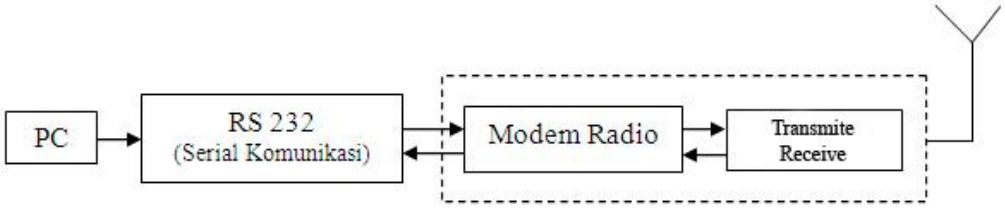
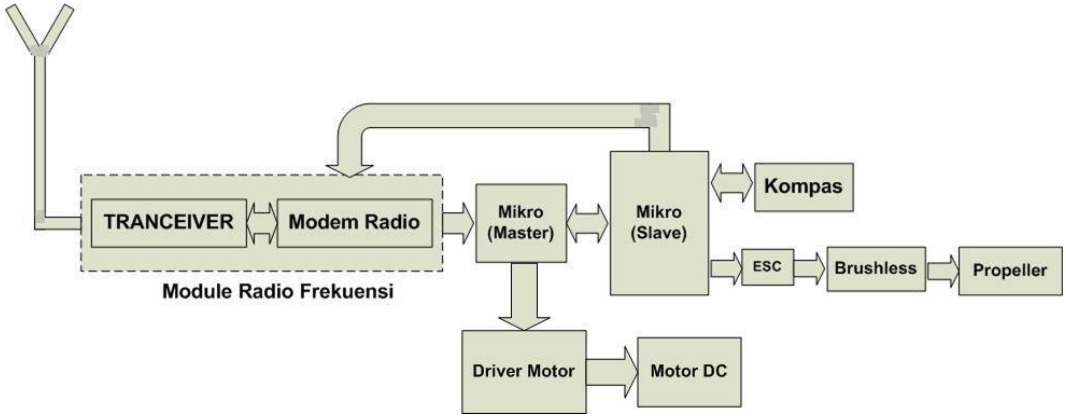
BAB III
PERANCANGAN SISTEM

3.1. Perancangan *Flying Robot* secara umum

Pada bab ini akan dijelaskan mengenai blok sistem secara keseluruhan serta komponen yang membangun perancangan sistem navigasi *flying robot* secara *deadreckoning* menggunakan kompas digital.



Gambar 3.1 Diagram blok sistem pemancar



Gambar 3.2 Diagram blok sistem penerima

Cara kerja dari setiap blok sistem pemancar dan penerima

Blok sistem pengirim/pemancar terdiri dari beberapa bagian yaitu :

1. PC (*Personal Computer*) : sebagai antar muka dari sistem kontrol *flying robot*, dimana PC akan mengirimkan data melalui *transmiter* sebagai perintah untuk *flying robot* bernavigasi. PC juga berfungsi sebagai indikator kondisi atau keadaan kompas sebagai penunjuk arah posisi *flying robot*.

2. RS 232 (Serial Komunikasi): sebagai *buffer*/penyangga atau serial komunikasi, dimana data yang dikirim *transmitter* atau yang diterima oleh *receiver* berbentuk serial, sehingga dibutuhkan *converter* data paralel ke serial. RS232 akan mengkoneksikan antara PC dengan modem pemancar, sehingga data bisa dikirim ke modem penerima dan akan memerintahkan *flying robot* bernavigasi dan juga menerima data dari kompas yang dikirim untuk ditampilkan di PC. Karena data dikomputer diolah secara paralel maka data harus dikonversikan ke bentuk serial untuk bisa digunakan secara *hardware*, hal ini bisa diubah oleh UART (*Universal Asynchronus Receiver Transmitter*).
3. Modem Radio Pemancar (*Transmitter*) : dirancang untuk data sistem transmisi UART dalam jangkauan pendek. Modul dapat langsung berhubungan dengan PC, perangkat RS232, dan lain-lain. Data yang dikirim dari PC akan dikirimkan ke modul penerima pada modem radio, sehingga akan diolah dan akan mengghasilkan *output*, *output* ini berupa gerak *flying robot* secara keseluruhan. Pada bagian ini data yang dibangkitkan dari sistem, sumber tidak ditransmisikan secara langsung dalam bentuk aslinya namun pada sebuah *transmiter* cukup memindahkan informasi dengan menghasilkan sinyal elektromagnetik yang dapat ditransmisikan dengan beberapa sistem transmisi berurutan.

Blok sistem penerima terdiri dari beberapa blok yaitu :

1. Modem Radio Penerima (*Receive*) : digunakan sebagai penerima data dari *transmitter*, kemudian data akan dikirim ke mikrokontroler yang nantinya akan memberikan perintah untuk menggerakkan *flying robot*. Pada bagian ini sinyal dari pengirim diterima dari sistem transmisi dan memindahkan bentuk sinyal elektromagnetik menjadi digital yang dapat ditangkap oeh tujuan, serta mengirim kembali data kompas yang terjadi setelah proses navigasi ke PC.
2. Mikrokontroler : sebagai pengontrol dari sistem. Mikrokontroler terdiri dari dua mikro yaitu mikro *master* dan mikro *slave*. Dimana fungsi masing- masing mikro adalah sebagai berikut :

a. Mikro *Master*
    - Sebagai Pengontrol dari seluruh sistem
    - Sebagai pemberi perintah kepada mikro *slave* untuk menggerakan *brushless* cepat atau lambat.
    - Sebagai pengontrol pergerakan motor *servo*
    - Sebagai *receiver* (penerima) data dari PC melalui radio frekuensi.

b. Mikro *Slave*
    - Sebagai pengolah data kompas
    - Sebagai pengerak motor *brushless* yang dikendalikan oleh ESC.
    - Sebagai *input* dan *output* dari sistem komunikasi antar mikro dan sebagai transmiter data kompas ke .

