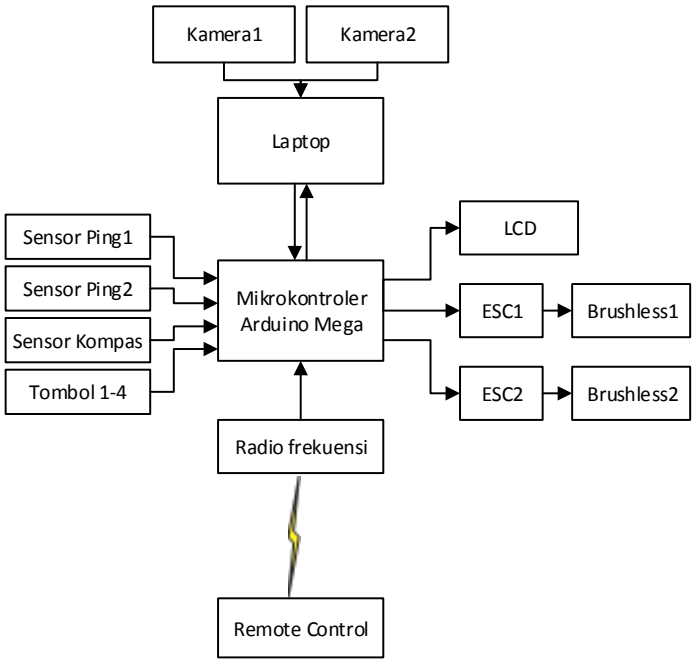
BAB III
PERANCANGAN

Bab ini membahas semua perancangan sistem dari yang dibuat guna memenuhi maksud dan tujuan penelitian tugas akhir yaitu studi komparasi navigasi robot kapal berbasis sensor ultrasonik, kamera dan *dead reckoning*. Berikut ini adalah diagram blok sistem secara keseluruhan:



Gambar III. 1 Diagram blok sistem.

Penjelasan lengkap setiap blok di atas adalah sebagai berikut.

1. Kamera

   Kamera yang digunakan dalam penelitian ini adalah jenis USB kamera Logitech C310, dimana kamera digunakan untuk melihat dan menangkap objek.

   a. Kamera1 berada pada sisi kiri kapal untuk menangkap objek bola berwarna yang berada pada posisi sisi kiri kapal.

b. Kamera2 berada pada sisi kanan kapal untuk menangkap objek yang bola berwarna yang berada pada posisi kanan kapal.

2. Sensor Kompas

Kompas yang digunakan pada kapal adalah Dt-Sense 3 axis-compass di mana kompas berperan dalam pembacaan orientasi kapal sebagai referensi kendali PID agar kapal dapat bergerak lurus pada sudut tertentu, dan kembali ke tempat kapal mulai bergerak.

3. Sensor Ultrasonik

Sensorping ultrasonik yang digunakan pada penelitian ini adalah jenis *Devantech* SRF04 *Ultrasonic Range Finder* sebagai referensi jarak terhadap objek, di mana objek yang digunakan disini adalah bola berwarna.

a. Sensor ultrasonik1 berada pada sisi kiri kapal untuk mendeteksi objek di sebelah kiri kapal.
b. Sensor ultrasonik2 berada pada sisi kanan kapal untuk mendeteksi objek di sebelah kanan kapal.

4. Mikrokontroler

Mikrokontroler berfungsi untuk mengolah data sensor-sensor, Radio Frekuensi, menerima dan mengirimkan data dari/ke laptop serta mengontrol *ESC* untuk mengatur kecepatan putaran motor *brushless*.

5. *ESC*

ESC (*Electronic Speed Control*) berfungsi untuk mengontrol putaran motor *Brushless.*

6. Motor *Brushless*

Motor *Brushless* berfungsi untuk menggerakkan *propeller*, agar kapal dapat bergerak maju.

7. *Push Button*

Push button disini berperan untuk memilih apakah jenis mode yang akan dijalankan oleh robot kapal. Pada robot kapal ini ada tiga jenis mode kapal yang digunakan yaitu mode *speed test*, *maneuver test*, dan *remote control*.

8. *Liquid Crystal Display (LCD)*

LCD digunakan untuk menampilkan informasi-informasi kapal yang dibutuhkan seperti derajat kompas aktual kapal dan settingan mode yang digunakan.

9. PC/Laptop

PC/ Laptop berisi aplikasi *image processing* dan *dead reckoning* yang berperan untuk menerima, menyimpan dan menampilkan data yang diolah oleh mikrokontroler.

Keseluruhan bahasan bab ini dibagi menjadi perancangan mekanik, *hardware* dan *software*.

3.1 Rancangan Mekanik

Perancangan mekanik merupakan pembuatan sebuah bangun robot kapal yang mengikuti standard aturan KKCTBN 2013 dengan ketentuan:

- panjang maksimal kapal : 130 cm
- lebar maksimalkapal : 70 cm
- tinggi maksimal kapal : 90 cm

Kapal pada penelitian ini dibuat dengan mengikuti ketentuan yang sudah ditetapkan. Kapal ini dirancang agar mampu menampung membawa beban yang dimasukkan kedalam kapal dan pergerakan kapal tetap stabil.
Ukuran kapal yang akan dibuat adalah sebagai berikut:

- panjang maksimal kapal : 97 cm
- lebar maksimalkapal : 49 cm
- tinggi maksimal kapal : 27 cm

Dengan ukuran ini robot kapal sudah dapat menampung komponen elektronik, baterai, dan laptop.



Gambar III. 2 Gambar robot kapal tampak atas



Gambar III. 3 Gambar robot kapal tampak samping

3.2 Rancangan Perangkat Keras

3.2.1 Sensor Kompas

Pada implementasi sistem PID dari robot kapal ini menggunakan Modul Dt-Sense 3 axis-compass sebagai referensi sudut acuan. Dt-Sense 3 axis-compass merupakan suatu modul sensor medan magnet yang menggunakan IC HMC5883L produksi Honeywell. IC HMC5883L merupakan *chip* yang didesain untuk membaca medan magnet yang cocok untuk aplikasi penunjuk arah dan *magnetometry.*

Tabel III. 1 Konfigurasi PIN DT-Sense 3 Axis compass

Pin DT-Sense	Pin Arduino Mega
1	-
2	-
3	-
4	-
5	SDA 20
6	SCL 21
7	GND
8	Power 3.3 Volt

Spesifikasi dari sensor kompas dt-sense 3 axis adalah sebagai berikut:

1. Tegangan kerja 3,3V dan konsumsi arus rendah (hingga 100 µA).
2. Memiliki sensor *magnetoresistive* 3 sumbu.
3. Memiliki jangkauan pembacaan medan magnet sampai dengan ±8 Gauss dengan resolusi 5 milligaus.
4. Output rate maksimum sampai dengan 160 Hz (*Single Measurement Mode*).
5. Output rate 0,75 Hz sampai dengan 75 Hz (*Continous measurement mode*).
6. Antarmuka i²C.

