

BAB III

PERANCANGAN SISTEM

3.1. Tujuan Perancangan

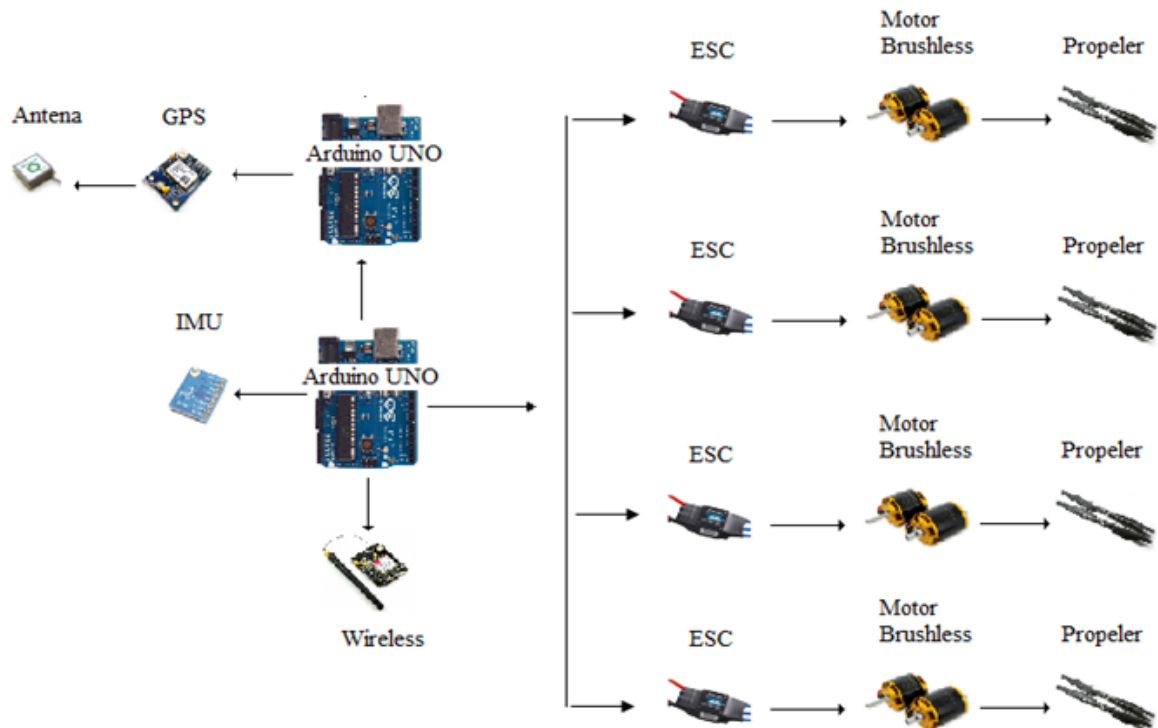
Perancangan adalah tahap terpenting dalam seluruh proses pembuatan alat. Tahap pertama yang paling penting dalam perancangan adalah membuat diagram blok rangkaian, kemudian memilih komponen dengan karakteristik yang sesuai dengan kebutuhan. Untuk pemilihan komponen ini diperlukan data book serta petunjuk lain yang dapat membantu dalam mengetahui spesifikasi dari komponen tersebut sehingga komponen yang didapat merupakan pilihan yang tepat bagi alat yang akan dibuat.

Perancangan alat ini mempunyai tujuan untuk mendapatkan hasil akhir yang baik seperti yang diharapkan dengan memperhatikan penggunaan komponen yang murah serta mudah didapatkan di pasaran. Selain itu dengan adanya perancangan tersebut akan mempermudah kita mencari dan memperbaiki kerusakan peralatan atau rangkaian tersebut. Dengan adanya perancangan yang baik, maka didapatkan suatu alat yang sesuai dengan keinginan dari perancang alat itu sendiri.

Dalam pembuatan alat ini terdapat beberapa langkah perencanaan dan perancangan yang saling berkaitan satu dengan yang lainnya. Secara garis besar, langkah-langkah perencanaan dan perancangan terdiri atas dua bagian yaitu perancangan hardware dan perancangan software.

Berikut ini adalah blok diagram dari Sistem Navigasi *Waypoint* pada *Quadcopter*:

- *Quadcopter*



Gambar 3.1 Blok Diagram *Quadcopter*

- Segment pengontrol



Gambar 3.2 Blok Diagram Segment Pengontrol

Dengan adanya diagram alat maka perencanaan dan pembuatan perangkat keras atau *hardware* terdiri dari beberapa rangkaian, yaitu:

a. *Quadcopter* :

1. Sensor IMU berfungsi sebagai penyetabil *quadcopter*.
2. Modul chip GPS dan antenna sebagai receiver data koordinat posisi dari satelit GPS.
3. Arduino UNO sebagai pusat pengendali dan pemrosesan data pada sensor.
4. Rangkaian driver motor (ESC) dan motor brushless untuk menggerakkan putaran propeler *quadcopter* agar dapat terbang.
5. Modul wireless untuk mengirimkan data dari *quadcopter* ke ground segment dan sebaliknya.

b. Penampil Data :

1. Modul wireless berfungsi menerima data yang dikirimkan dari *quadcopter* secara nirkabel.
2. PC yang didalamnya telah terdapat program GUI (*Graphic User Interface*) berfungsi sebagai penampil data secara *real time*.