3. *Electronic Speed Controller* (ESC) - *Brushless* : digunakan sebagai pengatur putaran pada motor *brushless*, ESC akan mendapatkan data dari mikrokontroler yang kemudian akan memberikan sinyal pulsa pada brushless dan menggerakkan *propeller* sesuai data inputan dari mikro.
4. *Propeller* : digunakan pada *flying robot* sebagai alat gerak, biasanya dipasang di atas *flying robot* dan di ekornya. *Propeller* ini akan menghasilkan angin dari putarannya, sehingga membantu motor untuk bisa terbang. Pada blok diagram *propeller* akan digerakkan oleh ESC,Motor Brushless yang sudah di kontrol oleh mikrokontroler, kemudian *propeller* akan menghasilkan putaran yang akan menggerakkan *flying robot* ke arah sesuai yang kita tentukan, setelah *flying robot* bergerak maka kompas akan mengecek sudut yang dihasilkan oleh gerak *flying robot*.
5. Kompas Digital : berfungsi sebagai penunjuk arah dari gerak *flying robot*. Kompas digital ini dipasang pada *flying robot*. Pada saat *flying robot* bergerak ke suatu arah yang kita tentukan maka kompas digital ini akan membaca nilai sudut tertentu sesuai dengan arah yang ditunjuk oleh *flying robot*. Selain *output* yang dihasilkan kompas juga akan di set terlebih dulu, yang nantinya nilai *set point* ini akan dibandingkan dengan nilai *output* yang

dihasilkan oleh kompas. Sudut yang digunakan sesuai dengan arah mata angin. Tapi hanya akan digunakan beberapa sampel.

6. Motor driver : berfungsi sebagai penggerak motor dc dan mengatur kecepatannya.
7. Motor DC : sebagai actuator.

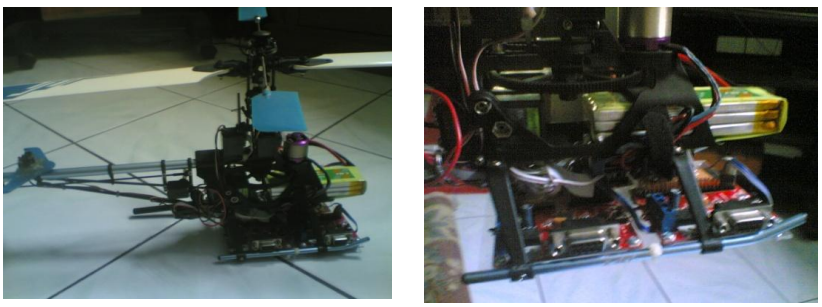
3.2. Perancangan Perangkat Keras (*Hardware*)

3.2.1. Perancangan Mekanik

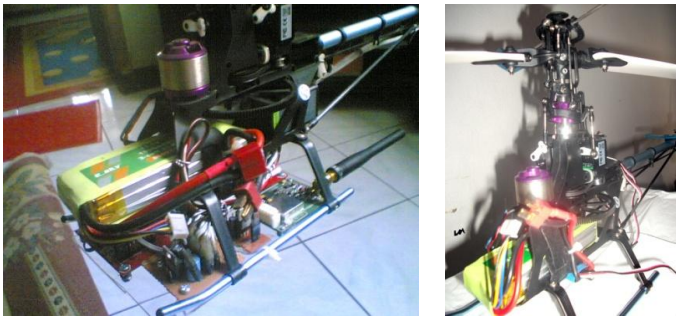
Sebuah sistem *flying robot* sebenarnya tertumpu pada sistem keseimbangannya, ilmu yang memepelajarinya adalah *aeromodeling*, perancangan *frame* bentuk *flying robot* mengadopsi dari *frame* helikopter mainan yang biasa dijual bebas dipasaran, karena selain telah teruji keseimbangan dalam bernavigasi, juga telah adanya sistem *propeler* yang lebih *sinkron* untuk bentuk *frame* keseluruhan, dimana dimensi/ukuran dari *frame flying robot* itu sendiri adalah:

1. Panjang = 550 mm
2. Lebar = 120 mm
3. Tinggi = 200 mm
4. Berat = 390 g
5. Bentuk 3 dimensi (3D)

maka dari itu perancangan mekanik secara keseluruhan dapat dilihat seperti gambar dibawah ini.

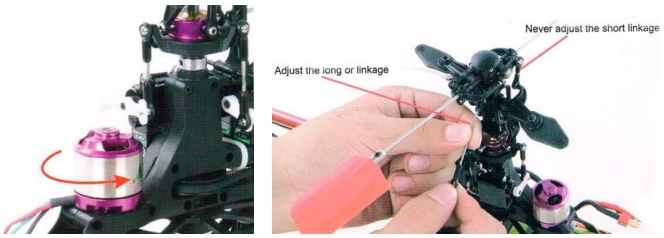


Gambar 3.3 *Frame flying robot* tampak samping



Gambar 3.4 *Frame flying robot* tampak samping depan

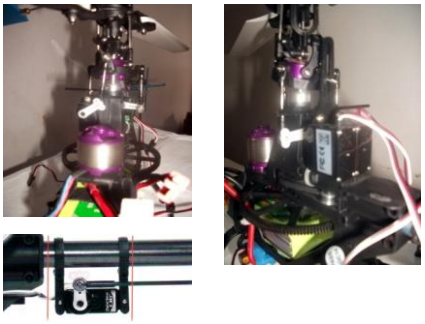
Pada dasarnya *propeler* akan bergerak secara simultan dan tidak berubah-ubah, yang menyebabkan *flying robot* dapat bernavigasi maju, mundur ke kiri dan ke kanan adalah terdapat pada *balancer propeler* yang digerakan oleh *servo* motor, sehingga akan mengubah posisi / bentuk kemiringan *propeler* dan menyebabkan perubahan tekanan udara, sehingga menyebabkan terjadi gerakan dorong atau tekan sesuai arah tekanan udara, dengan demikian terciptalah navigasi *flying robot* yang diinginkan.



Gambar 3.5 posisi motor brushless dan *balancer propeller*

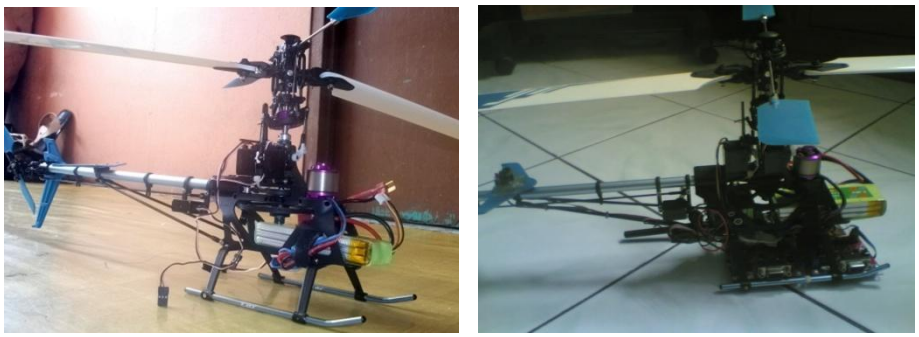
Sedangkan untuk navigasi *flying robot* ke samping (kanan-kiri), tidak hanya pergerakan dari *propeler* utama saja, karena *propeler* utama hanya menghasilkan tekanan udara dari arah atas dan bawah saja, sehingga lebih cenderung untuk navigasi maju, mundur, naik dan turun, serta membantu membentuk sudut *elevasi* sebagai acuan gerak pada saat benavigasi ke arah samping, jadi untuk navigasi ke arah samping bukan hanya satu *propeler* saja yang bekerja tetapi didukung oleh *propeler* kecil dibelakang, sehingga navigasi *flying robot* untuk gerak ke kiri dan ke kanan menggunakan *propeler* belakang.