3.2.2 Mikrokontroler

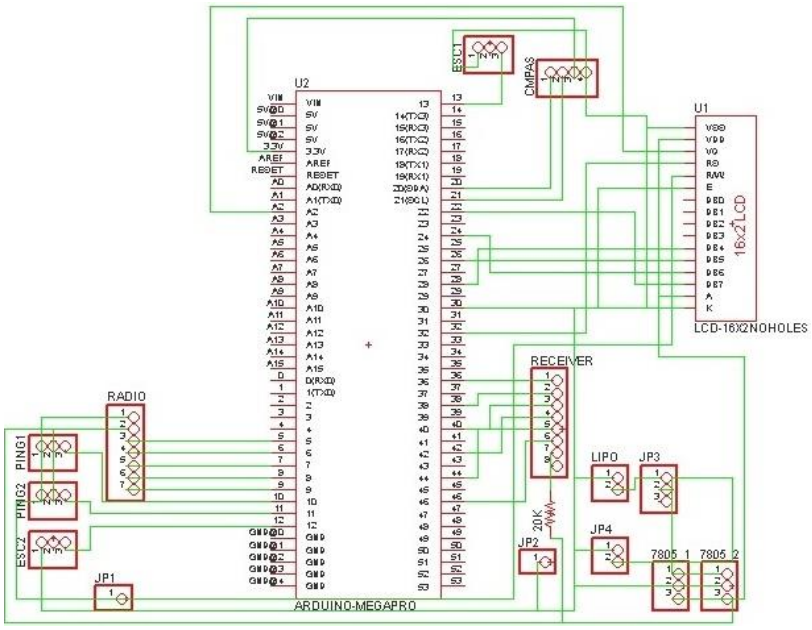
Mikrokontroler yang digunakan adalah mikrokontroler jenis Arduino Mega 2560. Mikrokontroler ini dapat bekerja dengan baik pada kondisi adanya goncangan. Skema penggunaan pin-pin *input output* pada mikrokontroler ditunjukkan pada Tabel III-2

Tabel III. 2 Konfigurasi penggunaan pin pada mikrokontroler Arduino Mega 2560

PIN	Keterangan
D13	*ESC* kanan
D12	*ESC* kiri
D11	Ping kanan
D10	Ping Kiri
D9	Ch1
D8	Ch2
D7	Ch3
D6	Ch4
D5	Ch5
D20	SDA
D21	SCL
D22	DB7
D24	DB6
D26	DB5
D28	DB4
D30	E
D34	RS
D36	Push button1
D38	Push button2
D40	Push button3
D42	Push button4

3.2.3 Skematik Rangkaian

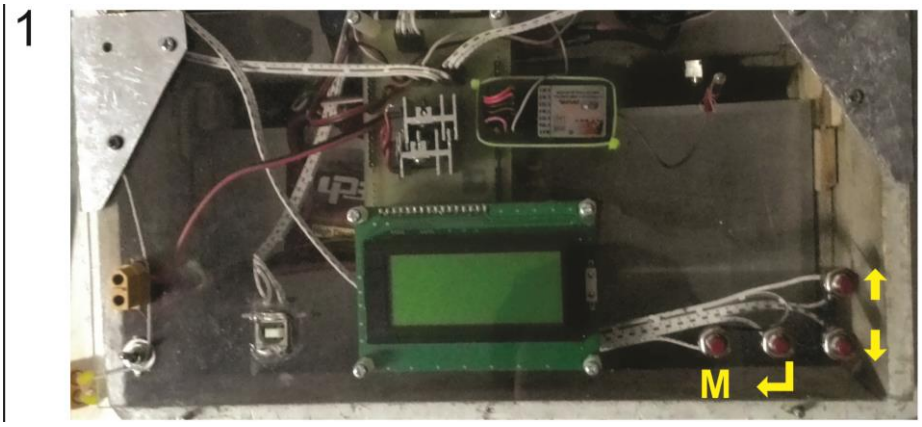
Dalam pengerjaan robot kapal ini maka dibutuhkan beberapa komponen pendukung seperti Arduino Mega 2560, sensor ultrasonik, radio, *ESC*, dan rangkaian regulator. Maka dibuatlah skematik rangkaian untuk mempermudah dalam perancangan yang ditunjukkan pada Gambar III-4.



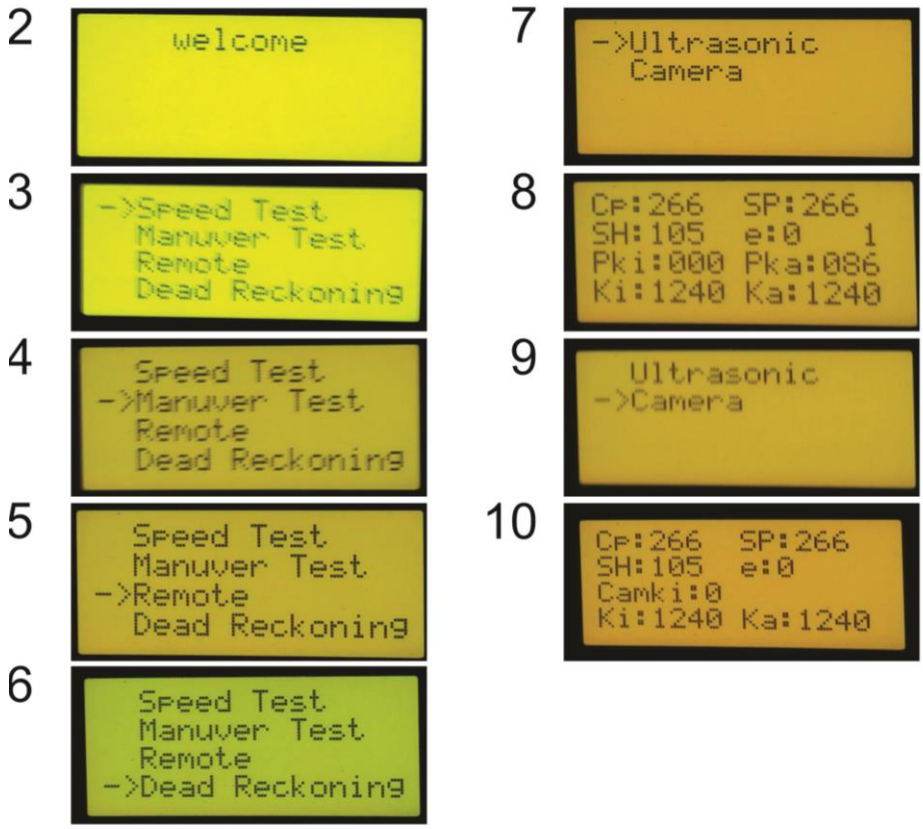
Gambar III. 4 Gambar skematik rangkaian.

