BAB III

PERANCANGAN SISTEM

Pada bab ini membahas tentang perancangan sistem yang menacu kepada bentuk dan sistem kerja dari robot yang telah dirancang, dan integrasi dari semua komponen yang ada pada robot.

Sistem perancangan akan dibagi menjadi tiga bagian yaitu:

1. Perancangan mekanik (bentuk dan implementasi robot)
2. Perancangan perangkat keras (*hadware*)
3. Perancangan sofware

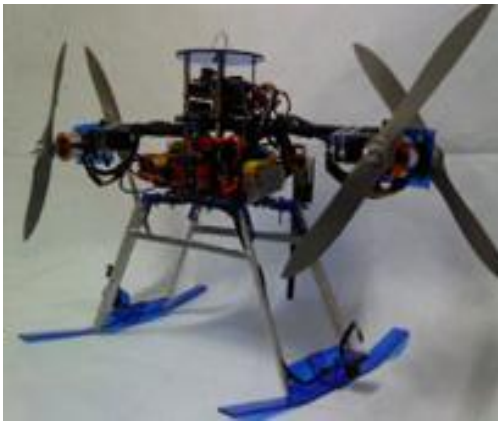
3.1 Bentuk Implementasi Robot

Pada Gambar 3.1 menunjukkan bentuk robot terbang yang telah dirancang dilihat dari berbagai sisi. Terdapat dua mode pada bentuk robot. Mode 1 yaitu keadaan saat arah propeller kearah depan robot. Pada mode ini robot berfungsi untuk menarik parasut kearah yang diminta oleh ground segment. Adapun mode 2 yaitu keadaan saat propeller menghadap kearah atas robot. Sehingga saat motor brushless bekerja, robot akan berfungsi sebagai *tandem rotor helicopter*. Perubahan arah propeller menggunakan dua buah motor servo digital yang diletakkan disetiap sisi robot.

Adapun penjelasan posisi-posisi komponen yang ada pada model robot terbang yang dibuat ini ditunjukkan pada Gambar 3.2 menunjukkan bagian-bagian robot jika dilihat dari arah depan, adapun Gambar 3.3 menunjukkan bagian-bagian robot jika dilihat dari arah belakang.