Propeler kecil dibelakang ini juga adalah sebagai penjaga keseimbangan pada saat posisi diam pada porosnya dan juga pada saat navigasi maju, mundur, naik dan turun.



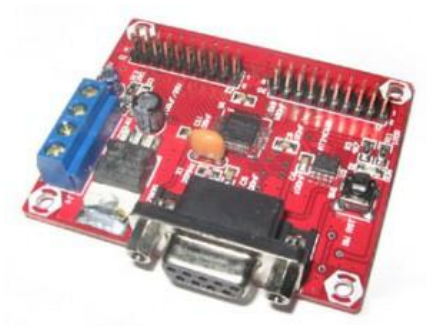
Gambar 3.6 posisi *servo* motor penggerak tuas pengatur *propeller*

Dengan adanya kebutuhan sebuah bentuk *frame flying robot*, maka digunakanlah *frame* helikopter mini *e-sky* sebagai bahan perancangan.



Gambar 3.7 Bentuk Utuh *Flying Robot*

3.2.2. Mikrokontroler *Basic Stamp* (BS2P40)



Gambar 3.8 mikrokontroller BS2P40

Mikrokontroler yang digunakan adalah mikrokontroler jenis *basic stamp* BS2P40 yang dilengkapi 40 *port* yang bisa digunakan sebagi *input* ataupun *output*. Pemilihan mikrokontroler jenis ini didasari karena mudah dalam penggunaannya, serta *compatible*-nya mikrokontroler ini dengan modul alat yang menjadi pendukung dalam perancangan *flying robot* ini, diantaraya adalah pada modul kompas digital yang memang satu produk dan juga *compatible*, serta untuk aplikasi yang menggunakan sistem pulsa yaitu *servo* motor dan brushless, mikrokontroller ini dirasa cukup baik untuk pengontrolannya. Mikrokontroler yang digunakan yaitu sebanyak dua buah mikro, mikro master dan mikro slave. Berikut konfigurasi portnya yang dipakai.

Tabel 3.1 Konfigurasi *port* mikro *master* yang digunakan.

Nama Pin	**No. Pin**	**Fungsi**	**Keterangan**
MAIN I/O	0	Radio *In*	Sebagai penerima data dari PC ke Mikro.
MAIN I/O	1	Radio *Out*	Sebagai Pengirim data dari mikro ke ke PC.
MAIN I/O	2	*Servo Belakang*	Penggerak posisi arah Baling Belakang
MAIN I/O	3	Komunikasi*on*	Komunikasi antar mikro untuk perintah aktivasi *brushless*
MAIN I/O	4	*Servo* Kanan	Penggerak posisi baling utama
MAIN I/O	5	Komunikasi*up*	Komunikasi antar mikro untuk pergerakan *brushless* semakin cepat.
MAIN I/O	6	*Servo* kiri	Penggerak posisi baling utama
MAIN I/O	7	Ngobrol	Komunikasi data antar mikro untuk pengiriman data kompas.
MAIN I/O	8	*Servo* depan	Penggerak posisi *balancer* baling utama.
MAIN I/O	11	Aen	*Enabel* motor a

MAIN I/O	13	Aplus	Polaritas positif untuk motor a
MAIN I/O	15	Amin	Polaritas negatif untuk motor a
AUX I/O	0 – 15	-	-

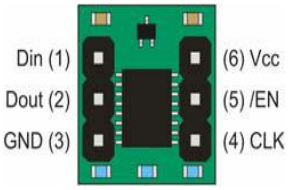
Tabel 3.2 Konfigurasi *port* mikro *slave* yang digunakan.

Nama Pin	No. Pin	Fungsi	Keterangan
MAIN I/O	0	*Brushless*	Penggerak motor *Brushless*
MAIN I/O	1	Radio *Out*	Pengirim Data Kompas ke PC
MAIN I/O	2	*Din/Dout*	Kirim dan Terima data dari kompas.
MAIN I/O	3	Komunikasi*on*	Komunikasi untuk aktivasi *brushless*
MAIN I/O	4	*Clk*	*Clock* untuk kompas
MAIN I/O	5	Komunikasi*up*	Komunikasi untuk percepatan kecepatan motor *brushless*
MAIN I/O	6	*En*	*Enable* untuk kompas
MAIN I/O	7	Ngobrol	Komunikasi dua mikro untuk baca data kompas.
AUX I/O	0 – 15	-	-

3.2.3. Sensor Kompas

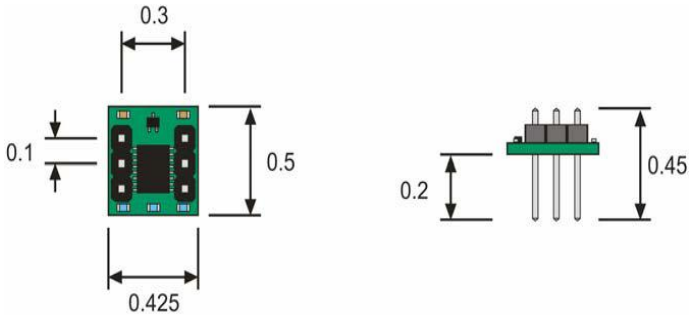
Sensor Kompas yang digunakan adalah sensor kompas digital HM55B. Alasan pemilihan sensor ini karena ukuran kecil dan relatif ringan sehinga kemungkinan tidak akan mengganggu keseimbangan *flying robot* karena beban yang berat, selain itu *output* yang dihasilkan tergolong baik dan akurat.

Gambar bentuk dari sensor kompas HM55B.



Gambar 3.9 Kompas digital HM55

Dimensi dari Kompas Digital HM55B.



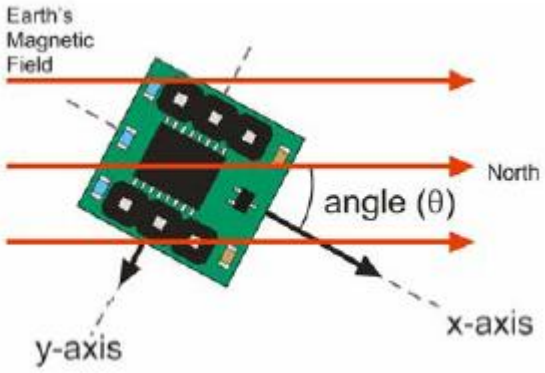
Gambar 3.10 Dimensi Kompas Digital

Jumlah sensor kompas digital yang digunakan yaitu satu buah dengan konfigurasi pin sebagai berikut:

Tabel 3.3 Konfigurasi pin pada kompas digital HM55B.