3.3 Rancangan Tampilan Menu Program

Untuk memudahkan dalam pengoperasian robot kapal ini maka dirancang program tampilan. Program tampilan yang telah dibuat akan diperlihatkan pada gambar berikut:



Gambar III. 5 Penempatan LCD dan push button



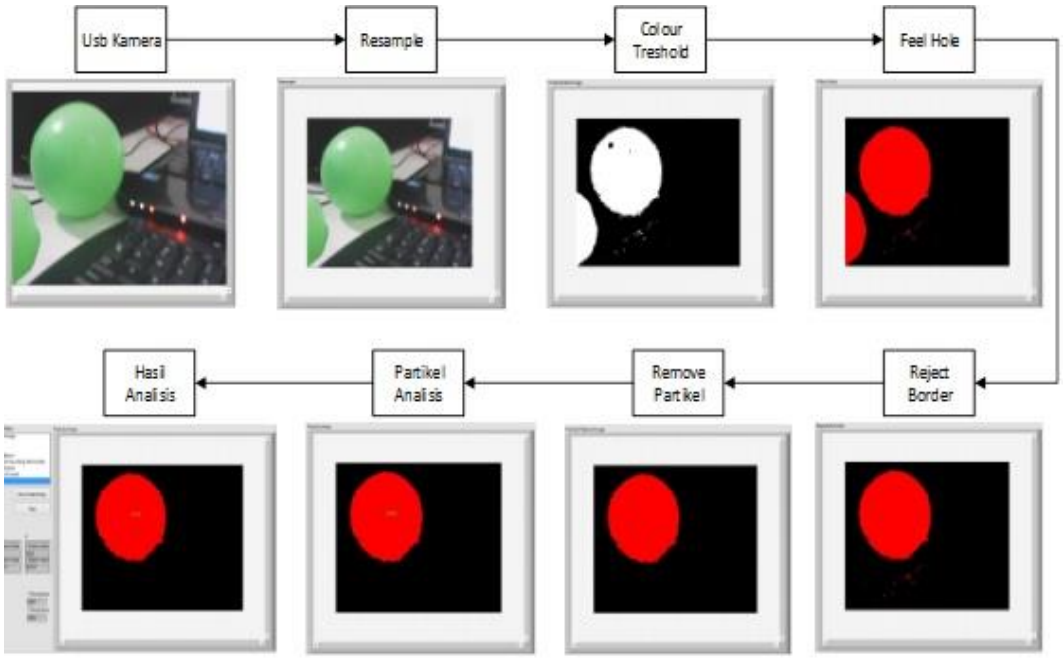
Gambar III. 6 Tampilan menu program lcd 4x16 pada robot kapal.

Pada Gambar III-5 ditunjukkan isi dari menu program, pada saat pertama kali saklar pada posisi "ON" maka akan tampil salam pembuka "*welcome*" (no.2 Gambar III-5) kemudian apabila kita menekan push button M maka akan masuk pada sub menu berikutnya, terlihat ada beberapa sub menu yaitu *speed test*, *maneuver test*, *remote*, dan *dead reckoning* (no.2 Gambar III-5). Jika kita hendak masuk pada mode *speed test*, ketika kursor berada pada tulisan *speed test* (no.3 Gambar III-5) kemudian kita tekan *enter* maka akan masuk pada tampilan berikutnya yaitu tampilan mode pilihan menggunakan sensor ultrasonik atau kamera (no.7 dan 9 Gambar III-5). Bila kita memilih ultrasonik maka akan muncul tampilan terakhir yang ditunjjukkan pada no.8 Gambar III-5.

3.4 Rancangan Perangkat Lunak

3.4.1 Deteksi Objek

Objek yang ditangkap oleh kamera diproses menggunakan *software* LabVIEW dengan alur sebagai berikut:



Gambar III. 7 Deteksi objek menggunakan LabVIEW 2012

Penjelasan lengkap setiap blok Gambar III-7 adalah:

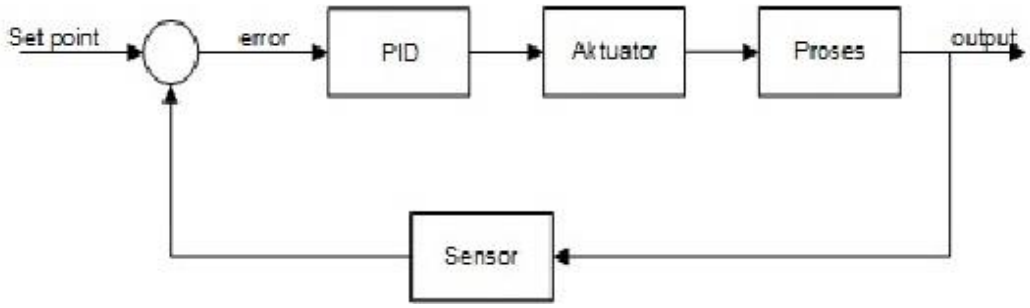
1. USB kamera berfungsi untuk menangkap citra objek yang akan diproses.
2. Ukuran objek yang ditangkap oleh USB kamera akan diubah dengan menggunakan *resample* sesuai dengan ukuran objek yang diinginkan.
3. *Colour Treshold Hue Saturation Value (HSV)* akan menyaring warna objek yang sudah diproses *resample.* Ini dilakukan agar didapatkan warna objek yang diinginkan. Pada proses ini akan terjadi proses penghapusan beberapa pixel yang tidak sesuai dengan nilai warna yang diinginkan sebelumnya.
4. *Fill Hole* berfungsi untuk menutup lubang yang diakibatkan oleh proses *Colour Treshold HSV* ini dilakukan agar warna tetap utuh.
5. *Reject Border* berfungsi untuk menghapus semua partikel yang masih tersisa atau menempel pada border.

6. *Remove Partical* berfungsi untuk menghapus partikel-pertikel yang lebih kecil dari objek yang diinginkan sehingga hanya partikel yang diinginkan yang tersisa.
7. *Partical Analisis* berfungsi untuk menganalisis partikel yang tersisa apakah sesuai dengan ukuran partikel yang sudah ditentukan sebelumnya.

3.4.2 Pengontrolan *Proporsional Integral Derivative*

Sistem kontrol merupakan proses pengendalian *error* dengan cara memasukkan *error* tersebut ke dalam *input* yang akan dibandingkan dengan sistem pengendalian. Tujuan dari PID untuk menghasilkan *output* atau keluaran yang sesuai dengan *set point* yang diberikan dengan cara mengurangi *error* tersebut. Pengontrolan *PID* menggunakan *close loop* atau umpan balik, yaitu program diolah pada mikrokontroler, lalu menjalankan aktuator setelah itu mengeluarkan *output.* Keluaran atau *output* akan dibandingkan dengan sensor agar dapat mencapai *set point* yang diinginkan.

Di bawah ini diperlihatkan gambar blok diagram pengontrol *PID* dengan *close loop.*



Gambar III. 8 Pengontrolan PID dengan Close Loop.

PID dapat juga digambarkan dengan persamaan

$u(t) = Kp\ e(t) + Ki \int_0^1 e(t)dt + Kd\ \frac{de(t)}{dt}$…… ……..pers (1)

Dengan:

$Ki = Kp\ x\ \frac{1}{Ti}\ dan\ Kd = Kp\ x\ Td$ ……pers(2)

Keterangan:

u(t) : *Out put* dari pengontrol *PID.*

Kp : *Gain proporsional*.

Ti : *Time integral.*

Td : *Time derivative.*

Ki : *Gain Integral*

Kd : *Gain derivative.*

Pengontrol *PID* merupakan paduan dari Pengontrol *Proporsional*, Pengontrol *Integral* dan Pengontrol *Derivatif*. Cara kerja dari masing-masing pengontrol adalah sebagai berikut:

1. Pengontrol *proporsional*

   Pengontrol *proporsional* memiliki keluaran yang sebanding/ *proporsional* dengan besarnya sinyal kesalahan.

   ciri-ciri pengontrol *proporsional*:

   a. Jika nilai Kp kecil, pengontrol *proporsional* hanya mampu melakukan koreksi kesalahan yang kecil, sehingga akan menghasilkan respon sistem yang lambat.
   b. Jika nilai Kp dinaikkan, respon sistem akan semakin cepat mencapai keadaan mantapnya.
   c. Jika nilai Kp diperbesar sehingga mencapai harga yang berlebihan, akan mengakibatkan sistem bekerja tidak stabil atau respon sistem akan berosilasi.
   d. Nilai Kp dapat diatur sehingga mengurangi *stady state error* tetapi tidak menghilangkannya.

