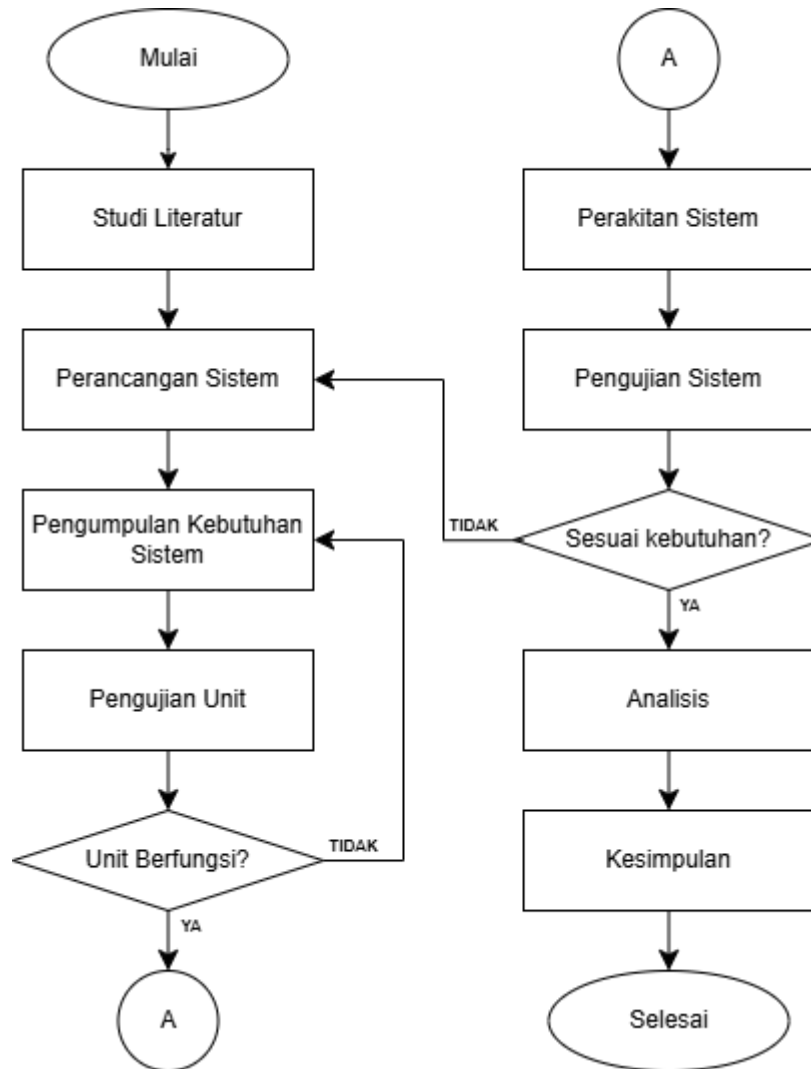


BAB III

METODE PENELITIAN

3.1 Flowchart Penelitian



Gambar 3. 1 *Flowchart* Penelitian

Gambar 3.1 menunjukkan flowchart tahapan penelitian yang dilakukan. Penelitian ini diawali dengan studi literatur untuk mengumpulkan informasi dan teori yang relevan, dilanjutkan dengan perancangan sistem serta pengumpulan kebutuhan sistem yang mencakup komponen dan alat yang dibutuhkan. Setelah itu dilakukan pengujian unit untuk memastikan setiap komponen berfungsi dengan baik. Jika unit berfungsi, maka dilanjutkan ke tahap perakitan sistem.

Setelah sistem dirakit, dilakukan pengujian sistem secara menyeluruh untuk melihat apakah sistem berjalan sesuai dengan kebutuhan. Jika hasil pengujian belum sesuai, maka dilakukan perbaikan atau penyesuaian hingga sesuai dengan spesifikasi yang ditentukan. Tahap selanjutnya adalah analisis data hasil pengujian dan diakhiri dengan pengambilan kesimpulan dari seluruh proses penelitian.

3.2 Studi literatur

Pada tahap ini, peneliti mempelajari penelitian-penelitian sebelumnya yang berkaitan dengan estimasi *SOC* baterai lithium ion, terutama penelitian yang menggunakan fungsi *cut of charge* dan *discharge*. Peneliti juga mempelajari literatur tentang baterai lithium ion, terutama literatur tentang karakteristik baterai lithium ion dan proses penuaan baterai lithium ion. Studi literatur ini bertujuan untuk memahami kebutuhan dan spesifikasi alat yang akan dibangun. Peneliti menggunakan informasi yang diperoleh dari studi literatur ini untuk membuat desain sistem alat yang tepat.

Berdasarkan studi literatur, peneliti menemukan bahwa estimasi *SOC* baterai lithium ion dapat dilakukan dengan berbagai metode. Namun metode yang akan peneliti gunakan adalah metode *Coulomb Counting*, yaitu menghitung muatan yang masuk dan keluar dari baterai. Peneliti juga menemukan bahwa fungsi *cut of charge* dan *discharge* penting untuk diterapkan pada mobil listrik tenaga surya. Fungsi *cut of charge* berfungsi untuk mencegah baterai terisi penuh hingga 100%, sedangkan fungsi *cut of discharge* berfungsi untuk mencegah baterai terpakai habis hingga 0%.