Nama PIN	Keterangan
Din	Serial data *input*
Dout	Serial data *output*
GND	*Ground* 0 *Volt*
CLK	*Clock input sinkron*
EN	*Enabel* (aktif *low*)
VCC	*Power input* 5 *Volt*

Berikut Gambar skema rangkaian sensor kompas digital HM55B.



Gambar 3.11 Skema sensor kompas tipe HM55B

Cara kerja sensor kompas digital HM55B :

Kompas digital ini cukup *supply* tegangan sebesar 5 Vdc dengan konsumsi arus 15mA. Kompas digital dibagi menjadi dua sumbu yang memiliki medan magnet sejajar X-axis (medan magnet) x Cos θ, Y-axis (medan magnet) x Sin θ. Rumus untuk mencari sudut yaitu menggunakan:

$$\text{angle } \theta = \arctan(-y/x)$$

Pada HM55B, lebar pulsa positif merepresentasikan sudut arah. Lebar pulsa bervariasi antara 1mS (00) sampai 36.99mS (359.90). Dengan kata lain lebar pulsa berubah sebesar 100uS setiap derajatnya. Sinyal akan *low* selama 65mS diantara pulsa, sehingga total periodanya adalah 65mS + lebar pulsa positif (antara 66mS sampai 102mS). Pulsa tersebut dihasilkan oleh *timer* 16 bit di dalam prosesornya, yang memberikan resolusi 1uS. Kompas bekerja selain mengecek sudut yang dituju juga membandingkan dengan sudut yang telah di set sebelumnya pada program.

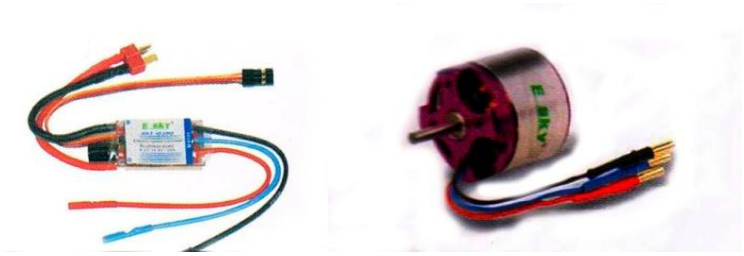
3.2.4. *Electronik Speed Controller* (ESC) - Motor Brushless

Elektronik Speed Controller (ESC) adalah sirkuit elektronik untuk mengatur kecepatan motor listrik secara bervariasi, mengatur putaran arah dan juga sistem pengereman secara dinamis dan biasanya digunakan pada sistem yang menggunakan radio kontrol.

Sebuah ESC sendiri biasa dihubungkan ke saluran kontrol penerima *throttle* atau dimasukkan ke penerima itu sendiri. Tergantung dari tipe yang digunakan, sebuah ESC memberikan informasi kontrol bukan gerak mekanis seperti yang pada *servo*, yaitu dengan cara yang bervariasi, mulai dari pelan hingga cepat tergantung tingkat *switching* dari efek medan transistor, atau FET. Perpindahan cepat transistor inilah yang menyebabkan motor berputar cepat atau lambat, hal ini juga memungkinkan variasi putaran yang lebih halus dan lebih tepat, serta kecepatan motor dengan cara yang jauh lebih efisien dibandingkan dengan tipe mekanik dengan *koil resistif*. ESCs Paling modern menggabungkan rangkaian

baterai *eliminator* (atau BEC) untuk mengatur tegangan penerima. BECs biasanya berbentuk linier atau berupa regulator tegangan.

DC ESCs dalam arti luas adalah pengendali PWM untuk motor listrik. ESC umumnya menerima 50 Hz frekuensi PWM *servo*, sinyal input yang bervariasi dari lebar pulsa 1-2 ms. Ketika diberikan dengan lebar pulsa 1 ms pada 50 Hz, ESC merespon dengan mematikan motor DC pada *output*-nya. Jika dibeikan 1.5 ms lebar pulsa, maka akan menggerakan motor 50% atau setengah dari kecepatan, dan jika diberi lebar pulsa 2ms maka kecepatan motor *full* (terus-menerus), ESC ini berfungsi sebagai pen-*drive* motor sehingga motor brushless dapat bergerak dan menggerakan *propeler*. ESC Brushless yang digunakan adalah 25A brushless *speed controlled* sedang motor brushless yang digunakan adalah motor brushless 400.



Gambar 3.12 ESC dan motor *Brushless*

3.2.5. Radio Frekuensi

Media komunikasi yang digunakan yaitu melalui frekuensi radio (RF). Untuk melakukan komunikasi melalui frekuensi radio menggunakan modem huawei YS-1020K.



Gambar 3.13 Modul Radio YS-1020K

YS-1020K merupakan modul komunikasi yang sangat aman dan mudah digunakan, radio ini mempunyai 8 kanal dengan frekuensi yang berbeda-beda, frekuensi yang digunakan pada sistem ini adalah 433 Mhz pada kanal 6. Jarak

jangkauan komunikasi dari radio ini maksimal sekitar 700 meter pada *baudrate* 9600 bps.

Tabel 3.4 Susunan pin radio YS-1020K yang digunakan.

PIN	NAMA PIN	FUNGSI	KETERANGAN
1	GND	*Ground*	*Negative* tegangan
2	VCC	Tegangan *input*	*Positive* tegangan
3	RXD/TTL	*Input* serial data	Baca data dari mikro
4	TXD/TTL	*Output* serial data	Kirim data ke mikro
5	DGND	Digital *Ground*ing	-
6	A(TXD)	A of RS-485 or TXD of RS-232	-
7	B(RXD)	B of RS-485 or RXD of RS-232	-
8	*SLEEP*	*Sleep control* (*input*)	-
9	*TEST*	*EX-Factory testing*	-