3.2. Alat dan Bahan

Dalam melakukan penelitian ini digunakan beberapa alat dan bahan sebagai berikut :

a. Perangkat lunak:

1. Software arduino untuk memprogram arduino board.
2. System operasi Windows 7 64 bit.
3. Microsoft office.
4. Google sketchup, dan
5. Layanan internet.

b. Perangkat keras:

1. Frame *Quadcopter* tipe S500.
2. ESC 30 A.
3. Motor brushless 920 KV.

4. Propeler.
5. Baterai LI-PO 3000 mAh.
6. Arduino board type uno.
7. Modul GPS Ublox NEO-7m.
8. Modul Radio Wireless
9. Laptop Sony.
10. Solder.
11. Timah.
12. Multimeter.
13. Obeng.

3.3. Perancangan Sistem

Pada penelitian ini, dirancang sebuah sistem navigasi *waypoint* pada *quadcopter* menggunakan GPS. *Quadcopter* akan melalui jalur yang dibuat dari titik-titik koordinat pada *google maps* yang akan ditanamkan pada modul arduino. program *Arduino compiler*. Data-data koordinat tersebut akan dibaca oleh modul GPS, kemudian *quadcopter* akan bernavigasi menuju titik-titik yang telah tertulis pada *scetch* program yang *diupload* pada arduino tersebut.

Metode perancangan ini terbagi menjadi 2, yaitu:

1. Perancangan perangkat keras (*Hardware*)
2. Perancangan perangkat lunak (*Software*)

3.3.1. Perancangan Perangkat Keras (*Hardware*)

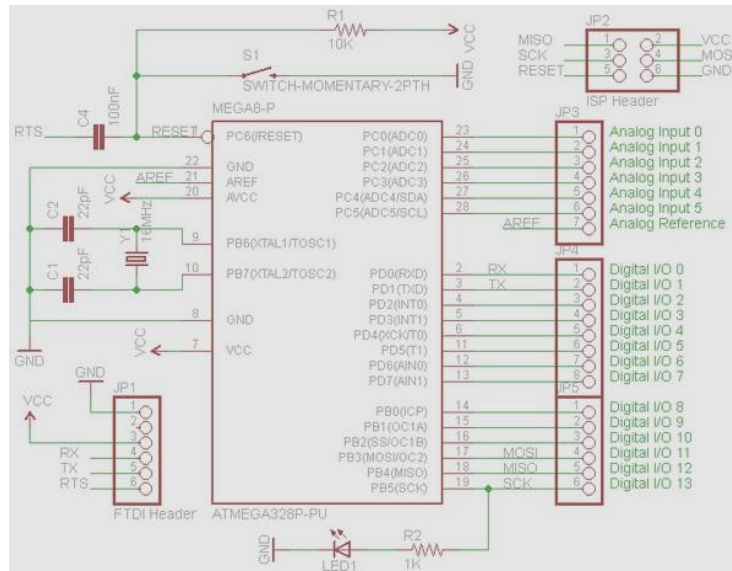
Perancangan Hardware terdiri dari beberapa jenis komponen elektronika. Untuk perancangan ini terdiri dari rangkaian power supply, Sensor GPS, Arduino Uno Atmega328, Driver Motor *Brushless (ESC)*, Motor Brushless, dan sensor IMU.

Pada perancangan *quadcopter* diperlukan rangkaian-rangkaian elektronika yang menunjang sistem kerja dari *quadcopter*. Berikut merupakan

rangkaian-rangkaian yang digunakan pada Sistem Navigasi Waypoint pada Quadcopter.

a. Mikrokontroler ATmega328 (Arduino Uno)

Mikrokontroler yang digunakan dalam sistem ini adalah mikrokontroler dengan jenis AVR Seri ATMEGA328. Mikrokontroler ini mempunyai 16 port(A,B,C,D) yang bisa dijadikan port/pin *input* dan *output*. Pemilihan ATMEGA ini diperkirakan akan memaksimalkan pembuatan alat sebagai pengolah data.



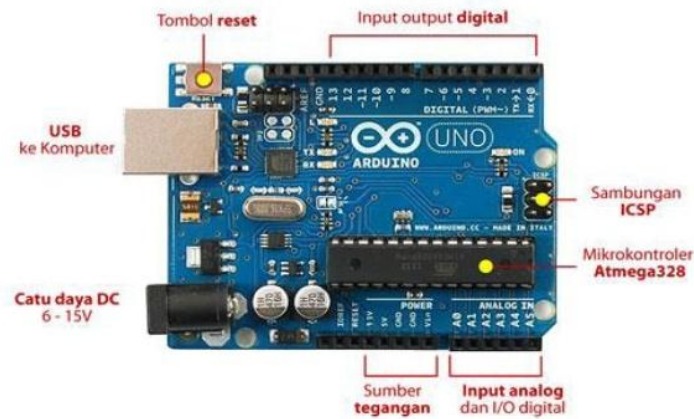
Gambar 3.3Rangkaian ATMEGA328P

(<https://klyx.wordpress.com/2012/09/20/simpl-arduino/>)

Dalam perancangan Arduino Uno, selain mikrokontroler ATmega328, terdapat juga IC ATmega16U2 yang berperan sebagai pengubah dari USB ke serial dalam komunikasi dengan komputer yang dimuat dengan sebuah bootloader FDU. Peran IC ATmega16U2 ini menggantikan ATmega8U2.

Arduino Uno beroperasi pada tegangan 5 volt. Namun, tegangan input yang disarankan pada board Arduino adalah 7 hingga 11 volt. Sedangkan tegangan input batasnya adalah 6 sampai 20 volt.

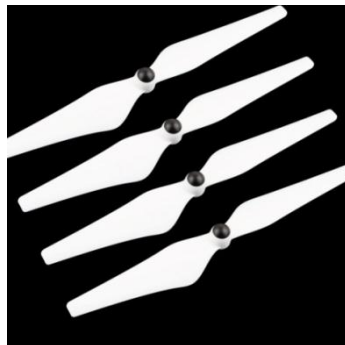
Gambar di bawah ini merupakan gambar fungsi dan bagian yang akan digunakan pada perancangan mulai dari catu daya, koneksi arduino dengan komputer, input dan output, serta mikrokontroller itu sendiri.