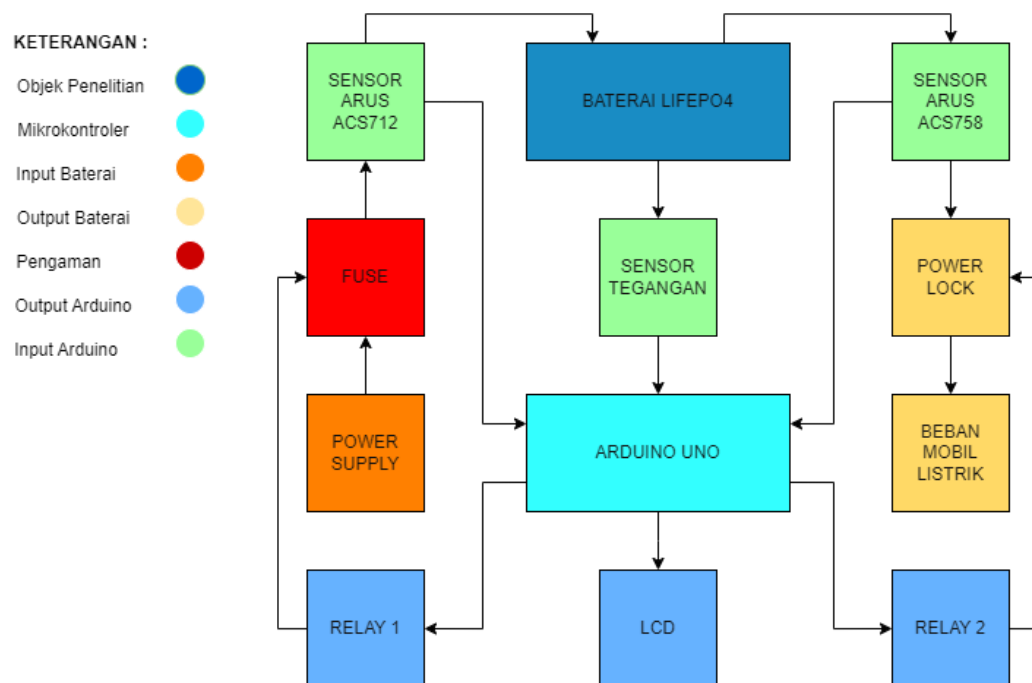
3.3 Perancangan Sistem

Pada tahap ini, dilakukan pembuatan desain sistem alat yang akan dibuat, termasuk pemilihan bahan, blok diagram, *flowchart* sistem serta skema *wiring* tiap

komponen. Desain sistem alat harus memperhatikan kebutuhan dan spesifikasi yang telah ditetapkan pada tahap studi literatur. Dalam hal pemilihan bahan, peneliti memilih bahan yang sesuai dengan karakteristik baterai lithium ion. Bahan-bahan yang sering digunakan untuk membuat alat estimasi *SOC* baterai lithium ion meliputi mikrokontroler, sensor tegangan, sensor arus, dan *LCD* untuk menampilkan tingkat *SOC* yang terukur.

3.3.1 Blok Diagram Sistem

Rencana teknis pertama dalam penelitian ini adalah membuat blok diagram untuk acuan atau gambaran besar tentang sistem yang akan dibuat.



Gambar 3. 2 Blok diagram sistem.

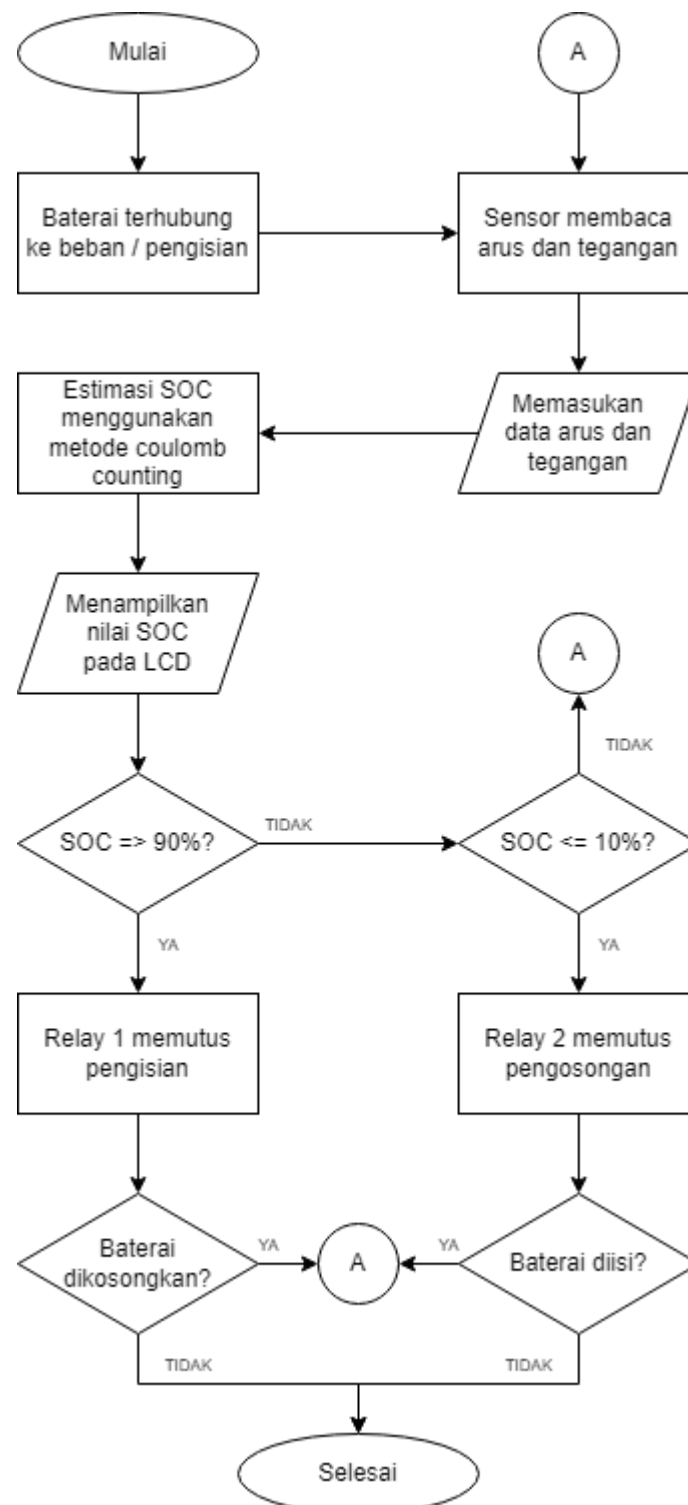
Gambar 3.2 menunjukkan perancangan sistem yang digunakan dalam penelitian ini. Baterai LiFePO₄ berfungsi sebagai objek utama yang diteliti. Sebagai sumber pengisian, menggunakan *power supply* (PSU), sementara beban pada sistem adalah motor BLDC.