(a) Robot pada mode 1 tampak depan



(b) Robot pada mode 1 tampak samping

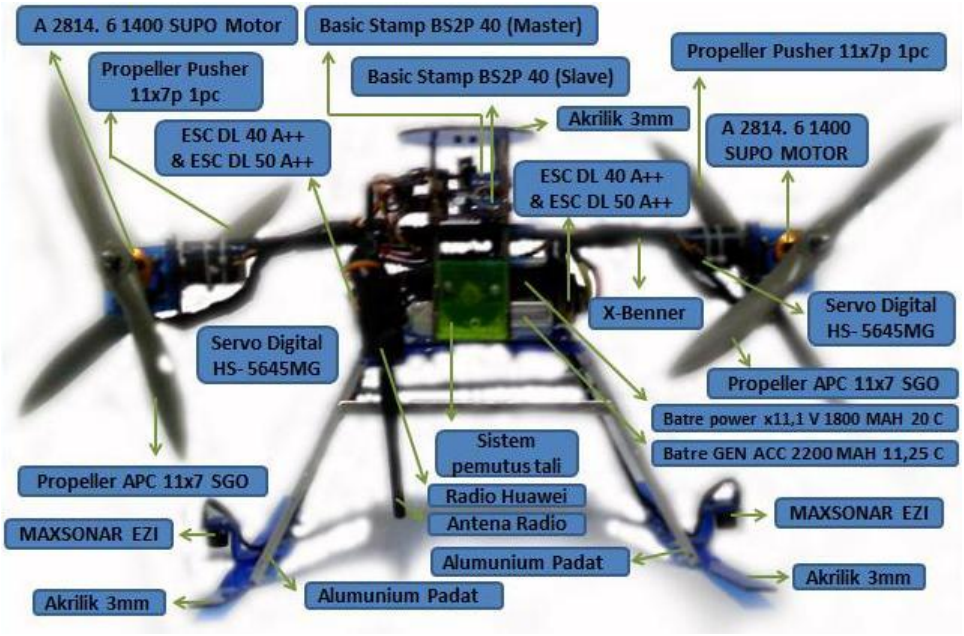


(c) Robot pada mode 2 tampak depan

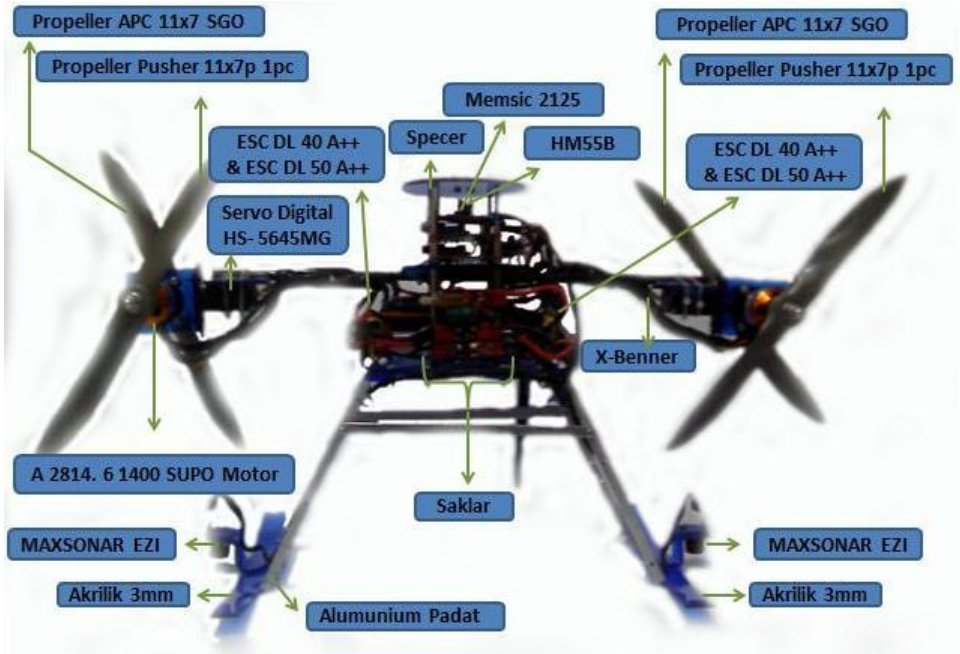


(d) Robot pada mode 2 tampak samping

Gambar 3.1 Hasil implementasi hardware yang telah dilakukan



Gambar 3.2 Posisi komponen-komponen pada robot dilihat dari arah depan



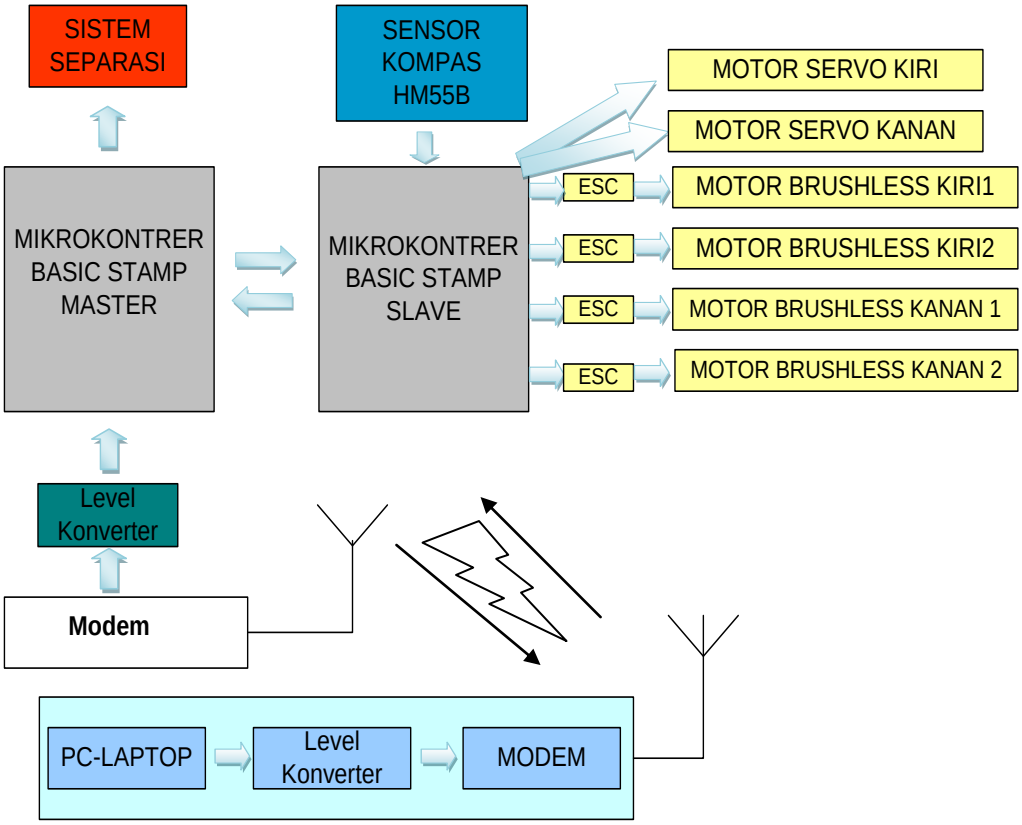
Gambar 3.3 Posisi komponen-komponen pada robot jika dilihat dari arah belakangPerancangan perangkat keras (*hadware)*

