}
 &= \frac{180 \text{ putaran}}{1 \text{ menit}} \\
 &= 180
 \end{aligned}$$

Selanjutnya menghitung kecepatan :

$$\begin{aligned}
 v &= \text{RPM} \times \frac{\text{Keliling}}{60 \text{ menit}} \\
 &= \frac{180 \text{ putaran}}{(1 \text{ menit}/60)} \times \frac{0,002 \text{ km}}{(60 \text{ menit}/60)} \\
 &= \frac{21,6 \text{ km}}{1 \text{ jam}} \\
 &= 21,6 \text{ km/h}
 \end{aligned}$$

Maka jika roda sepeda berputar sebanyak 3 kali dalam 1 detik akan menghasilkan kecepatan 21,6 km/h. Proses perhitungan kecepatan diperlihatkan dalam Gambar 3.9.



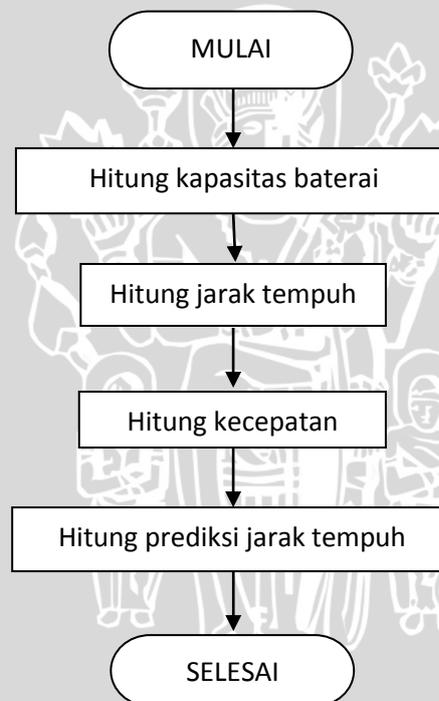
Gambar 3.9 Diagram alir perhitungan kecepatan

- **Perhitungan prediksi jarak tempuh**

Untuk menghitung prediksi jarak diperlukan data konsumsi arus motor listrik per kilometer. Data tersebut bisa diperoleh melalui pengujian secara eksperimental. Pengujian tersebut dilakukan dengan menjalankan sepeda listrik pada kecepatan konstan sejauh 100 meter dan diukur jumlah arus yang terpakai. Kemudian pengujian dilakukan berulang dengan kecepatan yang berbeda. Setelah didapat konsumsi arus per kilometer, prediksi jarak dapat dihitung dengan rumus berikut :

$$\text{Prediksi jarak (km)} = \frac{\text{Kapasitas baterai tersisa (mAh)}}{\text{Konsumsi arus per km (mAh/km)}}$$

Proses perhitungan prediksi jarak tempuh diperlihatkan dalam Gambar 3.10.



Gambar 3.10 Diagram alir perhitungan prediksi jarak tempuh

### 3.3 Pengujian Alat

Untuk menganalisis kinerja alat apakah sesuai dengan yang direncanakan maka dilakukan pengujian sistem. Pengujian dilakukan pada masing-masing blok pada perancangan perangkat keras serta pengujian keseluruhan untuk mengetahui perangkat lunak dapat bekerja dengan baik atau tidak.

Pada bagian ini pengujian dilakukan pada masing-masing blok. Pengujian ini bertujuan untuk mengetahui apakah masing-masing blok dapat bekerja sesuai dengan fungsinya seperti yang telah direncanakan. Pengujian tersebut meliputi :

### 3.3.1 Pengujian konsumsi arus listrik

Pengujian dilakukan di lintasan sejauh 100 meter dengan menggerakkan sepeda pada kecepatan tertentu. Kemudian arus yang terpakai selama sepeda bergerak diukur menggunakan Ahmeter dan dicatat hasilnya. Arus yang dibutuhkan motor listrik saat bekerja perlu diuji dan diketahui sebagai dasar perhitungan prediksi jarak tempuh.

Alat yang digunakan dalam pengujian pengukuran konsumsi arus motor listrik antara lain:

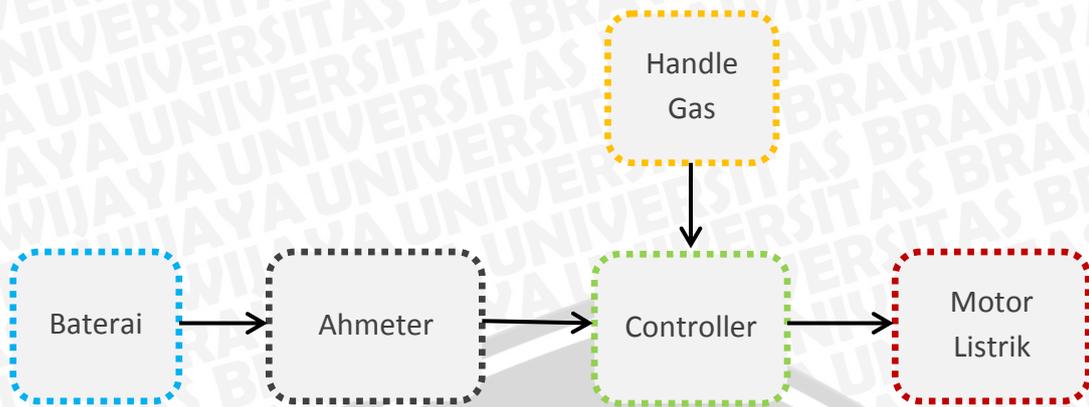
1. Ahmeter
2. Baterai
3. Motor listrik
4. *Controller* motor listrik
5. *Handle gas*.

Berikut merupakan spesifikasi dari motor listrik yang digunakan :

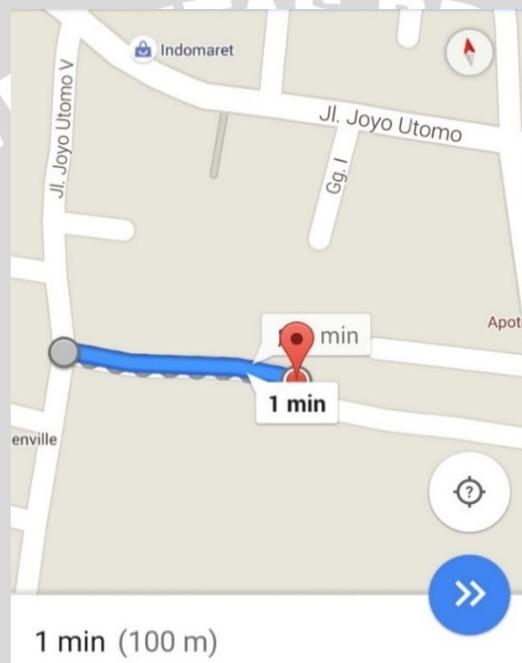
Putaran	300 rpm
Kecepatan	30 km/jam
Arus puncak	16 ampere
Daya	350 watt
Bobot	$\pm 5$ kg

Pengujian dilakukan sesuai prosedur berikut ini :

1. Hubungkan peralatan sesuai Gambar 3.11.
2. Bagi kecepatan motor listrik menjadi 7 menggunakan tuas pengatur kecepatan yang tersambung ke *controller* motor listrik.
3. Jalankan sepeda dengan kecepatan konstan sejauh 100 meter pada jalan datar. Pengujian dilakukan di Jl. Joyo Utomo, Merjosari, Kota Malang.
4. Catat jumlah arus yang ditampilkan Ahmeter.
5. Tiap kecepatan diulang sebanyak 5 kali.



Gambar 3.11 Skema pengujian konsumsi arus motor listrik



Gambar 3.12 Lokasi pengujian konsumsi arus motor listrik

### 3.3.2 Pengujian sensor arus

Pengujian dilakukan dengan cara menambah beban secara bertahap sehingga arus yang mengalir melalui sensor semakin besar. Kemudian hasil pengukuran sensor dibandingkan dengan alat ukur standar yaitu amperemeter. Pada bagian sensor arus perlu dilakukan pengujian untuk memastikan hasil pembacaan sensor arus ACS712 sesuai dengan alat pengukur arus yang sudah teruji.

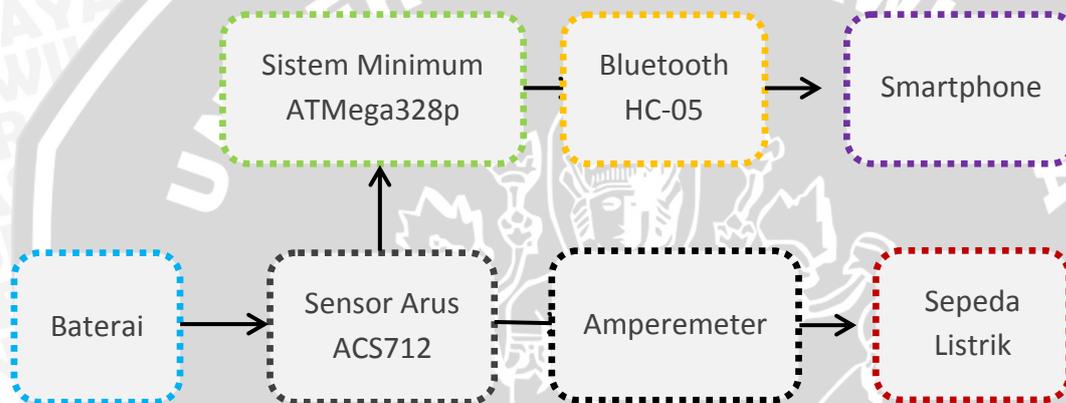
Alat yang dalam pengujian secara ini antara lain:

1. Sensor arus ACS712 – 30A.
2. Sistem minimum ATmega328p.
3. Amperemeter.

4. Modul *Bluetooth* HC-05.
5. *Smartphone*.
6. Sepeda listrik.

Pengujian dilakukan sesuai prosedur berikut ini :

1. Peralatan yang dibutuhkan dihubungkan seperti Gambar 3.13.
2. Arus yang mengalir ke motor listrik diubah semakin besar secara bertahap dengan cara mengatur *handle gas*.
3. Saat amperemeter menunjukkan nilai arus yang stabil, hasil pembacaan sensor arus ACS712 yang ditampilkan di *smartphone* dicatat.
4. Percobaan diulang untuk nilai arus selanjutnya.



Gambar 3.13 Skema pengujian pembacaan sensor arus ACS712

### 3.3.3 Pengujian komunikasi *bluetooth*

Pengujian dilakukan untuk mengetahui apakah mikrokontroler dapat mengirimkan data ke *smartphone*. Pengujian dilakukan dengan mengirim data dari mikrokontroler dengan jarak yang bervariasi. Dengan melakukan pengujian ini, dapat diketahui jarak maksimal komunikasi data dari modul *bluetooth* HC-05 ke *smartphone*.

Alat yang digunakan dalam pengujian ini antara lain :

1. *Smartphone* Android
2. Meteran gulungan
3. Modul *bluetooth* HC-05
4. Sistem Minimum ATmega328p

Pengujian dilakukan sesuai prosedur berikut ini :

1. Pengujian ini dilakukan di tempat outdoor, mengingat sepeda listrik lebih sering digunakan di ruangan terbuka.
2. Modul *Bluetooth* HC-05 dihubungkan ke sistem minimum ATmega328p, pin RX HC-05 dihubungkan pada pin TX ATmega328p, pin TX HC-05 dihubungkan dengan pin RX ATmega328p, pin Vcc pada HC-05 dihubungkan pada Vcc ATmega328p, dan pin GND HC-05 dihubungkan pada GND ATmega328p.
3. Setelah itu HC-05 akan mengirim data menuju *smartphone*, ubah jarak antara HC-05 dengan *smartphone* semakin jauh setiap 1 meter hingga koneksi terputus.



Gambar 3.14 Skema pengujian jarak komunikasi *Bluetooth* HC-05

### 3.3.4 Pengujian jarak tempuh

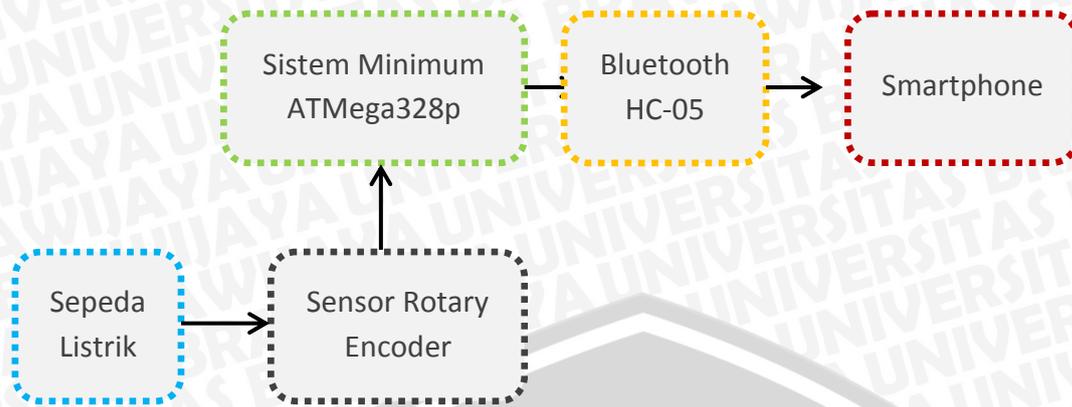
Pengujian dilakukan dengan membandingkan hasil pembacaan sensor *rotary encoder* dengan alat ukur jarak tempuh. Alat ukur standar yang digunakan adalah odometer. Pengujian ini bertujuan untuk mengetahui ketelitian hasil pengukuran jarak tempuh.

Alat yang digunakan dalam pengujian secara ini antara lain:

1. Sensor *rotary encoder*.
2. Odometer
3. Sistem minimum ATmega328p.
4. Modul *Bluetooth* HC-05.
5. *Smartphone*.
6. Sepeda listrik.