Gambar 3.4 Board Arduino Uno dan fungsi masing-masing pin

b. Propeller

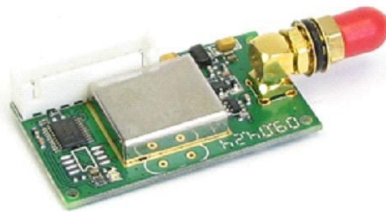
Propeller adalah baling yang digunakan untuk menggerakkan *quadcopter*. Propeller ini akan menghasilkan gaya angkat pada *quadcopter* sehingga *quadcopter* dapat terbang. Propeller yang digunakan menurut arah putarannya ada 2 jenis, yakni CW (*clock wise*) atau searah jarum jam dan CCW (*counter clock wise*) atau berlawanan arah jarum jam. Pada *quadcopter* ini, propeler yang digunakan adalah propeller DJI 9450 L. propeller ini memiliki ukuran 9.4x5” dengan bahan *carbon fiber*.



Gambar 3.5 Propeller DJI 9450 L

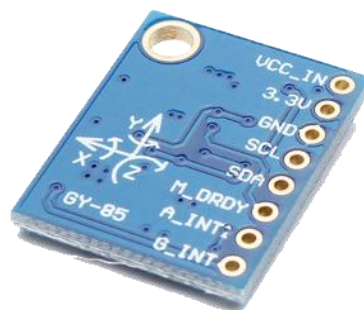
c. Wireless

Komunikasi *wireless* adalah komunikasi serial searah maupun dua arah yang berfungsi untuk pengiriman data dengan menggunakan frekuensi radio. Modul *wireless* berfungsi untuk mengirimkan data koordinat yang akan dituju dari *Ground Segment Control* ke *quadcopter* untuk bisa menjalankan sistem navigasi *waypoint*. Sedangkan pengiriman dari *quadcopter* ke *Ground Control Segment* adalah pengiriman data sensor dan data GPS untuk mengetahui posisi dan sikap *quadcopter*. Modul *wireless* yang digunakan dalam perancangan ini adalah modul RF dengan tipe 3DR radio *wireless*. Modul RF ini memiliki spesifikasi sebagai berikut :



Gambar 3.6 Modul *Wireless*

d. Sensor Gyro L3G4200D



Gambar 3.7 Sensor Gyro L3G4200D

Pada perancangan ini akan digunakan sensor *Gyroscope L3G4200D* yang memiliki pendeteksi navigasi sudut kemiringan (*Z-axis respon*) dengan tingkat penyimpangan dari keadaan lurus seimbang (*pivot point*) sebesar $\pm 300^\circ/\text{sec}$, mengeluarkan tegangan yang nilainya sebanding dengan nilai kecepatan sudut, nilai tersebut/ditentukan oleh nilai kepekaannya. Nilai kepekaan tersebut memiliki satuan milivolt per derajat per detik ($\text{mV}/^\circ/\text{s}$). *Gyroscope L3GD4200D* memiliki nilai kepekaan antara 5,52 $\text{mV}/^\circ/\text{s}$ hingga 6,48 $\text{mV}/^\circ/\text{s}$.

Pada rangkain sensor Gyroscope L3G4200D di atas terdapat 8 pin masukan, tapi di perancangannya hanya digunakan 4 pin, yaitu 3,3 V, GND, SCL dan SDA dengan ketentuan bahwa Pin 3,3 V sebagai tegangan masukan untuk L3G4200D, GND sebagai ground kemudian pin SCL dan SDA sebagai masukan analog untuk Arduino Uno.

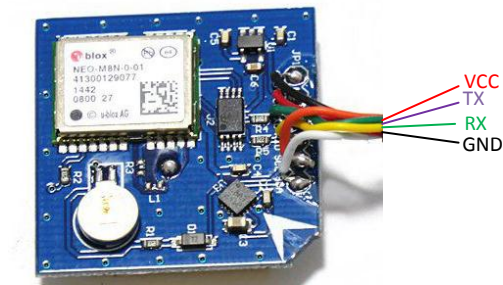
e. Sensor Kompas HMC5883L



Gambar 3.8 Sensor Kompas HMC5883L

Pada perancangan ini akan digunakan sensor kompas HMC5883L yang dapat membaca arah mata angin. Sensor ini pada dasarnya dapat membaca medan magnet dengan sumbu *3-axis*. Sensor magnetometer HMC5883L disebut juga kompas digital karena menggunakan komponen utama berupa IC HMC5883 yang merupakan IC kompas digital 3 axis yang memiliki interface berupa 2 pin I2C.

f. Modul GPS Ublox NEO-7m



Gambar 3.9 GPS Ublox Neo-7m ([http:// alexnld.com](http://alexnld.com))

Agar *quadcopter* dapat bernavigasi secara *autonomous waypoint*, perancangan ini menggunakan modul GPS Ublox Neo-7m. Modul GPS berfungsi sebagai pemberi informasi titik koordinat dari *quadcopter* sehingga *quadcopter* dapat bergerak secara *autonomous* dengan mengikuti titik *waypoint* yang diambil pada *google map*. *Ground Control Segment* akan mengirimkan informasi koordinat dari titik-titik yang telah dibuat ke *quadcopter* sehingga *quadcopter* dapat bernavigasi mengikuti titik *waypoint* tersebut. Modul GPS Ublox Neo-7m ini memiliki sensitivitas *tracking* sebesar -163 dBm dan memiliki akurasi sebesar <2.5 m dari titik yang dituju.