Sensor arus ACS712 dan sensor tegangan dipasang di jalur antara sumber pengisian (PSU) dan baterai untuk mengukur arus serta tegangan masuk selama proses pengisian. Di sisi lain, sensor arus ACS758 dipasang di antara baterai dan motor BLDC untuk memantau arus keluar saat baterai digunakan.

Semua sensor terhubung ke mikrokontroler Arduino Uno yang berfungsi memproses data dari sensor. Arduino menjalankan algoritma *Coulomb Counting* untuk menghitung nilai *State of Charge (SOC)* dari baterai secara *real-time*. Hasil pengukuran *SOC* ditampilkan melalui *LCD* dan digunakan sebagai acuan untuk mengaktifkan atau menonaktifkan *relay* 1 dan *relay* 2.

Jika nilai *SOC* melebihi atau turun di bawah batas tertentu, sistem akan secara otomatis memutus arus pengisian atau pengosongan melalui *relay*, sehingga mencegah terjadinya *overcharge* maupun *overdischarge* pada baterai. Fuse juga disisipkan sebagai pengaman tambahan untuk mencegah arus lebih yang dapat merusak komponen.

3.3.2 Flowchart Sistem



Gambar 3. 3 Flowchart Sistem

1. Inisialisasi Sistem

Pada awalnya, ketika sistem dinyalakan, *relay* pengisian (*Relay 1*) dan *relay* pengosongan (*Relay 2*) dirancang dalam keadaan *Normal Close (NC)*. Artinya, kedua *relay* memungkinkan arus mengalir normal untuk pengisian dan pengosongan. Sistem siap bekerja, dan *LCD* siap menampilkan informasi yang diperlukan, seperti *SOC* dan tegangan baterai.

2. Koneksi Baterai ke Beban atau Pengisian

Selanjutnya, baterai akan terhubung ke salah satu dari dua kondisi utama:

Pengosongan (*Discharge*): Baterai digunakan untuk memberikan daya ke beban, seperti motor listrik pada mobil listrik.

Pengisian (*Charge*): Baterai dihubungkan ke *Charger* untuk mengisi daya.

Proses ini menentukan alur selanjutnya pada sistem apakah untuk pengosongan atau pengisian, ataupun dua kondisi sekaligus.

3. Pembacaan Sensor Arus dan Tegangan

Setelah koneksi dilakukan, sistem akan membaca data dari sensor arus dan sensor tegangan yang terpasang pada baterai. Sensor arus mengukur jumlah arus yang mengalir ke dalam (*Charge*) atau keluar (*Discharge*) dari baterai. Sedangkan sensor tegangan mengukur tegangan pada baterai untuk memastikan bahwa tegangan berada pada level yang aman dan optimal.

4. Menghitung *SOC* dengan Metode *Coulomb Counting*

Setelah sensor membaca data arus dan tegangan dan kemudian mengirimkannya ke arduino, sistem akan melakukan perhitungan *SOC* menggunakan metode *Coulomb Counting*. Metode ini mengestimasi tingkat pengisian atau pengosongan baterai berdasarkan jumlah arus yang mengalir selama

periode waktu tertentu. Rumus yang digunakan untuk menghitung *SOC* adalah:

$$SOC(t) = SOC(t-1) \pm (I \times \Delta t) / C_n \quad (3.1)$$

Keterangan:

SOC(t): nilai *SOC* saat ini

I: arus masuk/keluar (A)

Δt : selang waktu (jam)

C_n: kapasitas baterai aktual (72 Ah)

Jika arus masuk (pengisian), *SOC* bertambah. Jika arus keluar (pengosongan), *SOC* berkurang. Perhitungan ini memberi nilai persentase *SOC* yang menunjukkan seberapa penuh atau habis baterai.

5. Menampilkan *SOC* dan Tegangan pada *LCD*

Setelah *SOC* dihitung, hasilnya akan ditampilkan pada layar *LCD* yang terpasang pada alat. Layar ini juga akan menampilkan tegangan baterai yang dapat membantu pengguna untuk memantau kondisi baterai secara lebih rinci. *LCD* memberikan informasi yang jelas mengenai status baterai, seperti apakah baterai sudah mendekati penuh atau hampir habis.

6. Pengambilan Keputusan Berdasarkan *SOC*

Berdasarkan nilai *SOC* yang dihitung, sistem akan mengambil keputusan untuk mengatur status *relay* pengisian dan pengosongan. Proses pengambilan keputusan ini terdiri dari beberapa tahap:

Jika *SOC* baterai mencapai atau melebihi 90%, itu berarti baterai hampir penuh. Pada titik ini, *relay* pengisian (*Relay 1*) akan diputus atau berubah menjadi *Normally Open (NO)*, yang berarti pengisian akan dihentikan. Baterai tidak akan menerima arus lebih banyak untuk mencegah *overcharging* (pengisian berlebih).

Sebaliknya, jika *SOC* baterai turun di bawah atau sama dengan 10%, itu berarti baterai hampir habis. Pada titik ini, *relay* pengosongan (*Relay 2*) akan diputus atau berubah menjadi *Normally Open (NO)*, yang berarti pengosongan akan dihentikan. Sistem akan menghentikan pengosongan untuk mencegah kerusakan pada baterai karena tegangan yang terlalu rendah.