Metode transmisi yang digunakan yaitu metode *Half Duplex* dimana :

a. Sinyal ditransmisikan ke dua arah secara bergantian

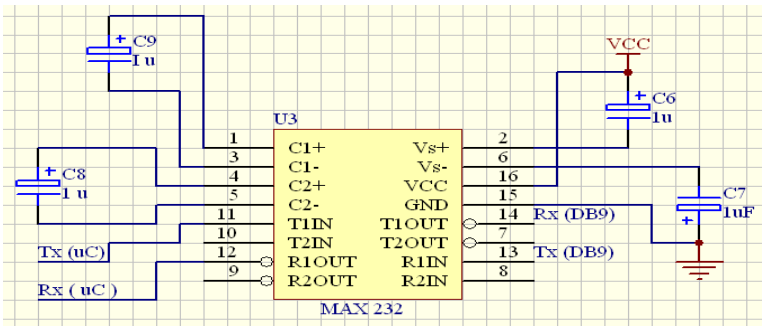
b. Kedua stasiun dapat melakukan transmisi tetapi hanya sekali dalam suatu waktu.

c. Terdapat "*turn around time*" (waktu untuk mengubah arah)



3.2.6. Modul Komunikasi Serial

Konektor pada komputer (DB-9) akan mengeluarkan data dalam level tegangan RS-232. Supaya data yang dikirim dapat diterima oleh PC, maka dibutuhkan suatu penghubung untuk menyamakan masing-masing level tegangan. Rangkaian tersebut dirancang dengan sebuah IC MAX 232. Kegunaan IC MAX232 adalah sebagai *driver*, IC MAX 232 ini berfungsi merubah/mengkonversi tegangan atau kondisi logika TTL dari *hardware* agar sesuai dengan tegangan pada komputer . IC yang dipakai pada sistem ini memiliki 16 pin. IC ini memerlukan komponen tambahan berupa kapasitor agar dapat dihubungkan dengan *port* serial PC dan terminal TTL. Kapasitor pada rangkaian IC MAX 232 berfungsi sebagai *charge* untuk menyuplay muatan ke bagian pengubah tegangan. Sistem ini dioperasikan dengan catu daya 5 *volt*.



Gambar 3.14 Skema Rangkaian IC Max232

3.2.7. *Propeller*

Baling-baling untuk menjalankan kapal atau pesawat terbang. Baling-baling ini memindahkan tenaga dengan mengkonversi gerakan rotasi menjadi daya dorong untuk menggerakkan pesawat terbang, untuk melalui suatu massa udara, dengan memutar dua atau lebih baling-baling yang sama dari sebuah poros utama. Baling-baling dari sebuah propeler berperan sebagai sayap berputar, dan memproduksi gaya yang mengaplikasikan prinsip *bernoulli* dan hukum gerak *newton*, menghasilkan sebuah perbedaan tekanan antara permukaan depan dan belakang baling-baling tersebut. Sehingga dengan adanya gaya tekan atau dorong yang dihasilkan akan dapat mengangkat *flying robot*.

Adapun *propeler* yang digunakan adalah *propeler tail boom esky*, dengan panjang 250 mm dan lebar 30 mm.



Gambar 3.15 *Propeler*

3.2.8. Sumber Tegangan

Sumber tegangan yang dipakai yaitu :

1. Lithium polymer

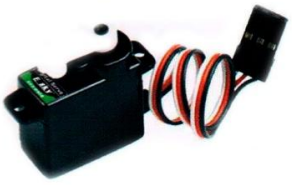
Baterai yang digunakan adalah *Li-Poly Battery* 1500 mAh, 11.1 V *discharge* sebagai sumber tegangan untuk motor brushless dan mampu bertahan selama 6-8 menit, serta *Li-Poly Battery* 1550 mAh, 11.1V sebagai sumber tegangan untuk *servo*, mikro, dan kompas.



Gambar 3.16 baterai Lithium polymer

3.2.9. Motor *servo*

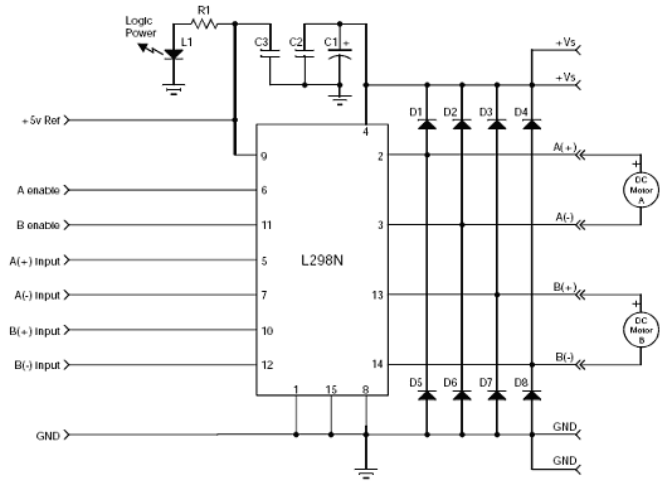
Motor *servo* yang digunakan adalah *e-sky* 7.5g digital *servo* sebanyak 4 buah, dimana fungsi dari motor *servo* itu sendiri adalah sebagai penarik atau pendorong tuas tempat dudukan *propeler* guna membuat perubahan bentuk kemiringan pada *propeler* yang akan menghasilkan gaya tekan dari udara sesuai yang diharapkan, dengan adanya *servo* ini maka tercipta navigasi *flying robot* maju, mundur dan gerak kesamping.



Gambar 3.17 motor *servo e-sky*

3.2.10. IC Motor Driver L298

L298N diproduksi *STMicroelectronic* merupakan komponen *Dual Full-bridge driver*. Komponen ini bisa men-*drive* dua buah motor DC sampai tegangan 46 Vdc dan arus sebesar 2A untuk tiap kanal. Satu motor dapat dikontrol dengan 3 pin seperti A*enable*, A+ dan A-. Pin *output* yang menuju motor DC dihubungkan dengan dioda 1N4002 yang dihubungkan ke sumber Vcc dan *ground* motor DC. Pemilihan dioda ini berdasarkan kemampuannya untuk menangani arus tinggi yang mampu mengurangi *spike* yang terjadi akibat perubahan arah gerak robot.