Pengujian dilakukan sesuai prosedur berikut ini :

1. Peralatan dihubungkan sesuai Gambar 3.15.
2. Sepeda listrik dihidupkan.
3. Menjalankan sepeda dengan memutar *handle gas* hingga jarak tertentu.
4. Sepeda dihentikan kemudian jarak yang ditampilkan sensor dan odometer dicatat.
5. Langkah ke-3 dan ke-4 diulang sebanyak 4 kali dengan jarak berbeda.



Gambar 3.15 Skema pengujian pengukuran jarak tempuh sepeda listrik

### 3.3.5 Pengujian kecepatan

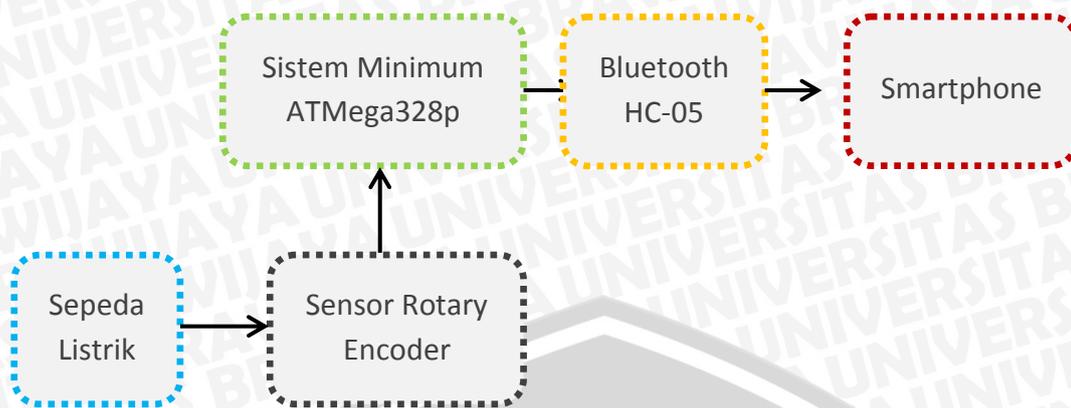
Pengujian dilakukan dengan menjalankan sepeda dengan kecepatan yang berubah-ubah. Kemudian hasilnya dibandingkan dengan alat ukur kecepatan yaitu speedometer. Pengujian ini bertujuan untuk mengetahui tingkat ketelitian pengukuran kecepatan.

Alat yang digunakan dalam pengujian secara ini antara lain:

1. Sensor *rotary encoder*.
2. Speedometer.
3. Sistem minimum ATmega328p.
4. Modul *Bluetooth* HC-05.
5. *Smarthphone*.
6. Sepeda listrik.

Pengujian dilakukan sesuai prosedur berikut ini :

1. Peralatan dihubungkan sesuai Gambar 3.16.
2. Menyiapkan *speedometer*..
3. Sepeda listrik dihidupkan.
4. Menjalankan sepeda dengan memutar *handle gas* hingga kecepatan konstan.
5. Kecepatan yang ditampilkan *Speedometer* dan hasil pengukuran *rotary encoder* dicatat.
6. Mengulang langkah ke-4 dan ke-5 sebanyak 9 kali dengan kecepatan berbeda.



Gambar 3.16 Skema pengujian pengukuran kecepatan sepeda listrik

### 3.3.6 Pengujian keseluruhan sistem

Pengujian keseluruhan sistem ini dilakukan dengan menghubungkan semua hardware yang dibuat sesuai diagram blok dan memasukkan program berupa *software* yang bekerja untuk mengendalikan hardware yang telah dibuat. Sistem bekerja dengan baik jika dapat berjalan sesuai *flowchart* yang telah direncanakan. Tujuan pengujian keseluruhan adalah untuk mengetahui apakah sistem yang telah dibuat dapat bekerja sesuai dengan perancangan. Parameter dalam pengujian keseluruhan antara lain hasil perhitungan kapasitas baterai, kecepatan, dan prediksi jarak tempuh serta keberhasilan pengiriman data dari mikrokontroler ke *smartphone*.

Alat yang digunakan dalam pengujian secara ini antara lain:

1. *Speedometer*.
2. *Odometer*.
3. *Smartphone*.
4. Sepeda listrik.

Pengujian dilakukan sesuai prosedur berikut ini :

1. Membuka aplikasi *smartphone* “GOWEZ”.
2. Sepeda listrik dihidupkan.
3. Tuas pengatur kecepatan diputar hingga sepeda bergerak.
4. Hasil pengukuran yang ditampilkan di *smartphone* dicatat.
5. Hasil pengukuran *speedometer* dan *odometer* dicatat.
6. Pengujian diulang sebanyak 4 kali.



Gambar 3.17 Skema pengujian keseluruhan sistem