Tabel 3.1**.** Daftar material dan komponen robot

Bagian Robot	Material/Komponen	Keterangan
Kerangka utama	Akrilik, spacer, rangka x-banner	Kuat, cukup ringan dan mudah dibentuk
Sistem pengiriman data	Radio modem Huawei	Menggunakan komunikasi data UART dan cukup handal
Sensor	HM55B, memsic 2125, maxsonar	Daya jangkau maxsonar hingga 254 inci
Processor	Basic Stamp BS2p40	Bahasa pemrogramannya mudah dan sederhana
Sistem penggerak	Motor brushless, ESC, propeller 26 cm, motor servo digital	Magnet pada motor brushless tidak mengganggu kompas. Panjang propeller dipilih agar memperbesar daya angkat/daya dorong robot
Catu daya	Baterai Lipo 2200 mAh	Daya tahan cukup lama serta dapat melakukan charging dengan cepat

3.2 Blok Diagram Sistem

Pada perancangan perangkat keras akan menampilkan blok diagram dari robot yang akan diranacang dimana semua komponen sudah terintegrasi, Blok diagram sistem yang dirancang ditunjukkan pada Gambar 3.4 berikut. Terdapat dua buah mikrokontroller yang digunakan. Sebuah mikrokontroller, disebut sebagai Master, berfungsi untuk melakukan telemeteri, telecommand serta separasi. Sedangkan mikrokontroller lainnya, disebut sebagai Slave, berfungsi untuk melakukan navigasi menggunakan pengontrol PID. adapun blok diagram sistem perangkat keras adalah sebagai berikut.



Gambar 3.4 Blok diagram sistem robot terbang yang dirancang

Tugas utama robot dari awal hingga akhir, secara garis besar, adalah sebagai berikut.

1. Setelah catu daya diaktifkan, mikrokontroller akan masuk ke mode stand by dengan tugas menunggu sinyal aktivasi dari pengontrol di ground station. Saat mode stand by ini, robot akan mengatur arah propeller ke arah depan. Posisi arah propeller ini disebut mode 1, dimana pada mode ini robot berfungsi untuk menarik parasut ke arah setpoint.
2. Pada mode standby inilah paket bantuan bencana alam dengan robot yang telah dilengkapi dengan parasut dilempar.

3. Saat pengontrol di ground station mengirimkan sinyal aktivasi, ground station juga akan mengirimkan sinyal arah dalam derajat kepada robot.
4. Setelah robot menerima sinyal aktivasi, maka tiga pengontrol PID akan melakukan komputasi agar keempat motor brushless yang ada pada robot dapat menggerakkan robot berputar kearah yang diminta oleh ground station. Sebagai acuan arah, maka robot dilengkapi dengan sensor kompas.
5. Robot akan mengirimkan sinyal ke ground station berupa arah robot saat ini dalam derajat.
6. Jika pada ground segment diterima data bahwa arah robot saat ini telah berada pada kisaran arah yang diharapkan, maka ground segment dapat memerintahkan robot untuk maju.
7. Saat robot menerima perintah untuk maju, maka pengontrol PID pada robot akan mengatur agar motor brushless berputar lebih cepat tetapi tetap berada pada arah yang diharapkan. Putaran motor brushless yang lebih cepat diharapkan akan menarik parasut untuk maju kearah yang diharapkan.
8. Jika robot telah mendekati lokasi pendaratan bantuan bencana alam, maka ground segment dapat mengirimkan perintah untuk melakukan separasi.
9. Saat sinyal separasi diterima oleh robot, maka pertama-tama robot akan mengubah arah propeller ke arah atas, sehingga menjadi mode *tandem rotor helicopter*, disebut sebagai mode 2. Kemudian robot akan melepas parasut yang masih tersambung dengan paket bantuan