Gambar 3.1 Rangkaian motor driver L298N

Tabel 3.5 Tabel kebenaran untuk satu motor

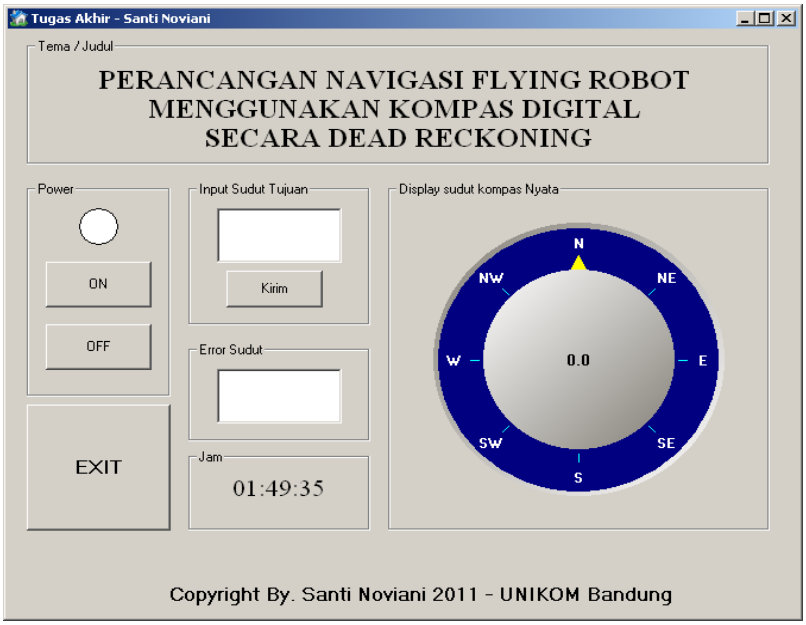
Aen	A(+)	A(-)	Status Motor
L	X	X	Motor *off*
H	L	L	Stop (rem)
H	H	L	Berputar *clockwise* (cepat)
H	L	H	Berputar *counter clockwise* (cepat)
H	H	H	Stop (rem)
P	H	L	Berputar *clockwise* (cepat)
P	L	H	Berputar *counter clockwise* (cepat)

H = *High* ; L = *Low* ; P = *Pulsa* ; X = *Don't care*

3.3. Perancangan Perangkat Lunak (*Software*)

3.3.1. *Software* pengontrol navigasi *flying robot.*

Dalam pengontrolannya, *flying robot* akan dikontrol melalui *software* yang dibuat pada *software visual basic*, dimana ilustrasi tampilan pengontrolan dapat dilihat seperti gambar di bawah ini.



Gambar 3.19 *Software* pengontrol *flying robot*

Berikut keterangan dari gambar tampilan interface *flying robot* pada *software* visual basic.

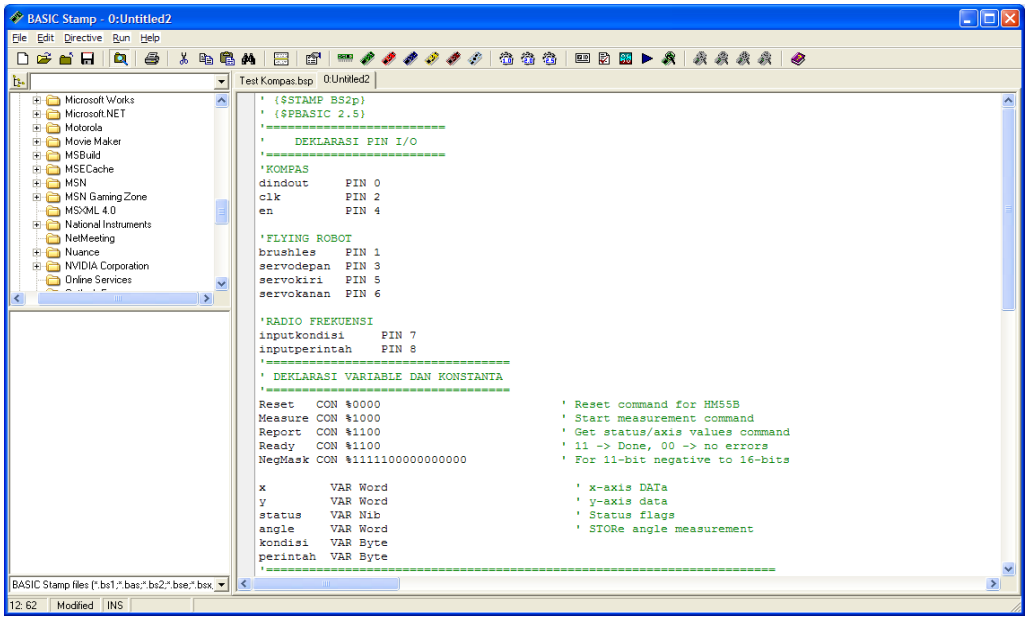
Tabel 3.6 Keterangan instruksi pada tampilan *interface flying robot*

No	Nama instruksi	Keterangan
1	*ON*	Instruksi untuk menggerakkan *flying robot.* Dilakukan dengan penekanan tombol 'q' pada *keyboard* PC.
2	*OFF*	Instruksi untuk mematikan *flying robot.* Dilakukan dengan penekanan tombol 'w' pada *keyboard* PC.
3	*EXIT*	Instruksi untuk keluar dari proses.
4	Sudut tujuan	Meng*input*kan data sudut.
5	*Error* kompas	Tampilan data *error* kompas.
6	*Send* data	Proses pengiriman *input* sudut.

7	*Time*	Tampilan jam.
8	*Display* Kompas	Tampilan sudut yang dibaca kompas termasuk sudut terakhir.

Tombol *ON-OFF* adalah perintah kepada *flying robot* untuk menghidupkan dan mematikan kondisi motor atau *flying robot*, dimana setiap penekan tombol akan memberikan perintah kepada *flying robot* bergerak selama kurang lebih 1 detik, sedang panel sudut tujuan yaitu tampilan untuk meng*input*kan sudut yang ingin dicapai dengan cara memberikan besaran sudut yang akan atau harus ditempuh oleh *flying robot*. *Range* sudut yang digunakan dari 0-359. Setelah sudut di*input*kan maka robot akan bergerak ke arah sudut yang di*input*kan dan kondisi *display* akan membaca sudut terakhir, jika melebihi sudut yang di*input*kan atau sudut tujuan maka *error* akan ditampilkan pada tampilan *error* kompas.

3.3.2. *Software* gerak navigasi *flying robot*.