Ketika *SOC* berada di antara 10% dan 90%, kedua *relay* (pengisian dan pengosongan) akan kembali ke posisi *Normally Close (NC)*. Dengan posisi ini, sistem memungkinkan pengisian dan pengosongan baterai berjalan seperti biasa, tergantung pada kebutuhan beban atau *Charger*.

7. Kembali ke Pembacaan Sensor (*Looping*)

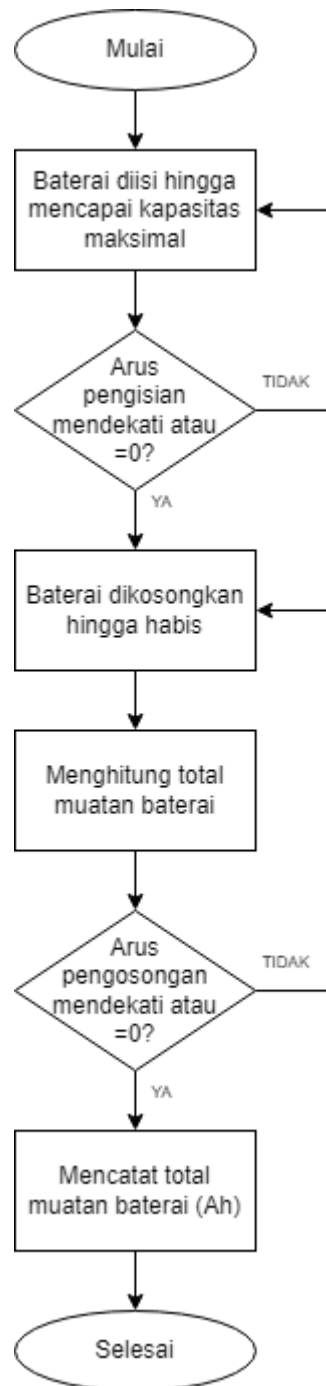
Setelah pengambilan keputusan, sistem tidak berhenti, melainkan kembali ke proses pembacaan sensor arus dan tegangan. Hal ini memastikan sistem memantau keadaan baterai secara terus-menerus dan mengatur pengisian serta pengosongan dengan tepat. Proses ini berjalan dalam sebuah *looping* yang tak terputus selama sistem aktif.

8. Selesai

Sistem akan terus melakukan proses ini selama beroperasi, secara otomatis memantau *SOC* dan melakukan *cut-off* sesuai dengan kondisi baterai. Sistem dapat terus berfungsi tanpa penghentian otomatis kecuali sistem dimatikan.

3.3.2.1 Initial Kapasitas Awal Baterai

Karena baterai yang dipakai dalam penelitian ini berstatus sebagai baterai bekas pakai, maka penentuan status kapasitas awal baterai diperlukan untuk keakuratan perhitungan *state of Charge* baterai *Lithium-Ion*.



Gambar 3. 4 Flowchart Initial Kapasitas Baterai

Gambar 3.4 merupakan diagram alir proses penentuan kapasitas awal (initial capacity) baterai *Lithium-Ion* yang digunakan dalam penelitian ini. Mengingat baterai yang digunakan adalah baterai bekas pakai, maka pengukuran kapasitas aktual sangat penting untuk memperoleh dasar yang akurat dalam perhitungan State of Charge (*SOC*) selama pengujian sistem. Langkah-langkah proses pada diagram

dijelaskan sebagai berikut:

1. Mulai

Proses dimulai dengan mempersiapkan baterai dan sistem pemantauan arus serta tegangan.

2. Pengisian Baterai hingga Kapasitas Maksimal

Baterai diisi daya hingga mencapai tegangan maksimum sesuai spesifikasi.

Tujuannya adalah memastikan baterai dalam kondisi penuh sebelum dilakukan pengosongan total.

3. Deteksi Arus Pengisian

Sistem memantau arus pengisian, dan proses pengisian dianggap selesai apabila arus mendekati nol (0), yang menandakan baterai sudah penuh dan masuk ke fase *constant voltage*.

4. Pengosongan Baterai Hingga Habis

Setelah pengisian selesai, baterai kemudian dikosongkan hingga arus pengosongan mendekati nol, atau hingga mencapai batas tegangan terendah yang aman (*cut-off voltage*).

5. Menghitung Total Muatan Baterai

Selama proses pengosongan, sistem menghitung total muatan yang dikeluarkan (biasanya dalam satuan Ah), dengan mengintegrasikan arus terhadap waktu.

6. Deteksi Arus Pengosongan

Ketika arus pengosongan mendekati nol, proses pengosongan dianggap selesai.