2. Pengontrol *Integral* berfungsi untuk menghilangkan *stady state error*.

   Ciri-ciri dari pengontrol *integral* adalah sebagai berikut:

   a. Keluaran pengontrol *integral* membutuhkan selang waktu tertentu, sehingga pengontrol *integral* cenderung memperlambat respon.
   b. Keluaran pengontrol bertahan pada nilai sebelumnya, ketika sinyal kesalahan berharga nol.

c. Keluaran menunjukkan kenaikan atau penurunan yang dipengaruhi oleh besarnya sinyal kesalahan dan nilai Ki, jika sinyal kesalahan tidak berharga nol.

d. Ki yang berharga besar mempercepat hilangnya nilai *offset.* Tetapi semakin besar nilai Ki mengakibatkan peningkatan osilasi dari sinyal keluaran pengontrol.

3. Pengontrol *derivative* berfungsi untuk memperbaiki sekaligus mempercepat *respon transient.*

   Ciri-ciri pengontrol *derivative* adalah sebagai berikut:

   a. Pengontrol tidak dapat menghasilkan keluaran jika tidak ada perubahan pada *input* (berupa sinyal perubahan kesalahan).

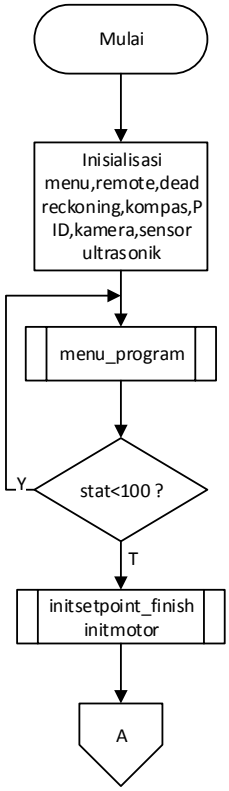
   b. Jika sinyal kesalahan berubah terhadap waktu, maka keluaran yang dihasilkan pengontrol tergantung pada nilai Kd dan laju perubahan sinyal kesalahan.

   c. Pengontrol diferensial mempunyai suatu karakter untuk mendahului, sehingga pembangkit kesalahan menjadi sangat besar. Jadi pengontrol diferensial dapat mengantisipasi pembangkit kesalahan, memberikan aksi yang bersifat *korektif* dan cenderung meningkatkan stabilitas sistem.

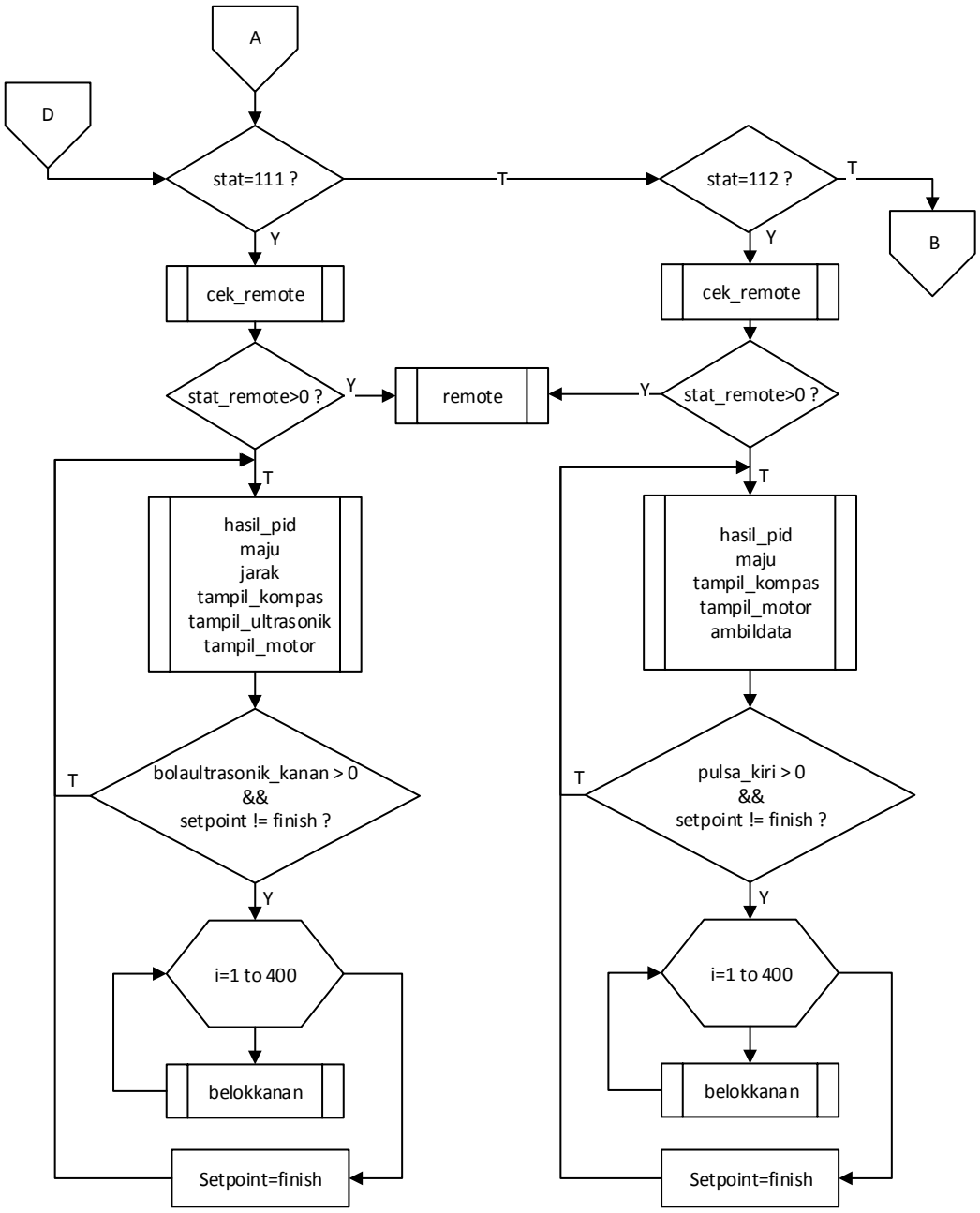
   d. Dengan meningkatkan nilai Kd, dapat meningkatkan stabilitas sistem dan mengurangi *overshoot.*