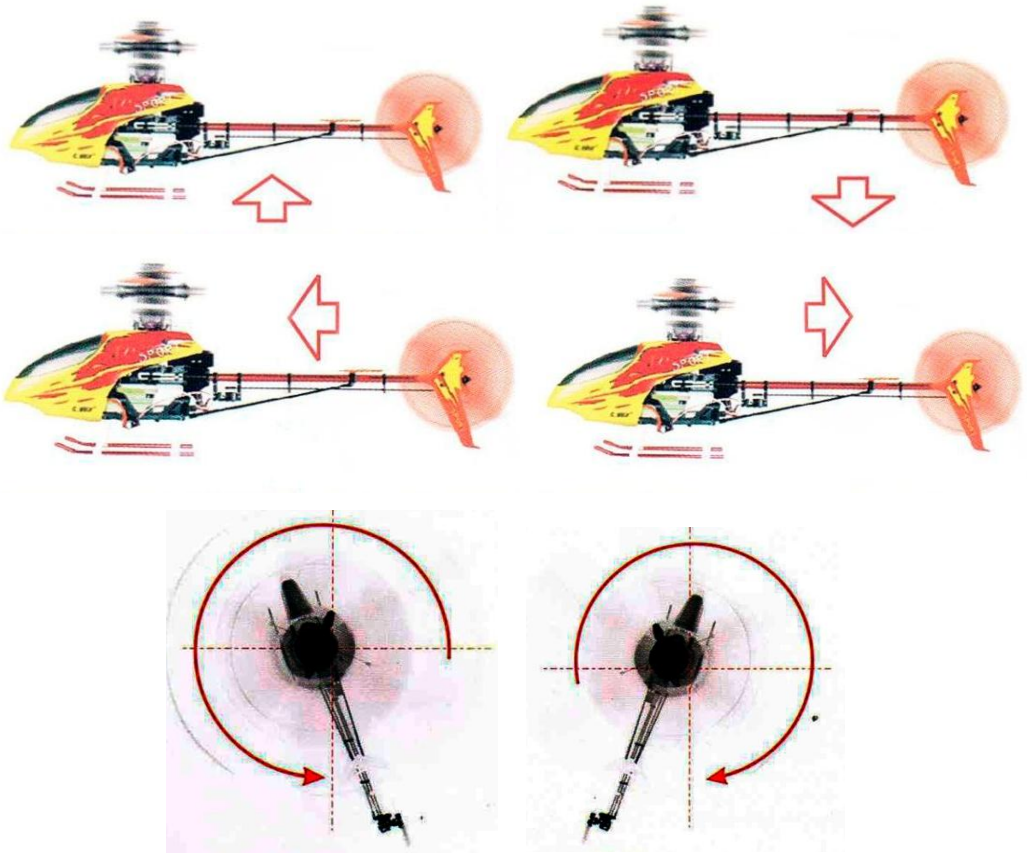


Gambar 3.20 *Software* program gerak *flying robot*

3.3.3. Algoritma dasar

Perancangan algoritma dasar sangat penting untuk sistem kendali pada robot, karena algoritma yang akan menjadi panduan dalam mengembangkan

algoritma kendali keseluruhan dari navigasi yang telah ditentukan. Berikut adalah bebrapa gerak navigasi *flying robot*.

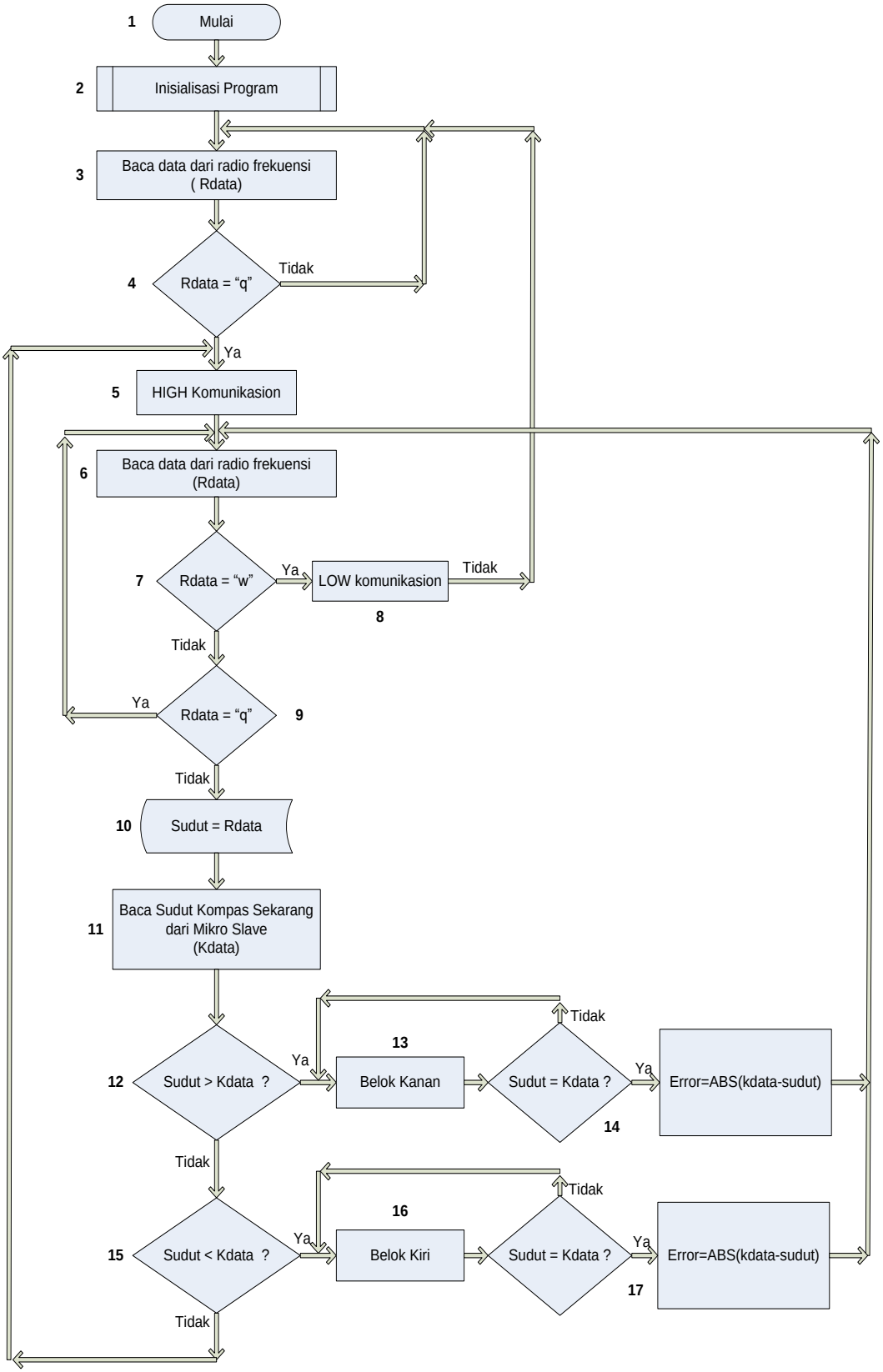


Gambar 3.21 Ilustrasi navigasi *flying robot*

Ketika pertama kali *flying robot ON*, robot akan menunggu data dari komputer, ketika di komputer di *ON*-kan maka *flying robot* hidup dimana *propeler* berputar, kemudian menunggu data untuk dapat terbang (*up*), dan menunggu data *input* sudut, setelah data sudut diinputkan maka robot akan bergerak kearah sudut yang di*input*kan.

3.3.4. Algoritma Seluruh Kendali Robot

Pada bagian ini adalah algoritma navigasi robot dengan tugas bergerak sesuai *input*an dari PC oleh *user*.