7. Mencatat Total Muatan Baterai

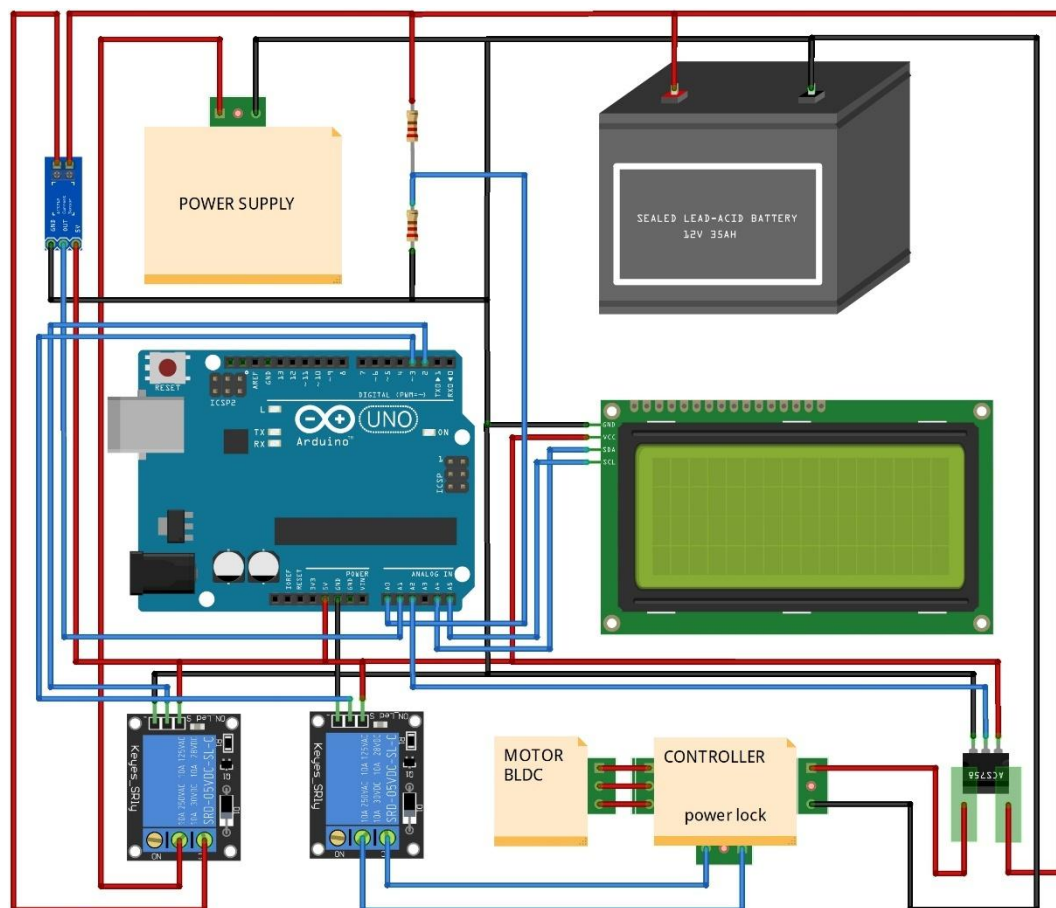
Nilai total muatan yang berhasil dikeluarkan dari baterai dicatat sebagai kapasitas aktual (initial capacity) baterai tersebut.

8. Selesai

Proses selesai dan nilai kapasitas awal dapat digunakan sebagai dasar perhitungan *SOC* pada siklus pengujian selanjutnya.

Dengan metode ini, sistem dapat menentukan kapasitas aktual baterai secara akurat, yang sangat penting dalam konteks penggunaan baterai bekas untuk memastikan kinerja dan perlindungan selama pengisian dan pengosongan.

3.3.3 *Wiring Diagram* Sistem



Gambar 3. 5 *Wiring Diagram* Sistem

Gambar 3.5 menunjukkan *Wiring diagram* sistem yang dirancang untuk alat

estimasi *State of Charge (SOC)* baterai *Lithium-Ion* dengan fungsi *cut-off Charge* dan *Discharge* pada mobil listrik. *Wiring* ini dibuat dengan tujuan memudahkan instalasi dan mengetahui komponen-komponen yang digunakan agar menghindari atau meminimalisir terjadinya kesalahan instalasi sistem yang dapat mengakibatkan kegagalan sistem seperti sensor tidak terbaca, atau bahkan terjadinya korsleting.

Sistem ini menggunakan beberapa komponen utama, seperti Arduino Uno, sensor arus, *LCD display*, dual modul *relay*, dan *power supply* tambahan. Berikut penjelasan masing-masing sambungan :

1. Arduino Uno

Arduino Uno berperan sebagai pengendali dari semua komponen yang terpasang. Mikrokontroler ini membaca data dari semua sensor, yaitu sensor tegangan dan sensor arus, mengolah informasi, mengendalikan modul *relay* untuk fungsi *cut-off*, dan menampilkan hasil pengolahan data ke layar *LCD*.

2. Sensor Arus

Sensor arus yang digunakan yaitu ACS758 dipasang secara seri dengan jalur arus dari baterai ke beban (motor BLDC), sedangkan ACS712 dipasang secara seri dari *Power supply* ke baterai. Sehingga dapat mengukur arus yang masuk atau keluar dari baterai. *Output* dari sensor arus ini dihubungkan ke pin analog Arduino untuk dibaca nilainya kemudian ditampilkan pada modul *LCD* serta direkam oleh modul micro sd untuk kemudian diolah datanya.

3. Power supply

Sistem ini dilengkapi dengan sumber daya tambahan untuk memberi catu daya ke Arduino dan modul lainnya. *Power supply* dihubungkan ke *Input power* Arduino melalui regulator agar tegangan tetap stabil.

4. Baterai *Lithium-Ion*

Sumber utama energi dalam sistem adalah baterai 48V 100Ah. Baterai ini terhubung ke beban (motor BLDC) melalui rangkaian kontrol, sekaligus menjadi objek pengukuran *SOC*.

5. Layar *LCD* 20x4

LCD digunakan untuk menampilkan informasi penting seperti persentase *SOC*, arus, tegangan, serta status pengisian atau pengosongan. *LCD* dihubungkan menggunakan protokol I2C untuk menghemat penggunaan pin digital Arduino.

6. Modul *Relay*

Terdapat dua modul *relay* yang berfungsi sebagai saklar elektronik untuk memutus dan menghubungkan jalur pengisian (charging) dan pengosongan (discharging) sesuai perintah Arduino. *Relay* ini dikendalikan berdasarkan nilai *SOC* yang dihitung, untuk menjaga baterai tidak *overcharge* maupun *overdischarge*.

7. Motor BLDC dan Controller

Motor BLDC adalah beban utama dalam sistem ini, dikendalikan oleh controller motor. Sistem pengunci daya (power lock) pada controller terhubung dengan *relay*, sehingga saat *relay* aktif/mati, arus ke motor juga diputus atau dihubungkan sesuai kebutuhan.