Modul GPS Ublox Neo-7m ini memiliki 6 pin. Namun yang digunakan dalam perancangan ini adalah 4 pin, yakni pin Tx, Rx, VCC, dan GND. Modul ini menggunakan supply dengan rentang 4.5-6 volt. Namun, tegangan rekomendasi agar modul dapat bekerja dengan maksimal yakni sebesar 5 volt.

g. Motor Brushless

Motor brushless adalah motor DC yang digunakan pada perancangan sistem ini untuk menerbangkan *quadcopter*. Berdasarkan arah putarannya, brushless yang digunakan ada 2, yakni dengan arah putar CW dan CCW. Motor brushless yang digunakan pada sistem ini adalah DJI 2212 920 KV dengan spesifikasi:

- KV(rpm/v): 920.
- Max Power: 370W.
- Max Thrust: 1200 grams.
- Weight: 53 grams.
- Shaft Diameter: 4mm.
- Shaft Length: 49mm



Gambar 3.10 Motor Brushless DJI 2212 920 KV

h. Driver motor brushless

Motor Brushless dapat bekerja saat diberi tegangan. Pada perancangan robot beroda, tegangan yang didapatkan oleh Motor Brushless berasal dari sinyal yang berupa logika yang dihasilkan oleh mikrokontroler ATmega328 yang diberikan ke pengendali motor. Pada sistem ini menggunakan ESC (*Electrical Speed Control*) dengan supply 30 A.



Gambar 3.11 ESC 30 A

Motor DC tidak dapat dikendalikan secara langsung oleh mikrokontroler karena kebutuhan arus listrik yang besar pada Motor DC sedangkan arus

keluaran pada mikrokontroler sangat kecil. *Driver* motor merupakan pilihan alternatif yang harus digunakan untuk mengendalikan Motor DC pada mekanik robot beroda ini. Ada beberapa *driver* motor yang sering digunakan pada aplikasi mekanik, yaitu menggunakan rangkaian H-Bridge transistor, H-Bridge MOSFET, dan IC driver motor. Pada alat yang akan dirancang, *driver* motor yang digunakan adalah *driver* motor dual H-bridge. Berikut ini adalah perhitungan arus ESC yang digunakan untuk menjadi driver motor *brushless*:

$$I = \frac{P}{V}$$

I = Arus Motor

P = Daya pada Motor Brushless

V = Tegangan Baterai

Dik: P = 370 W (daya maximum brushless)

V = 11.1 V (Baterai 3 S)

Dit: I = ...?

$$I = \frac{P}{V}$$

$$I = \frac{370 \text{ W}}{11.1 \text{ V}}$$

$$I = 33.33 \text{ Amper}$$

Maka ESC yang disarankan untuk motor *brushless* ini dengan baterai 3S adalah dengan arus 30 Amper.

i. Power Supply

Power supply pada suatu sistem harus mampu memenuhi kebutuhan tegangan setiap sistem blok. Dalam hal ini, power supply harus mampu memenuhi rangkaian pada modul GPS sebagai Penentu koordinat, rangkaian Arduino Uno sebagai pusat pengendali rangkaian, serta rangkaian driver motor brushless sebagai pengendali dan pemberi suplai bagi motor brushless.

Masing-masing dari sistem blok tersebut memiliki kebutuhan tegangan yang berbeda-beda. Oleh karena itu, power supply harus bisa memberi tegangan yang berbeda-beda pula. Bagian power supply terdiri dari baterai dengan tegangan 11.1 volt DC. Tegangan VDC akan diregulasi terlebih dahulu agar tegangan input sistem blok tidak melebihi batas maksimal.

Catu daya 11.1 volt dirangkai dengan IC 7805 untuk memberi tegangan keluaran 5 volt pada blok sistem. Kemudian dirangkai dengan IC 7809 untuk memberi tegangan keluaran sebesar 9 volt. Keluaran tegangan sebesar 9 volt ini sebagai tegangan masukan untuk Arduino Uno. Sedangkan tegangan supply untuk Driver Motor brushless diparalel langsung dari sumber baterai 11.1 Volt.