3.5 Rancangan Pengontrolan

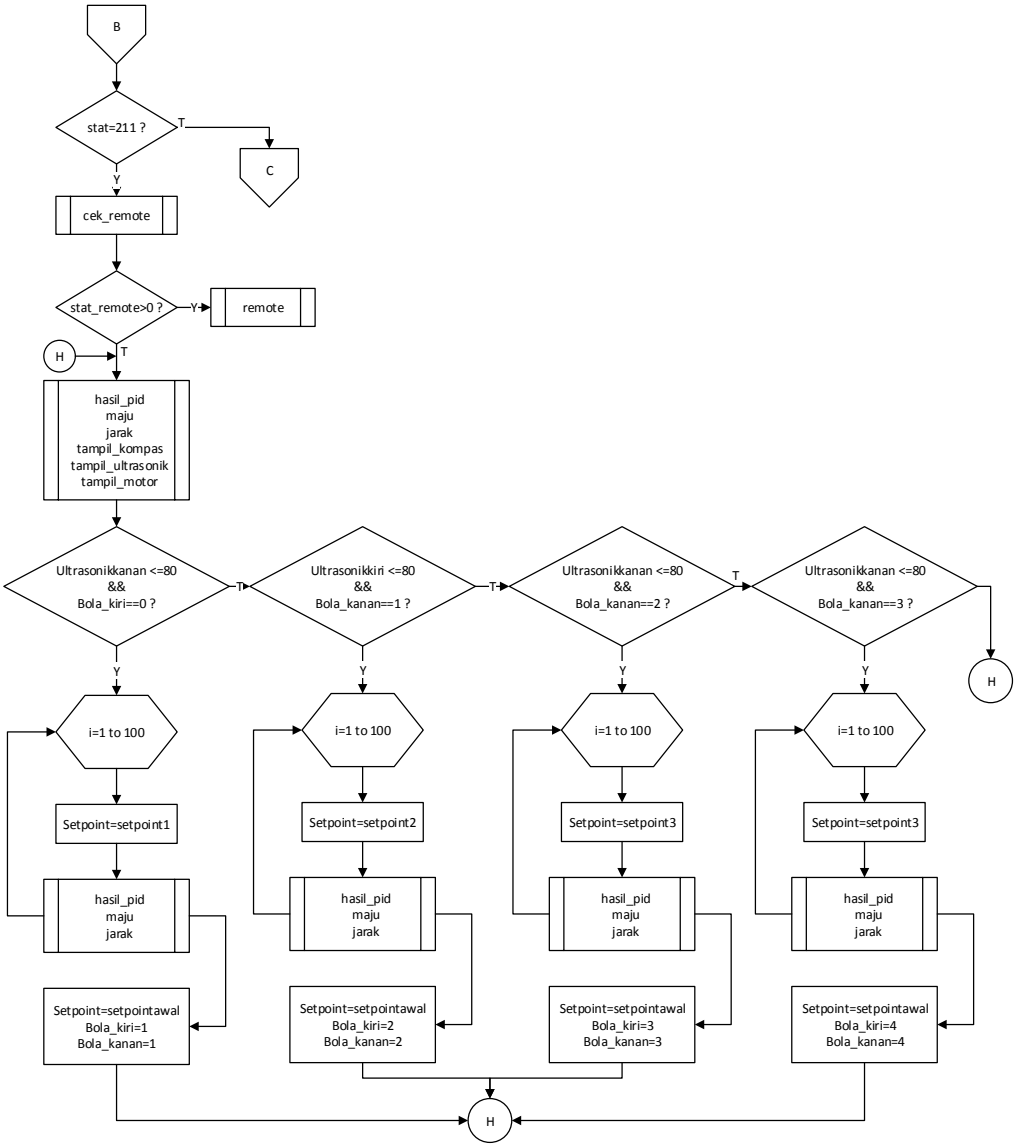
Berikut ini adalah diagram alir secara keseluruhan program yang ditanam pada arduino mega 2560.



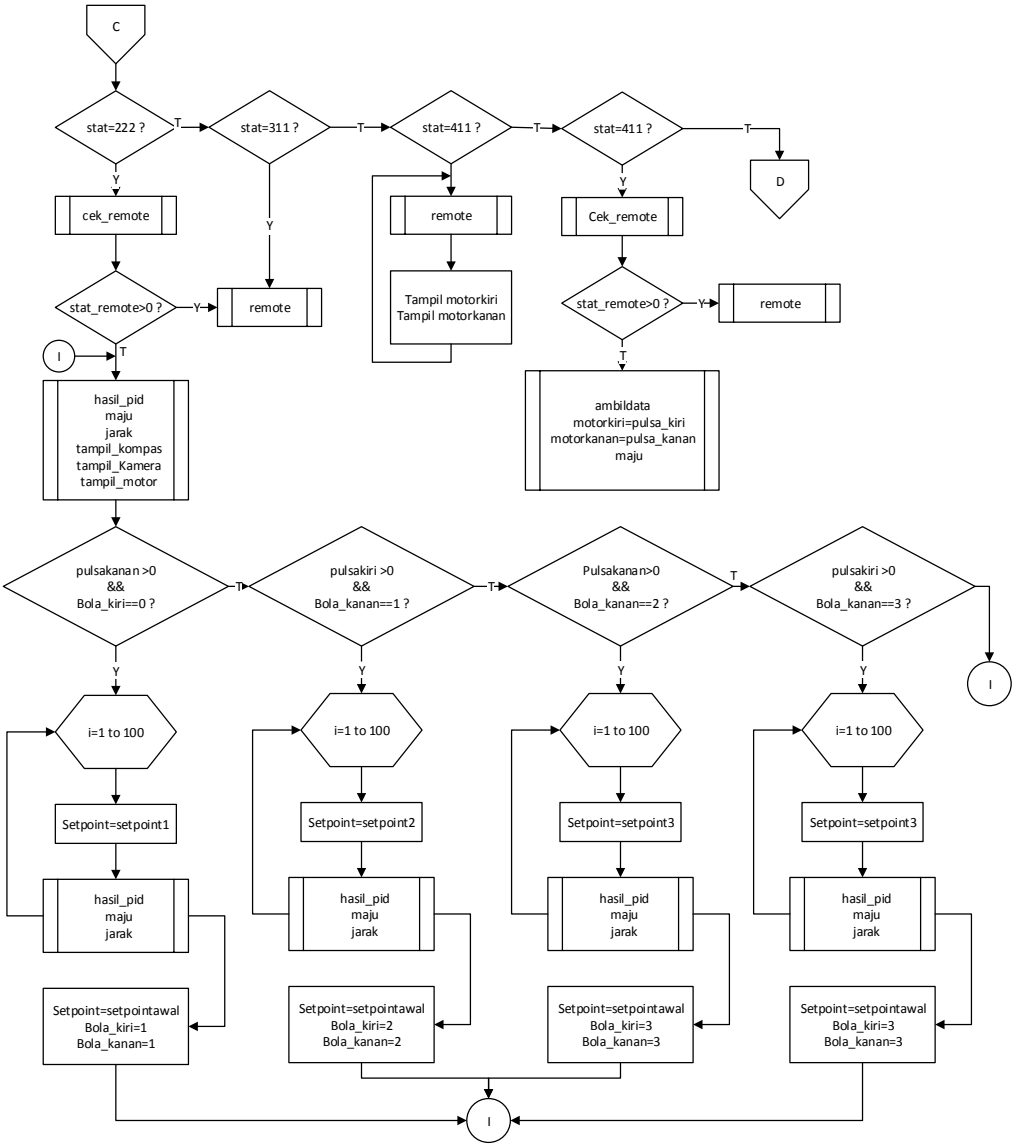
Gambar III. 9 Diagram alir program utama