8. Jalur *Grounding* dan *VCC*

Semua komponen memiliki jalur ground (*GND*) yang saling terhubung, memastikan sistem memiliki referensi tegangan yang sama. Jalur *VCC* dari *power supply* digunakan untuk mensuplai tegangan operasional ke Arduino, *relay*, dan *LCD*.

3.4 Pengumpulan Kebutuhan Sistem

Pada tahap ini, dilakukan proses pengumpulan alat dan bahan yang diperlukan untuk mendukung pelaksanaan penelitian sesuai dengan rancangan sistem yang telah dibuat sebelumnya. Penelitian ini berfokus pada pengujian dan pemantauan nilai *SOC* baterai *Lithium-Ion*, yang menjadi objek penelitian. Dalam pengujian ini, mobil listrik digunakan sebagai beban untuk menunjukkan kondisi kerja baterai secara nyata.

Selain itu, berbagai komponen pendukung juga diperlukan untuk membangun sistem monitoring dan *control*, seperti mikrokontroler, sensor, dan komponen elektronik lainnya. Alat dan bahan yang dikumpulkan mencakup perangkat keras untuk perakitan sistem, serta perlengkapan pendukung untuk proses instalasi dan pengujian.

Berikut adalah daftar alat dan bahan yang digunakan dalam penelitian ini:

Tabel 3. 1 Alat

No	Alat	Jumlah
1.	Laptop	1
2.	Multimeter	1
3.	Solder	1
4.	Obeng	1

Alat-alat tersebut digunakan dalam proses perancangan, instalasi, dan pengujian sistem. Laptop digunakan untuk pemrograman mikrokontroler, multimeter untuk pengukuran tegangan dan arus, serta solder dan obeng untuk perakitan komponen elektronik.

Tabel 3. 2 Bahan

No	Bahan	Jumlah
1.	Arduino Uno	1
2.	Baterai	1
3.	<i>LCD 20x4</i>	1
4.	Papan PCB	Secukupnya
5.	Kabel	Secukupnya
6.	Sensor Tegangan	1
7.	Plastik Box	1
8.	Timah (Tinol)	Secukupnya
9.	Sensor Arus ACS712	1
10.	<i>Relay</i>	2
11.	Baud	Secukupnya
12.	Sensor Arus ACS758	1

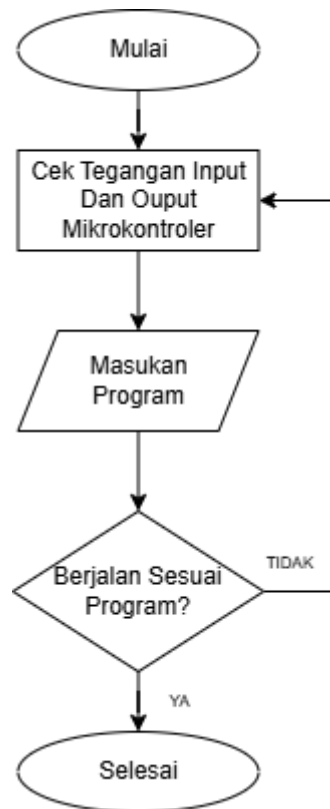
Bahan-bahan di atas merupakan komponen utama dalam perakitan sistem monitoring baterai. Arduino Uno berfungsi sebagai otak sistem yang mengatur dan memproses data dari sensor. Sensor tegangan dan arus (ACS712 dan ACS758) digunakan untuk mengukur kondisi baterai secara *real-time*. *LCD 20x4* digunakan untuk menampilkan data, sementara *relay* berfungsi untuk pengendalian beban. Komponen lain seperti kabel, papan PCB, dan timah solder digunakan dalam proses perakitan rangkaian.

3.5 Pengujian Unit

Pada tahap ini, tiap komponen yang akan digunakan diuji terlebih dahulu untuk mengetahui apakah komponen dapat bekerja dengan baik atau tidak. Pengujian ini

meliputi: Pengujian Arduino uno, pengujian *LCD*, pengujian sensor arus, pengujian sensor tegangan, *relay*, modul *SD Card*.

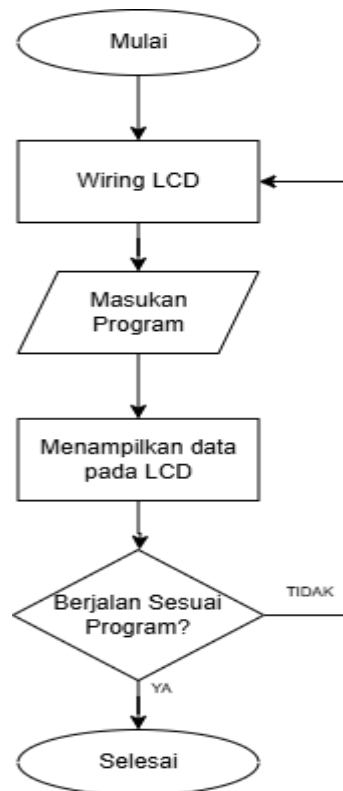
3.5.1 Pengujian Arduino Uno



Gambar 3. 6 Flowchart pengujian Arduino uno

Gambar 3.6 menunjukkan alur pengujian awal sistem untuk memastikan mikrokontroler berfungsi dengan baik. Pengujian dimulai dengan mengecek tegangan input dan output guna memastikan catu daya serta port I/O dalam kondisi normal. Selanjutnya, program dimasukkan ke mikrokontroler dan diamati apakah sistem berjalan sesuai instruksi. Jika tidak, dilakukan pengecekan ulang hingga sistem bekerja sebagaimana mestinya. Pengujian dinyatakan selesai apabila mikrokontroler menjalankan program dengan benar.