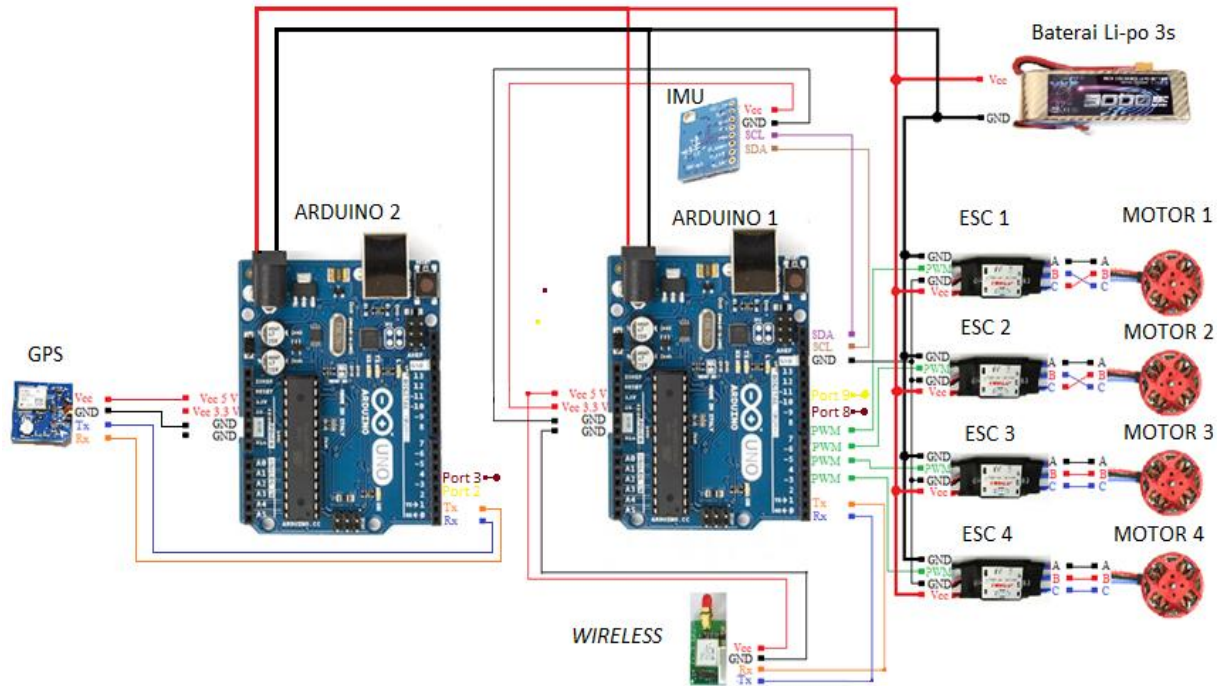


Gambar 3.12 Baterai LI-PO 3000 mAh 25C

Baterai yang digunakan pada sistem ini adalah LI-PO 3 Sel dengan kapasitas 3000 mAh dan *discharge* 25c. Discharge adalah notasi yang menunjukkan seberapa cepat baterai dapat dikosongkan. Untuk menghitung kapasitas baterai yang sebenarnya adalah dengan mengalikan antara kapasitas tertera baterai dengan *discharge*. Maka kapasitas baterai akan menjadi 75000 mAh. Baterai akan habis dalam satu jam jika di beri beban sebesar 75 A.

j. Konfigurasi Pin pada *Quadcopter*

Berikut ini adalah konfigurasi pin pada *quadcopter*, di mana semua modul *hardware* pada perancangan ini dihubungkan sehingga menjadi sebuah system terintegrasi.



Gambar 3.13 Konfigurasi PIN pada *Quadcopter*

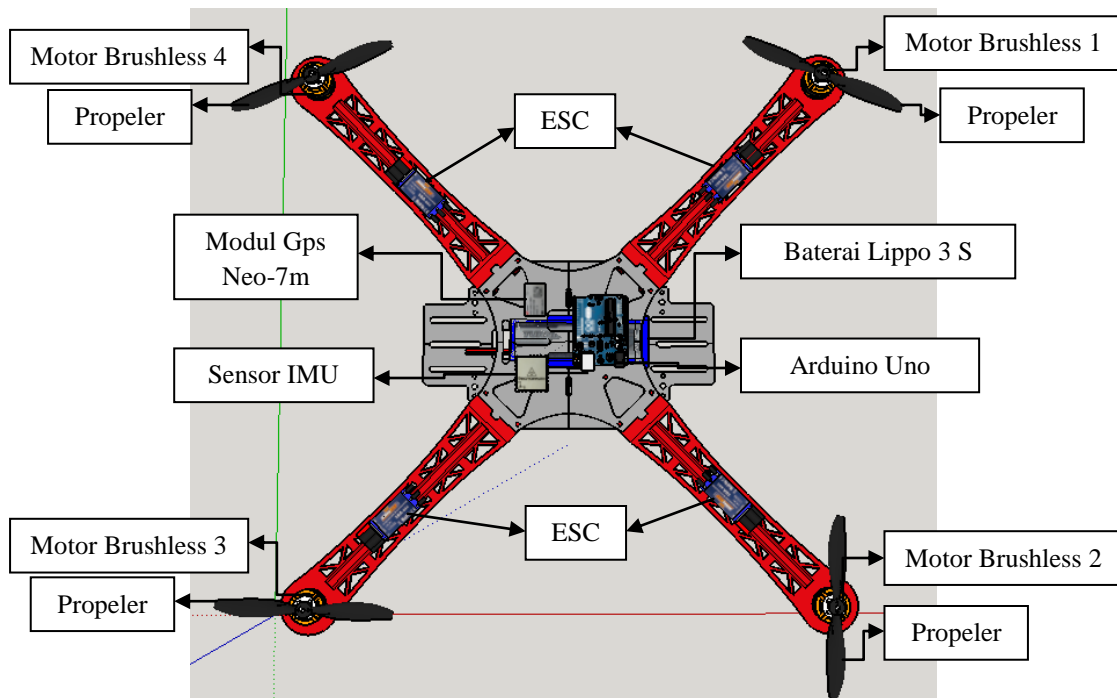
Berdasarkan gambar di atas, dapat dilihat bahwa perancangan quadcopter ini menggunakan dua arduino. Arduino 1 sebagai tempat pengolahan dan pengiriman data motor (utama), sedangkan arduino 2 sebagai penerima data posisi GPS yang akan dikirimkan ke arduino 1. Pada arduino 1, dihubungkan data PWM pada ESC sensor gyro, kompas dan *wireless3DR*. Data PWM pada ESC dihubungkan pada PORT digital yaitu 4,5,6,7 yang masing-masing merupakan motor 1,2,3,dan 4. Sensor gyro dan kompas dihubungkan pada port I2C yaitu SDA dan SCL. Modul *wireless 3DR* dihubungkan pada port UART yaitu Tx dan Rx. Sedangkan GPS, dihubungkan

pada arduino 2 yang merupakan tempat pengiriman data GPS dari arduino 2 ke arduino 1 melalui port UART.

k. PerancanganDimensi *Quadcopter*

Perancangan mekanik dilakukan untuk menentukan ukuran dan bahan yang tepat untuk digunakan. Bahan yang digunakan dalam perancangan mekanik *quadcopter* ini adalah *glass fiber* dengan bobot 405 gr tanpa komponen lainnya.