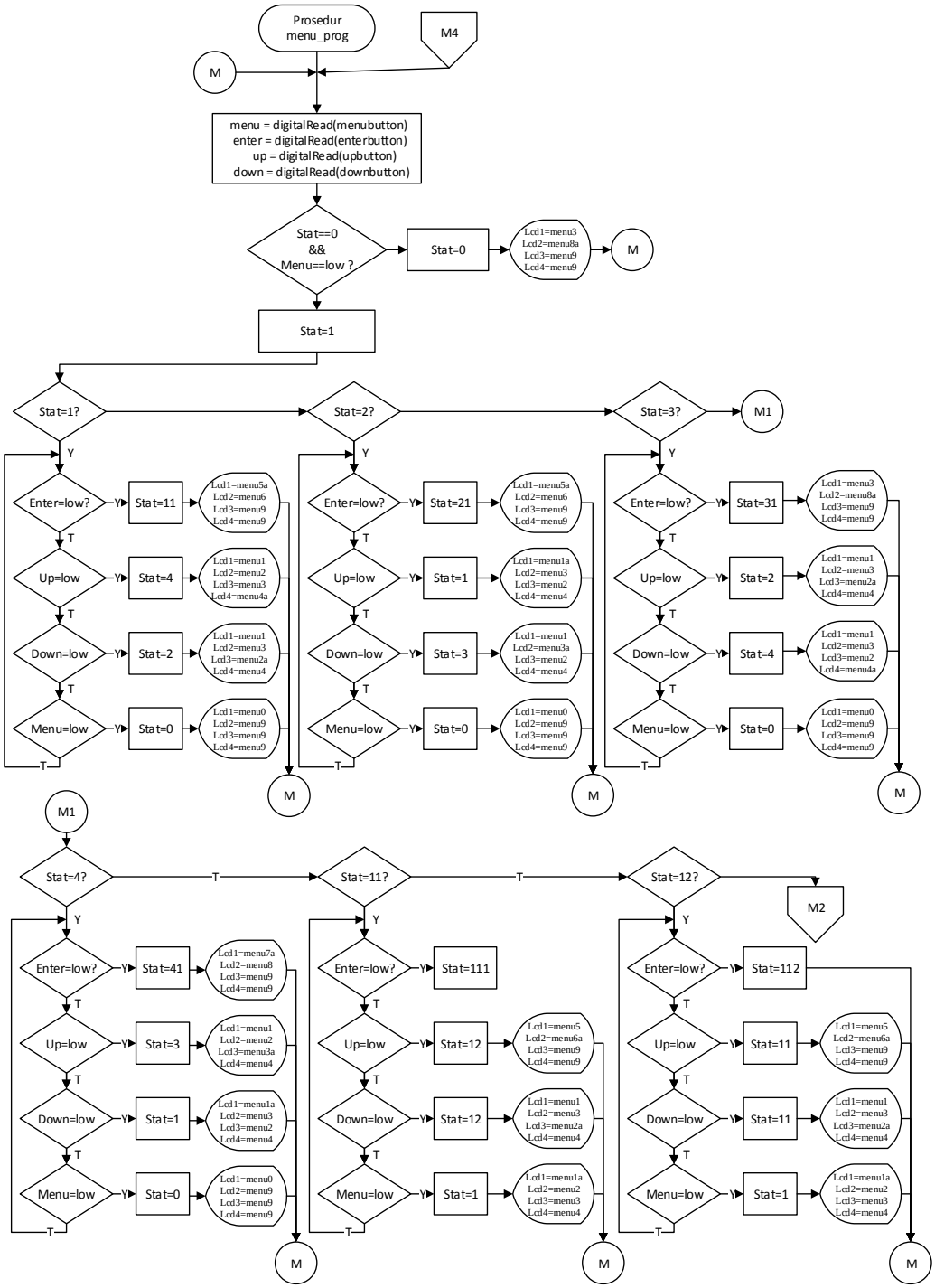
Gambar III. 10 Diagram alir program utama.



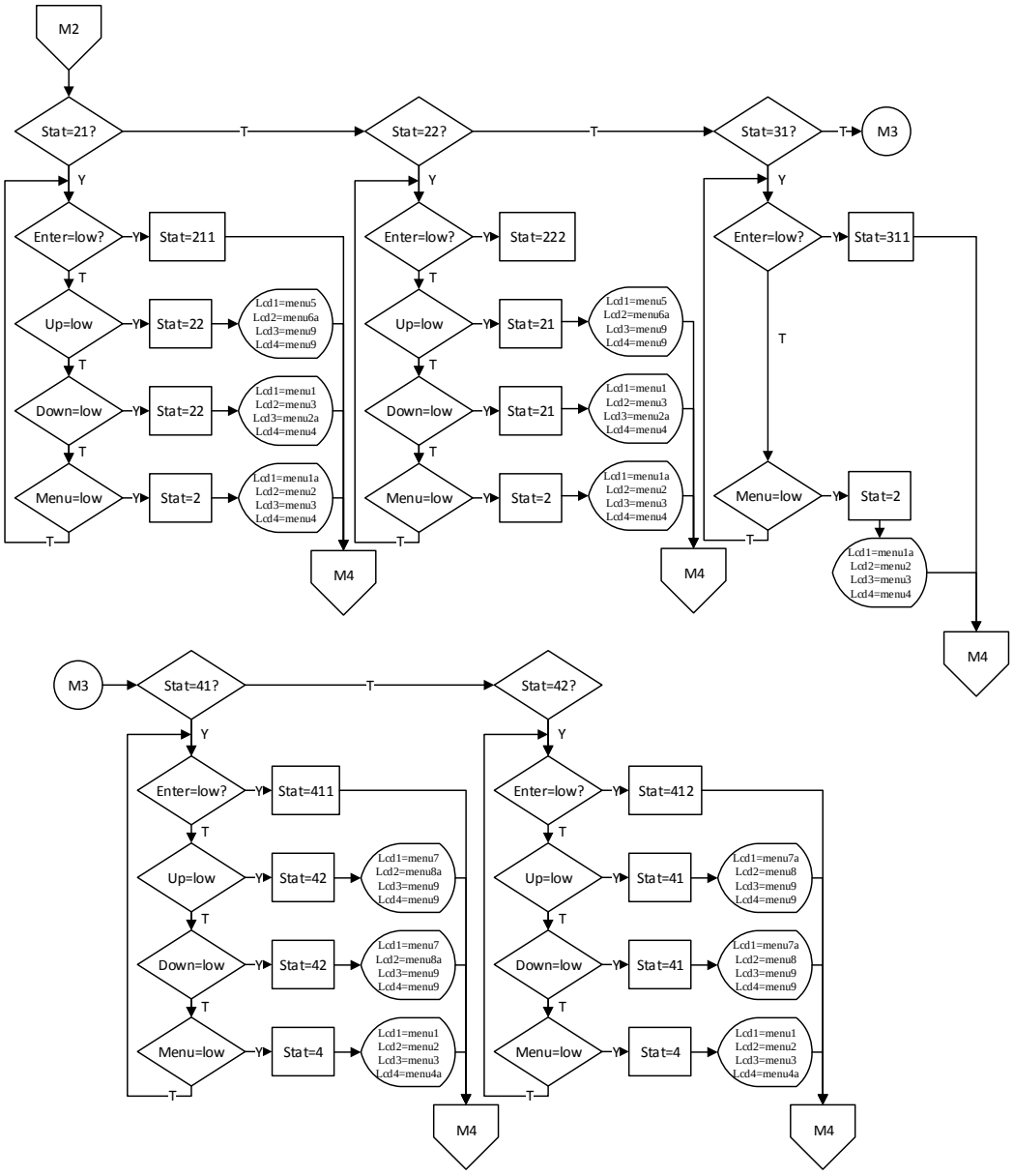
Gambar III. 11 Diagram alir program utama.