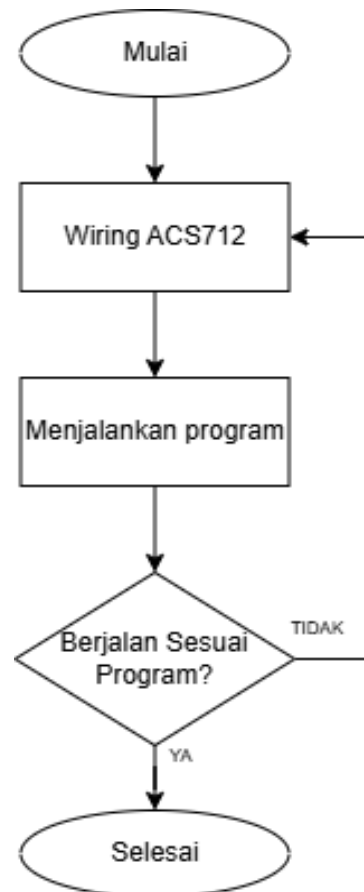
3.5.2 Pengujian *LCD*



Gambar 3. 7 Flowchart pengujian *LCD*

Gambar 3.7 menunjukkan alur pengujian *LCD* untuk memastikan bahwa perangkat dapat menampilkan data dengan baik. Pengujian ini diawali dengan proses wiring *LCD*, yaitu menyambungkan pin *LCD* ke mikrokontroler sesuai dengan konfigurasi yang dibutuhkan. Setelah koneksi fisik dilakukan, langkah berikutnya adalah memasukkan program sederhana ke dalam mikrokontroler yang bertujuan untuk menguji fungsi dasar *LCD*, seperti menampilkan karakter atau pesan teks. Selanjutnya diamati apakah data berhasil ditampilkan pada layar *LCD*. Jika *LCD* tidak menampilkan data sesuai dengan program, maka dilakukan pengecekan ulang terhadap sambungan kabel (*wiring*) dan program yang dimasukkan. Proses ini diulang hingga *LCD* bekerja sesuai harapan. Jika tampilan pada *LCD* sudah sesuai dengan instruksi dalam program, maka pengujian dinyatakan selesai.

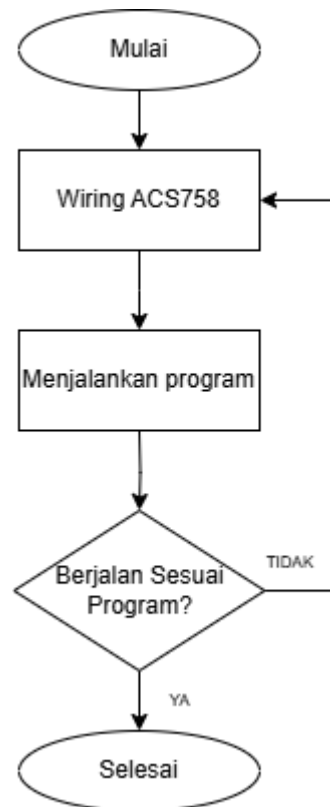
3.5.3 Pengujian sensor arus ACS712



Gambar 3. 8 Flowchart pengujian sensor arus

Gambar 3.8 menunjukkan alur pengujian sensor arus ACS712 untuk memastikan bahwa sensor berfungsi dengan baik. Pengujian diawali dengan proses *wiring* sensor ke mikrokontroler sesuai pin yang digunakan. Selanjutnya, program sederhana dimasukkan dan dijalankan untuk membaca nilai arus yang terdeteksi oleh sensor. Jika sensor tidak bekerja sesuai dengan program, maka dilakukan pengecekan ulang pada sambungan dan kode program. Pengujian dinyatakan selesai apabila sensor berhasil menampilkan pembacaan arus secara benar sesuai instruksi program.

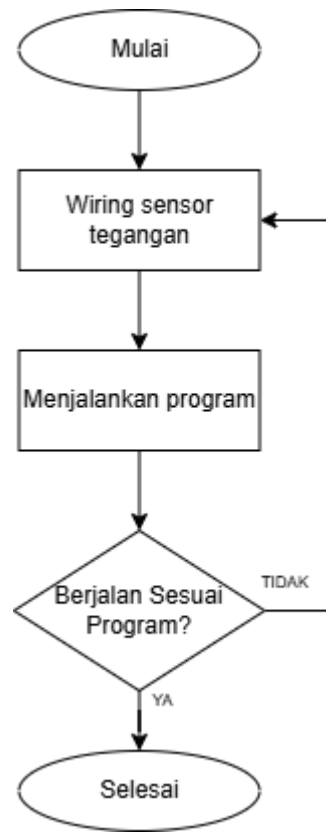
3.5.4 Pengujian Arus ACS758



Gambar 3. 9 Flowchart pengujian ACS758

Gambar 3.9 menunjukkan alur pengujian sensor arus ACS758 untuk memastikan bahwa sensor berfungsi dengan baik. Pengujian dimulai dengan melakukan *wiring* sensor ke mikrokontroler sesuai konfigurasi yang diperlukan. Selanjutnya, program sederhana dimasukkan dan dijalankan untuk membaca nilai arus dari sensor. Apabila hasil pembacaan tidak sesuai dengan yang diharapkan, maka dilakukan pengecekan ulang pada sambungan kabel dan kode program. Pengujian dinyatakan berhasil apabila sensor dapat menampilkan pembacaan arus secara akurat sesuai dengan instruksi program.