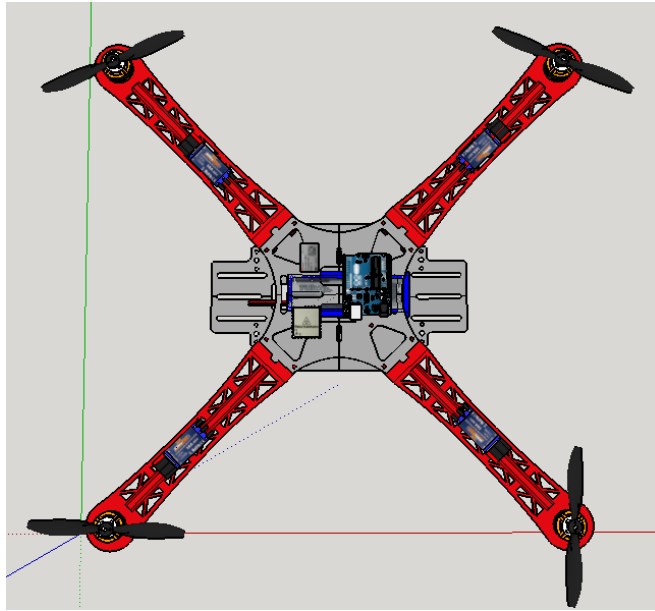
Berikut ini adalah penempatan komponen dan tampak dari *quadcopter*:



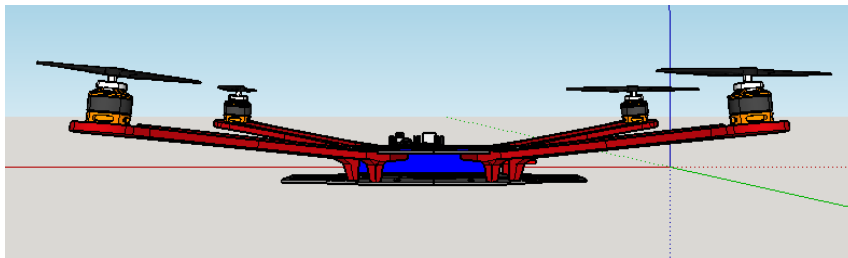
Gambar 3.14 Penempatan Komponen pada *Quadcopter*

Dari gambar diatas, terlihat posisi muka depan *quadcopter* terdapat motor 1 dan motor 4. Sedangkan muka belakang *quadcopter* terdapat motor 2 dan motor 3. Pada masing-masing *arm quadcopter*, diletakan ESC sebagai driver motor, agar tidak jauh posisinya dari motor brushless. Baterai disisipkan diantara frame

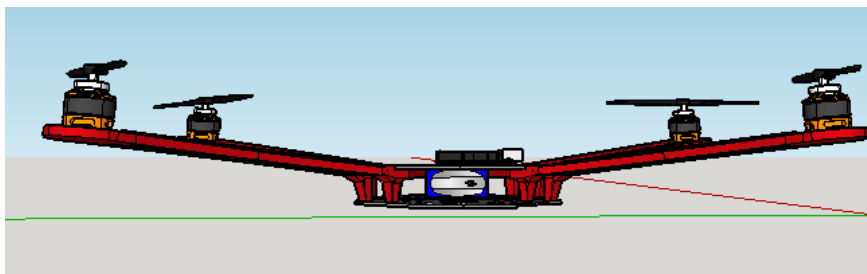
quadcopter. Komponen-komponen hardware lainnya seperti sensor, GPS, *wireless* dan Arduino UNO, diletakan pada posisi tengah atas *framequadcopter*.



Gambar 3.15*Quadcopter* Tampak Atas



Gambar 3.16*Quadcopter* Tampak Samping



Gambar 3.17*Quadcopter* Tampak Samping

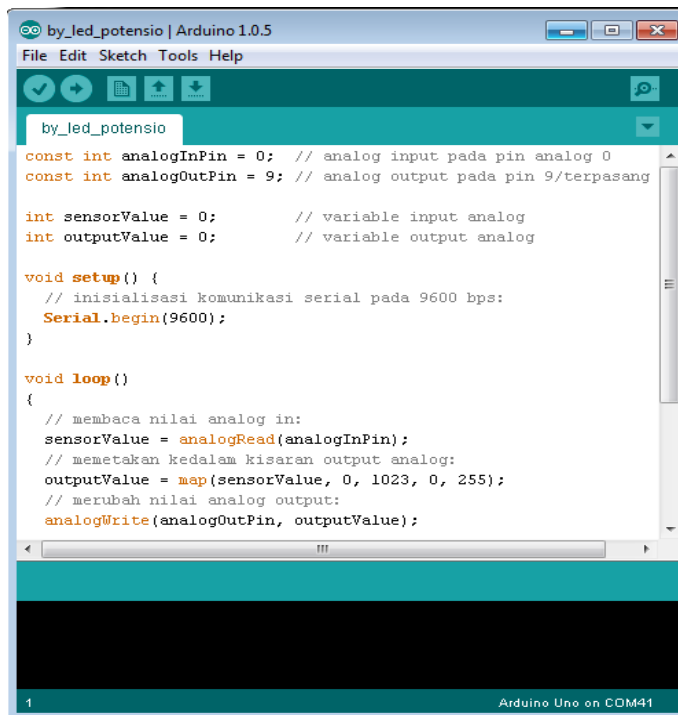
3.3.2. Perancangan Perangkat Lunak (*Software*)

a. Arduino IDE Software

Tugas dari *Arduino Software* adalah menghasilkan sebuah file berformat hex yang akan di-*download* pada papan arduino atau papan sistem mikrokontroler lainnya. Ini mirip dengan Microsoft Visual Studio, *Eclipse* IDE, atau Netbeans. Lebih mirip lagi adalah IDE semacam Code Blocks, CodeLite atau Anjuta yang mempermudah untuk menghasilkan file program. Bedanya kesemua IDE tersebut menghasilkan program dari kode bahasa C (dengan GNU GCC) sedangkan *Arduino Software* (*Arduino IDE*) menghasilkan file hex dari baris kode yang dinamakan sketch.