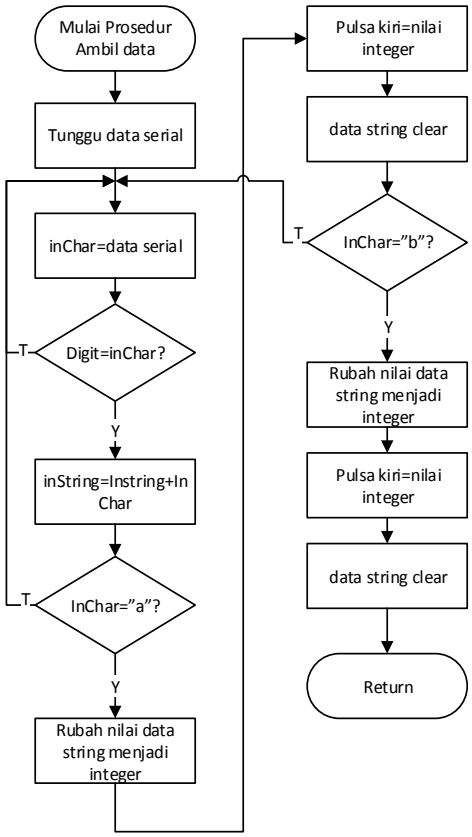
Gambar III. 12 Diagram alir program utama.