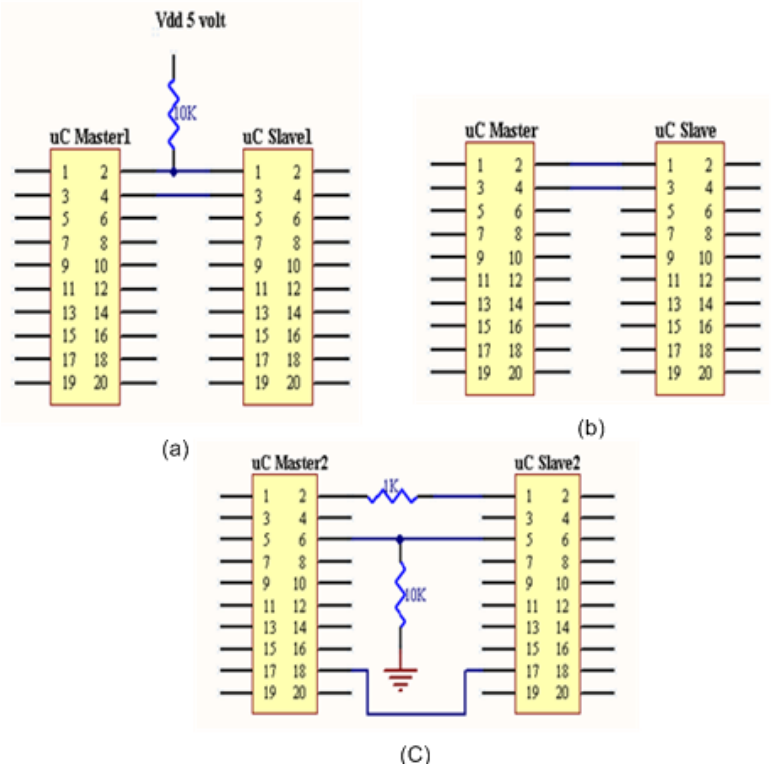
bencana alam agar parasut dan bantuan bencana alam tersebut jatuh tepat pada lokasi pendaratan.

10. Agar dapat mempertahankan keseimbangan robot pada mode *tandem rotor helicopter*, robot telah dilengkapi dengan gyroscope untuk mendeteksi kemiringan robot dan sensor ultrasonik untuk mengukur ketinggian terbang robot.
11. Pada pengembangan selanjutnya, diharapkan setelah robot bertansformasi menjadi mode *tandem rotor helicopter*, robot dapat dikendalikan oleh pengontrol ground station untuk terbang menuju lokasi pemberangkatan sehingga robot dapat digunakan berulang kali untuk pendistribusian paket lainnya.

3.2.1 Mikro Kontroler Master dan Mikro Kontroler Slave

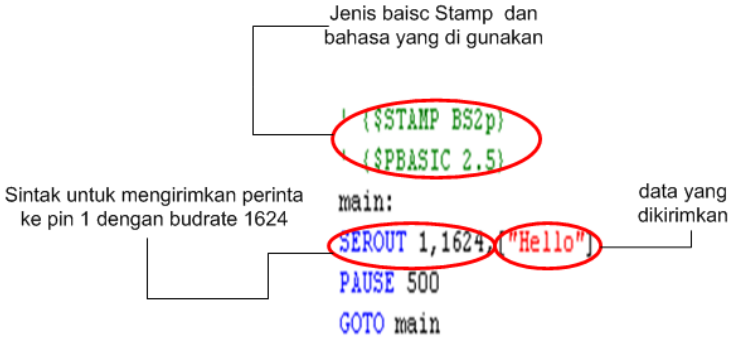
Mikrokontroller yang digunakan pada robot ini terdiri dari dua buah mikrokontroller Basic Stamp. Sebuah Basic Stamp, disebut sebagai Master, digunakan sebagai prosesor utama yang berfungsi mengirimkan dan menerima data dari *ground segment* serta melakukan separasi. Sementara Basic Stamp lainnya, disebut sebagai Slave, bertugas khusus untuk melakukan navigasi robot melalui pembacaan sensor dan mengatur gerakan motor. Alasan penggunaan dua buah mikrokontroller adalah agar proses telemetri dan telecommand tidak mengganggu proses pengontrolan PID. Jika hanya menggunakan sebuah mikrokontroller Basic Stamp saja yang berfungsi sebagai pengontrol PID dan juga melakukan proses telecommand, maka performansi pengontrol memburuk karena waktu keluaran sinyal pengontrol PID menjadi lebih lambat daripada respon time