Pada perancangan alat menggunakan software ini, diatur bagaimana kerja dari program dengan menggunakan instruksi-instruksi yang ada pada mikrokontroler Atmega328 atau *Arduino Uno* sehingga input dan *output*-nya dapat langsung bekerja.

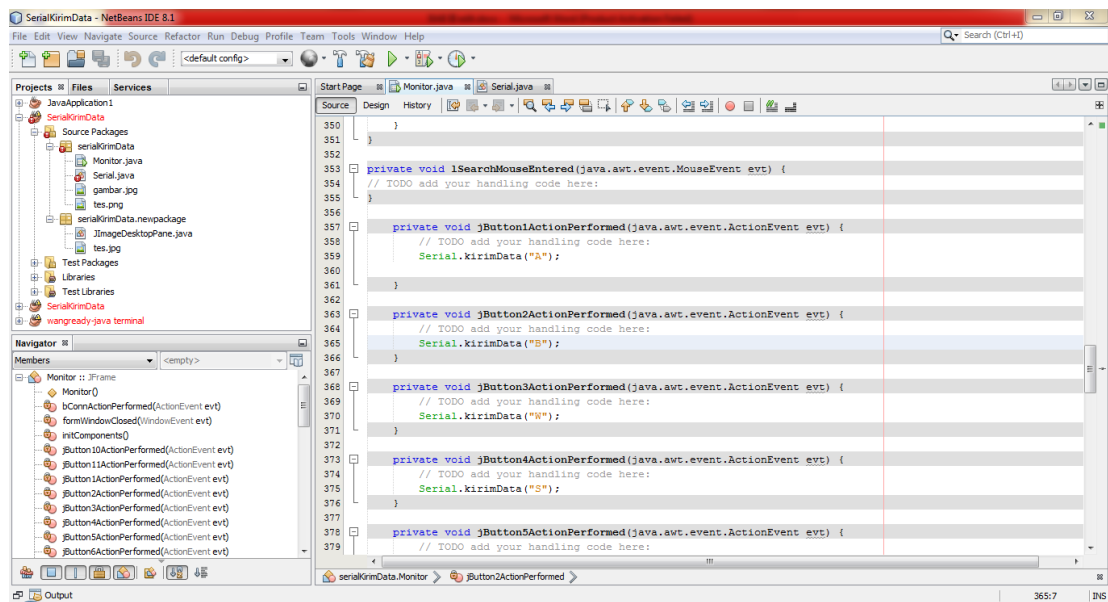
Program dibuat dalam bahasa C menggunakan *software Arduino.cc*, kemudian disimulasikan untuk mendapatkan file hexadesimal dari program yang telah dibuat. Selanjutnya file hexadesimal tersebut di-*download* ke dalam *flash eeprom Arduino UNO*. Berikut adalah tampilan utama *window editor software Arduino.cc*.



Gambar 3.18 Tampilan Aplikasi Arduino.cc

b. NetBeans IDE 8.1

GUI (*Graphical User Interface*) yang digunakan pada system ini adalah NetBeans dengan Java script. Dalam GUI, memuat tombol untuk control dan throttle *quadcopter* yang digunakan untuk mengatur manuver dari *quadcopter* secara manual. Selain itu terdapat tombol navigasi yang berfungsi sebagai perintah untuk navigasi *waypoint*. Berikut ini adalah tampilan gambar dari NetBeans IDE 8.1.

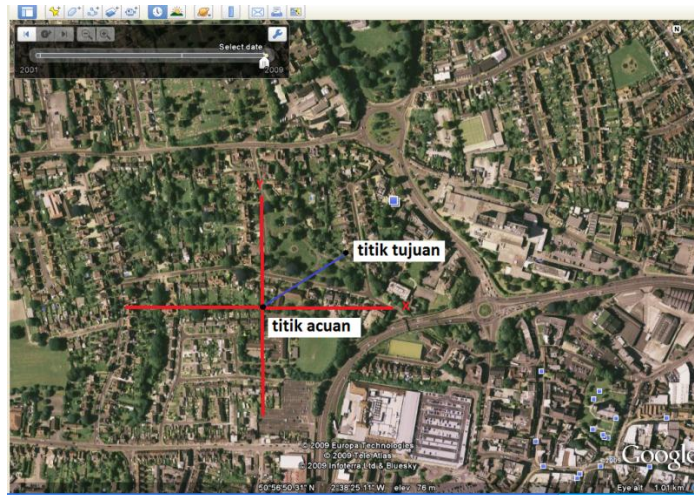


Gambar 3.19 Tampilan Aplikasi NetBeans IDE 8.1

c. Waypoint

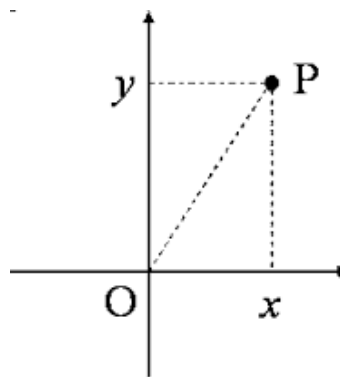
Navigasi *waypoint* adalah kumpulan beberapa titik koordinat yang digunakan untuk bernavigasi dari satu titik ke titik yang lain melalui titik acuan. Titik acuan yang dimaksud di sini adalah titik awal quadcopter untuk bernavigasi, sedangkan titik tujuan adalah titik yang akan dituju oleh quadcopter.