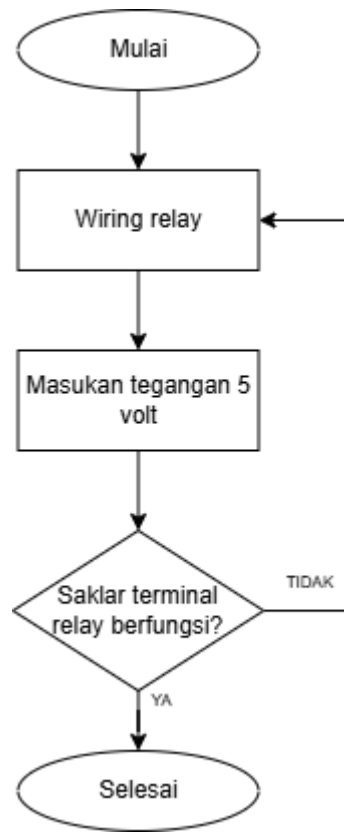
3.5.5 Pengujian sensor tegangan



Gambar 3. 10 Flowchart pengujian sensor tegangan

Gambar 3.10 menunjukkan alur pengujian sensor tegangan untuk memastikan bahwa sensor berfungsi dengan baik. Pengujian dimulai dengan melakukan *wiring* sensor ke mikrokontroler sesuai dengan konfigurasi rangkaian pembagi tegangan yang digunakan. Selanjutnya, program sederhana dijalankan untuk membaca nilai tegangan dari sensor. Jika pembacaan tidak sesuai dengan hasil yang diharapkan, maka dilakukan pengecekan ulang pada sambungan kabel dan kode program. Pengujian dianggap berhasil apabila sensor dapat menampilkan nilai tegangan secara akurat sesuai dengan program.

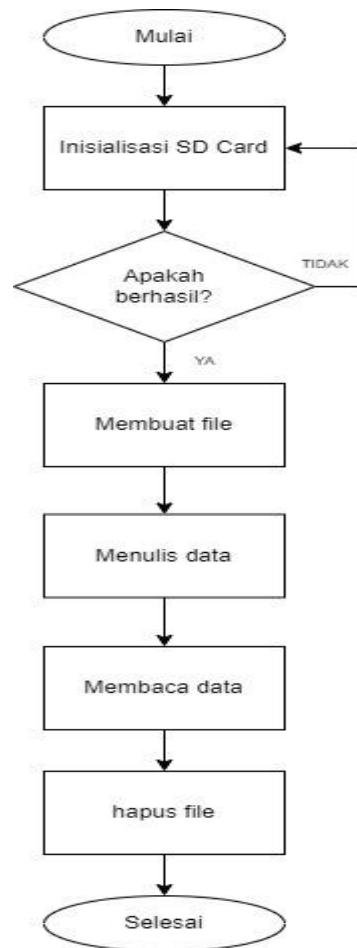
3.5.6 Pengujian *Relay*



Gambar 3. 11 Flowchart pengujian *Relay*

Gambar 3.11 menunjukkan alur pengujian modul relay untuk memastikan bahwa relay berfungsi dengan baik. Pengujian dilakukan dengan menyambungkan kabel (*wiring*) sesuai konfigurasi, kemudian diberikan tegangan sebesar 5 volt pada terminal koil relay. Setelah itu, dilakukan pengecekan apakah terminal saklar relay merespons dengan benar (terbuka atau tertutup sesuai karakteristik relay). Jika relay tidak berfungsi, maka dilakukan pengecekan ulang pada sambungan dan sumber tegangan. Pengujian dianggap berhasil apabila saklar relay dapat bekerja sesuai dengan kondisi aktif pada koil.

3.5.7 Pengujian Modul SD Card



Gambar 3. 12 Flowchart Pengujian Micro SD

Gambar 3. 12 menunjukkan pengujian dimulai dengan menginisialisasi modul SD card. Jika modul berhasil terhubung, maka dilanjutkan dengan membuat *file* baru, menulis data ke dalam *file*, lalu membaca kembali isi *file* tersebut. Setelah data terbaca dengan benar, *file* ditutup dan dihapus dari kartu memori. Namun, jika inisialisasi gagal, proses akan diulang dari awal sampai modul berhasil terhubung. Pengujian selesai jika semua langkah berjalan dengan baik tanpa kesalahan.

3.6 Perakitan Sistem

Apabila unit yang telah diuji bekerja dengan baik, maka pada tahap ini akan digabungkan menjadi satu sistem sesuai dengan perencanaan.

3.7 Pengujian Sistem

Pada tahap ini dilakukan pengujian keseluruhan sistem yang telah dibuat mengacu pada parameter, skenario, dan metode pengujian yang telah ditentukan.

3.7.1 Parameter Pengujian

Parameter pengujian yang akan diukur adalah sebagai berikut:

- Kapasitas baterai
- Tegangan baterai
- Arus
- Waktu

3.7.2 Skenario Pengujian

Baterai akan diisi dan dikosongkan selama jumlah siklus yang telah ditentukan.

Pengujian akan dilakukan dalam kondisi berikut:

1. Baterai akan diisi hingga *SOC* mencapai lebih dari atau sama dengan 90%.
2. Baterai akan dikosongkan hingga *SOC* mencapai kurang dari atau sama dengan 10%.

3.7.3 Metode Pengujian

Pengujian akan dilakukan dengan menggunakan baterai yang terpasang pada motor BLDC. Pengisian dan pengosongan baterai akan dikendalikan oleh model sistem yang telah dirancang.

3.8 Analisis Hasil

Tahap ini bertujuan untuk menganalisis hasil pengisian dan pengosongan baterai yang berdasarkan pemantauan nilai *SOC*

3.9 Kesimpulan

Tahap ini bertujuan untuk menarik kesimpulan dari penelitian. Kesimpulan

dapat berupa hasil optimasi, evaluasi dan rekomendasi untuk penelitian selanjutnya.

3.10 Lokasi Penelitian

Kegiatan penelitian ini akan dilaksanakan di Laboratorium Teknik Elektro, Fakultas Teknik, Universitas Siliwangi, Tasikmalaya.

3.11 Objek Penelitian

Objek penelitian yang akan digunakan adalah Baterai LiFePo₄ 48V, 100Ah.