sistem. Flowchart dari program mikrokontroller Master ditampilkan pada. Agar mikro kontroler master dan mikrokontroller slave dapat berkomunikasi maka hal yang perlu dilakukan adalah sebagai berikut:



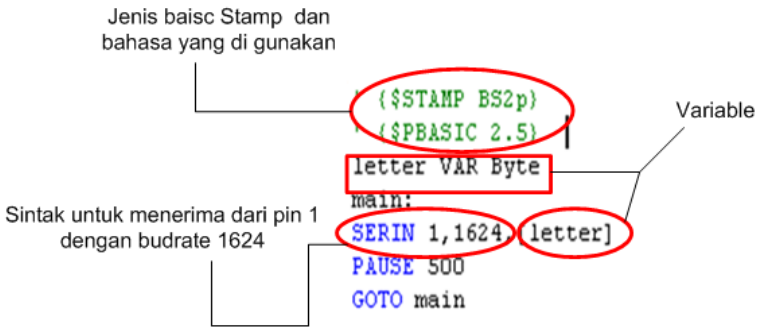
Gambar 3.5 Rangkaian komunikasi mikro kontroler Master dan Slave

Agar mikro kontroler master dapat berkomunikasi dengan *mikrokontroller slave* maka program yang di gunakan pada *mikrokontroler master* adalah sebagai berikut.