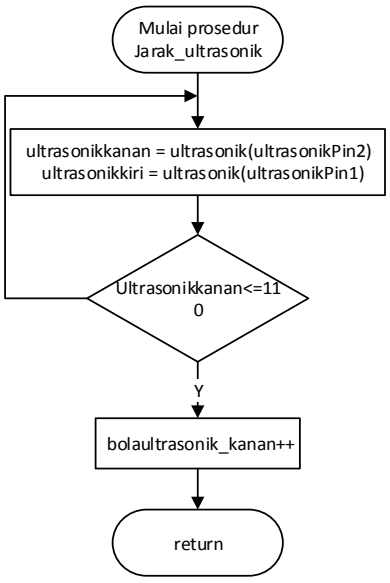
Gambar III. 13 Prosedur program menu



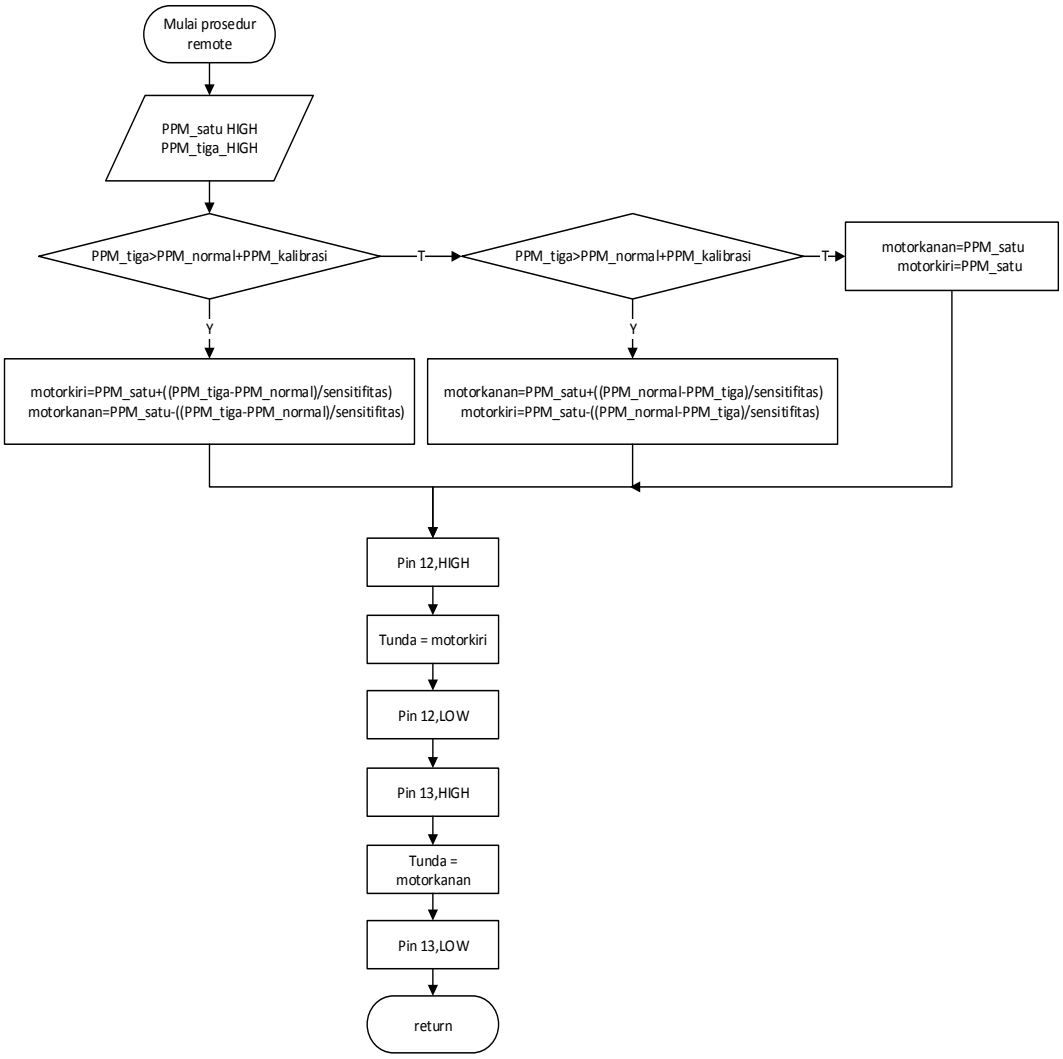
Gambar III. 14 Prosedur program menu



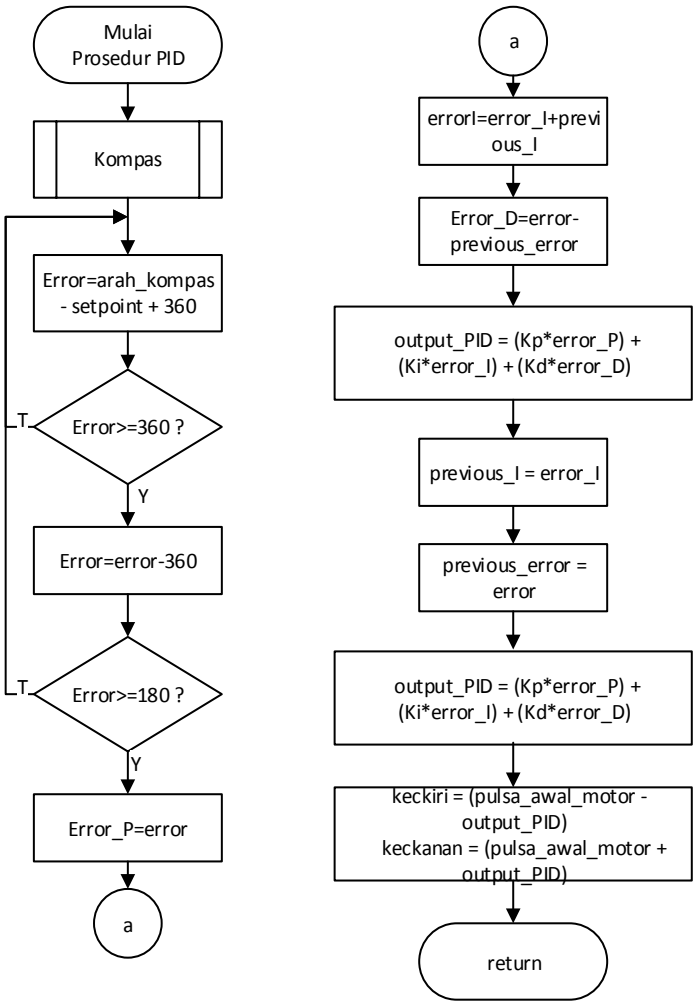
Gambar III. 15 Diagram alir prosedur ambil data



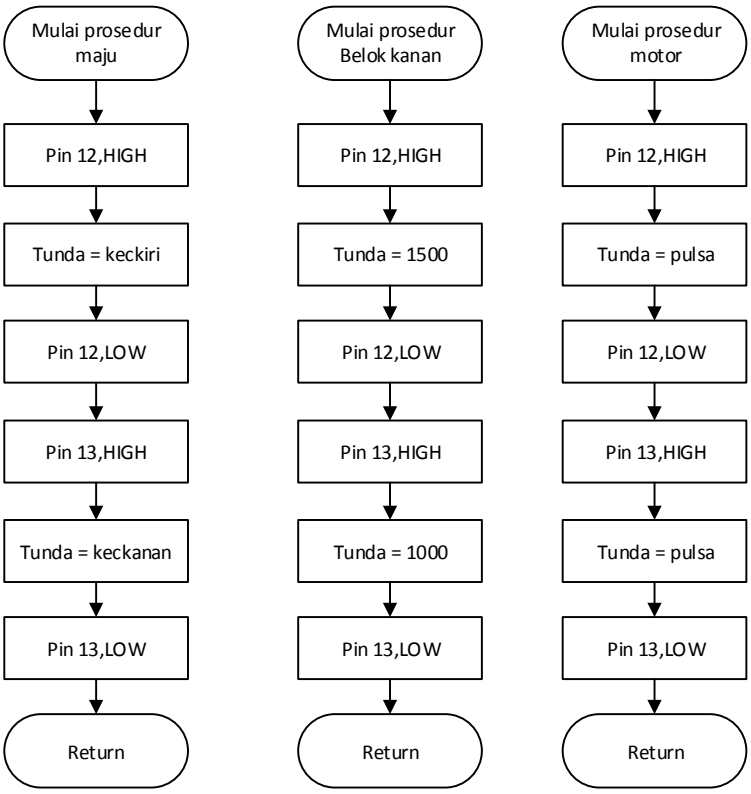
Gambar III. 16 Diagram alir prosedur jarak ultrasonik



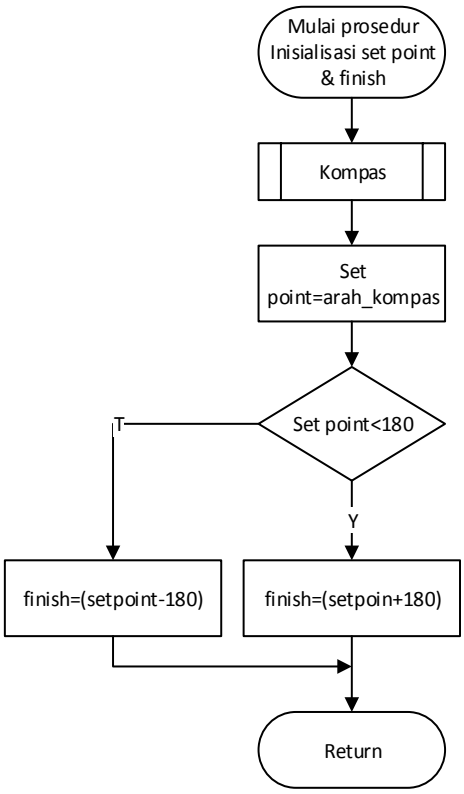
Gambar III. 17 Diagram alir prosedur remote



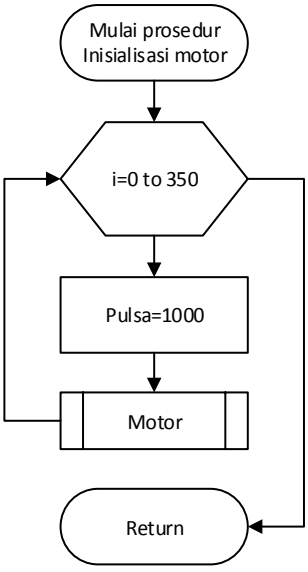
Gambar III. 18 Diagram alir prosedur PID



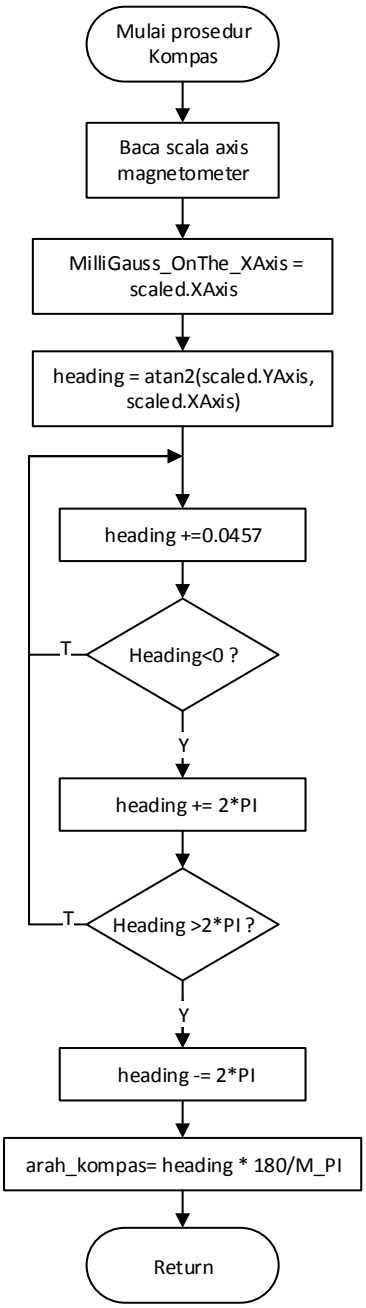
Gambar III. 19 Diagram alir prosedur maju, belok kanan dan motor



Gambar III. 20 Diagram alir prosedur set point dan finish



Gambar III. 21 Diagram alir prosedur inisialisasi motor



Gambar III. 22 Diagram alir prosedur kompas