Agar *quadcopter* dapat bernavigasi dari satu titik ke titik yang lain, dilakukan pencarian sudut heading dan bearing. Heading adalah posisi sudut muka quadcopter berdasarkan nilai longitude dan latitude titik acuan terhadap titik tujuan, sedangkan bearing adalah posisi sudut titik tujuan berdasarkan nilai longitude dan latitude titik acuan terhadap titik tujuan. Cara perbandingan antara koordinat (*longitude* dan *latitude*) titik acuan dengan koordinat (*longitude* dan *latitude*) tujuan, salah satunya adalah menggunakan koordinat kartesian.



Gambar 3.20 Garis Lurus Titik Acuan terhadap Titik Tujuan

Gambar diatas adalah contoh gambar garis quadcopter saat bernavigasi dengan membuat koordinat kartisan. Pada koordinat kartesian, diambil titik koordinat awal sebagai koordinat acuan (titik $(0,0)$ pada koordinat) dan mengambil satu titik sebagai koordinat tujuan, dapat dilihat pada gambar koordinat kartesian di bawah ini:

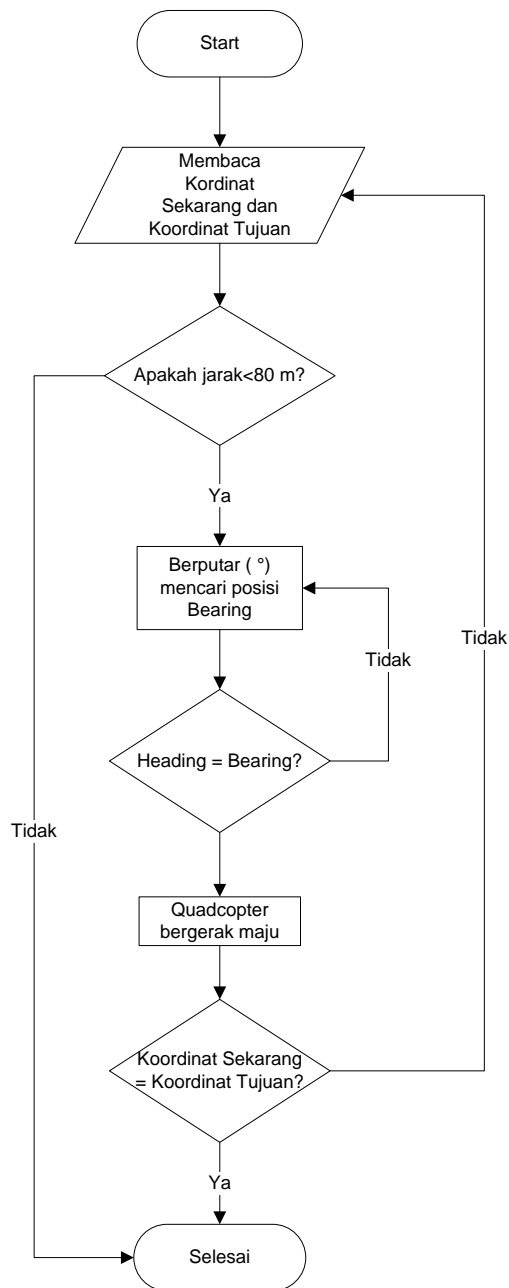


Gambar 3.21 Titik pada Koordinat Kartesian

Quadcopter akan bernavigasi mencari titik tujuan tersebut melalui titik acuan. Dari gambar, terlihat “O” dianggap sebagai titik acuan dan “P” adalah

titik tujuan. Quadcopter akan melakukan pengecekan koordinat untuk membandingkan apakah koordinat sekarang sudah sesuai dengan koordinat tujuan, jika sudah sesuai maka proses selesai.

d. Flowchart Sistem



Gambar 3.22 Diagram Alir Waypoint

Gambar di Atas merupakan gambar flowchart sistem. Dimulai dengan menghidupkan dan menerbangkan quadcopter. Kemudian saat akan bernavigasi *user* menekan tombol “Navigasi” pada GUI. Arduino akan mengambil data koordinat GPS pada posisi quadcopter terbaru dan posisi tujuan. Data koordinat tersebut akan diolah untuk mengetahui jarak, *heading*, dan *bearing*. Setelah terbaca, arduino pada *quadcopter* akan menghitung jarak dari *quadcopter* terhadap titik tujuan. Apabila terbaca lebih dari 80 m, maka proses akan selesai. Apabila terbaca kurang dari 80 m, maka akan menuju proses selanjutnya.

Setelah membaca jarak dan jarak tersebut berada dalam jangkauan, *quadcopter* akan mencari sudut bearing (titik tujuan) berdasarkan sudut headingnya (muka quadcopter). Apabila heading tidak sama dengan bearing, maka *quadcopter* akan berputar untuk menyamakan nilai heading dan bearing. Apabila bearing sama dengan heading, maka quadcopter akan bergerak maju. Apabila posisi koordinat quadcopter terbaru tidak sama dengan tujuan, maka quadcopter akan membaca koordinat *quadcopter* terbaru kembali dan koordinat tujuan. Apabila koordinat tujuan sama dengan koordinat quadcopter terbaru, maka proses selesai.