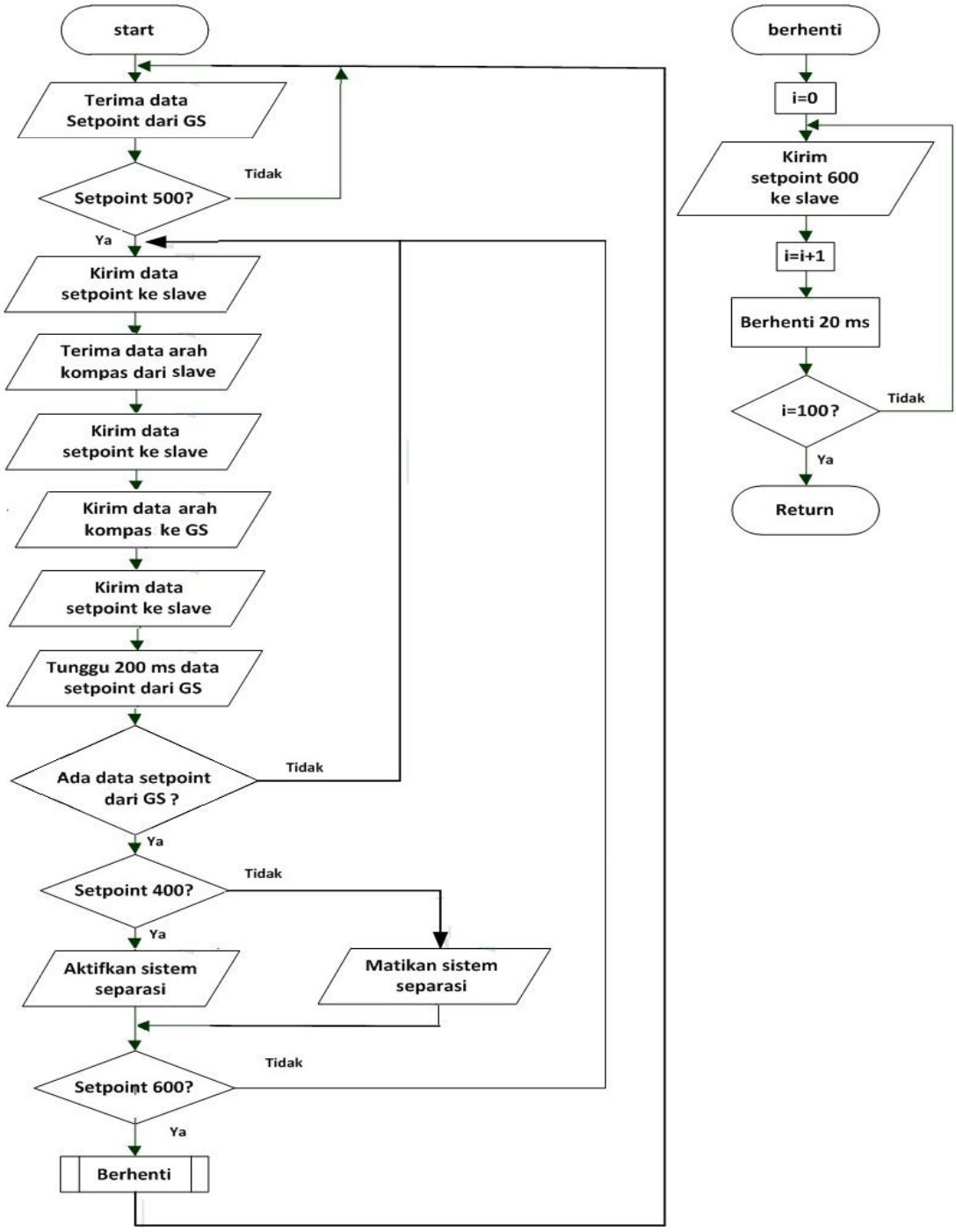


Gambar 3. 6 Contoh program pengiriman data dari *mikrokontroler Master* ke *mikrokontroler Slave*

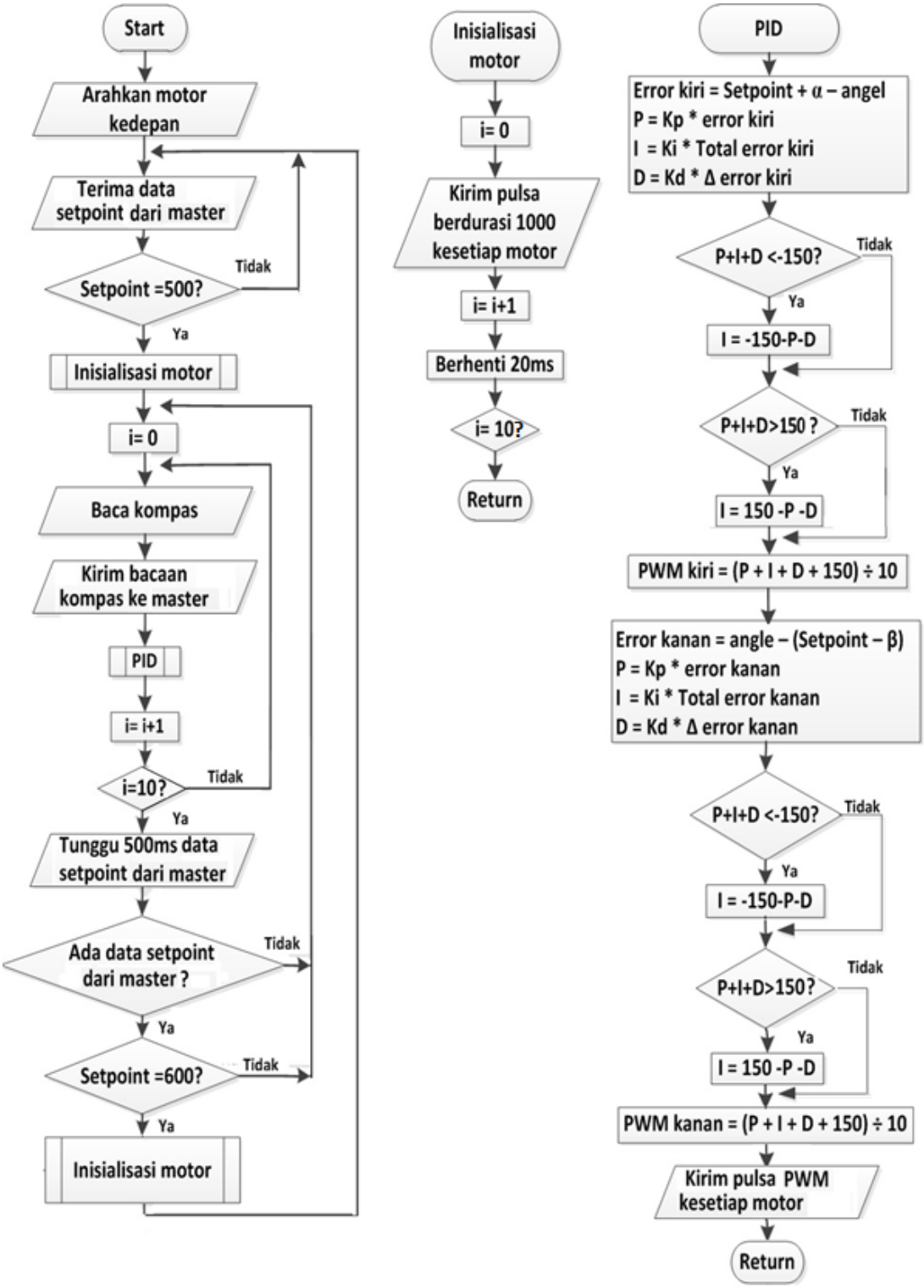
Berikut ini adalah conroh sintak sederhana yang didownload pada *mikrokontroller slave* agar dapat membaca data yang dikirimkan oleh *mikrokontroller master*.