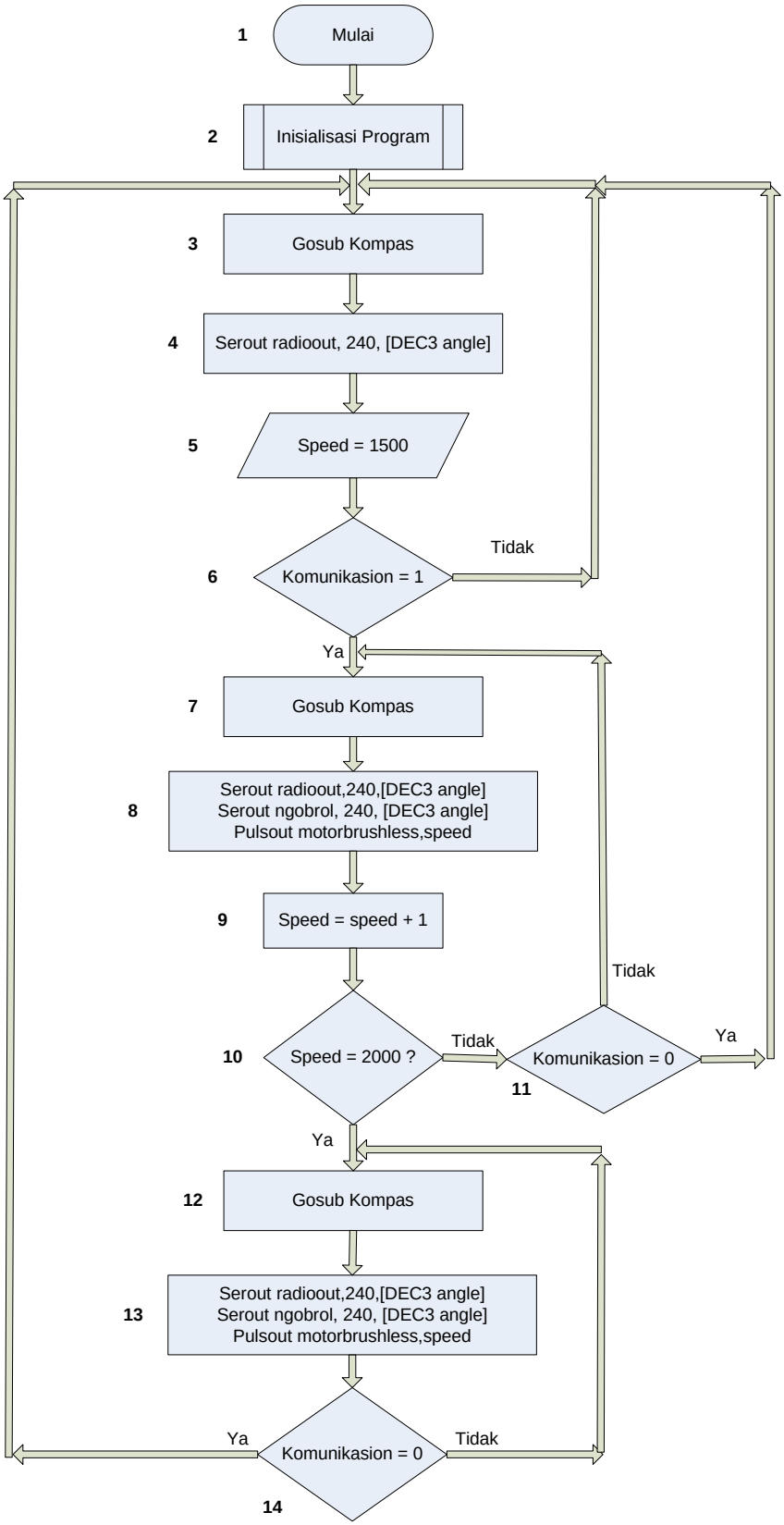


Gambar 3.22 *Flowchart* program utama pada Mikro *Master*

Berikut ini adalah penjelasan *flowchart* program utama pada mikro *master* :

Tabel 3.7 Penjelasan *flowchart* program utama pada mikro *master*

No	Keterangan
1	Memulai awal program.
2	Pendeklarasian program. Pengaturan nama prosesor,nama *kompiler*, konfigurasi pin-pin dan deklarasi variabel.
3	Pembacaan data yang dikirim oleh radio frekuensi dengan variable rdata.
4	Pengecekan tombol *ON*, karakter yang dikirim yaitu “q”
5	Perintah kepada mikro *slave* untuk menjalankan *brushless*
6	Pembacaan data dari radio frekuensi untuk kondisi *OFF*/ *input* data *set point*.
7	Pengecekan tombol *OFF*, karakter yang dikirim yaitu “w”.
8	Mematikan *Brushless*.
9	Pengecekan tombol *ON*, karakter yang dikirim yaitu “q”.
10	Data dengan variable rdata disimpan di variable sudut.
11	Pembacaan sudut akhir dari mikro *slave*.
12	Pengecekan data kompas jika sudut > kdata
13	Instruksi untuk belok kearah kanan sesuai input sudut
14	Pengecekan Sudut yang diinputkan = sudut akhir yang didapat
15	Jika tidak looping hingga tercapai, jika Ya hitung error dan kembali baca data input Radio Frekuensi.
16	Pengecekan data kompas jika sudut < kdata
17	Instruksi untuk belok kearah kiri sesuai input sudut
18	Pengecekan Sudut yang diinputkan = sudut akhir yang didapat
19	Jika Tidak, looping hingga tercapai, jika Ya hitung error dan kembali baca data input Radio Frekuensi.



Gambar 3.23 *Flowchart* program utama pada Mikro *Slave*

Berikut ini adalah penjelasan *flowchart* program utama pada mikro *slave* :

Tabel 3.8 Penjelasan *flowchart* program utama pada mikro *slave*

No	Keterangan
1	Memulai awal program.
2	Pendeklarasian program. Pengaturan nama prosesor,nama *kompiler*, konfigurasi pin-pin dan deklarasi variabel.
3	Pemanggilan program kompas.
4	Kirim data kompas
5	Pemberian pulsa untuk kecepatan motor *brushless*
6	Pengecekan komunikasi, komunikasi jalan.
7	Pemanggilan program kompas.
8	Pengiriman data kompas, data pulsa untuk menggerakan brushless.
9	Proses pengaturan kecepatan motor brushless.
10	Pengecekan pulsa bhrushless dengan kondisi kecepatan brushless sebesar 2000
11	Jika pulsa brushless terpenuhi maka komunikasi jalan
12	Pemanggilan program kompas
13	Pengiriman data kompas, data pulsa untuk menggerakan brushless.
14	Komunikasi low