Gambar 3.7 Contoh Program penerimaan data dari mikrokontroler *master*



Gambar 3.8 Flowchart program mikrokontroler *master*



Gambar 3.9 Flowchart program mikrokontroler *slave*

3.3 Sistem kendali PID

untuk melakukan pengontrolan motor agar motor dapat bergerak stabil dengan power yang sama maka di butuhkan pengontrolan PID. Sistem kontrol merupakan proses pengendalian *error* dengan cara memasukkan *error* tersebut kedalam input yang akan dibandingkan dengan sistem pengendalian. Tujuan untuk mengurangi *error* tersebut dan menghasilkan *output* atau keluaran yang sesuai dengan *set point* yang diberikan. Agar *set ponit* tercapai maka perancangan *payload* ini menggunakan pengontrolan *Proporsional Integral dan Derivatif* (PID). Secara umum, Berikut persamaan pengontrol PID yang diperlihatkan pada persamaan 3.1.

$$mv(t) = K_p\, e(t) + K_i \int_0^1 e(t)dt + K_d \frac{de(t)}{dt} \quad (3.1)$$

Dengan :

$$K_i = K_p \times \frac{1}{T_i} \text{ dan } K_d = K_p \times T_d$$

Keterangan :

$mv(t)$: *output* dari pengontrol PID atau *Manipulated Variable*

K_p : *gain proporsional*

T_i : *time integral*

T_d : *time derivatif*

K_i : *gain integral*

K_d : *gain derivatif*

Pada dasarnya aksi kontrol PID bertujuan untuk menggabungkan kelebihan komponen-komponen dasar PID, kelebihan komponen tersebut antara lain:

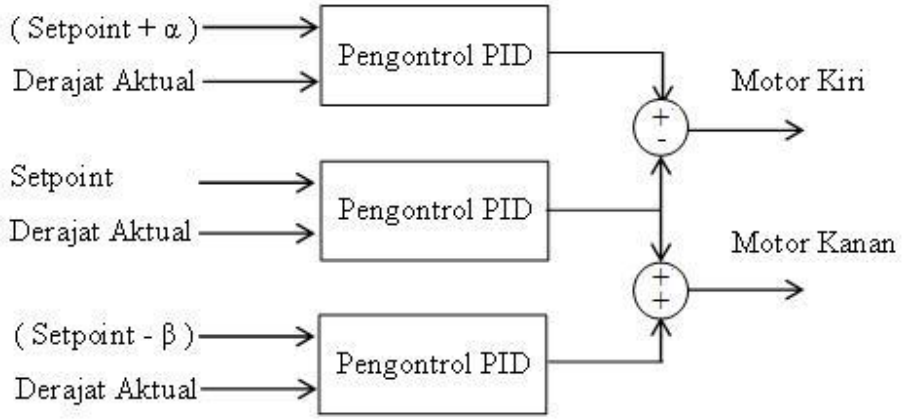
1. Pengontrol *proporsional* berfungsi untuk mempercepat respon. Ciri-ciri pengontrol *proporsional*:

a. Jika nilai Kp kecil, pengontrol *proporsional* mampu melakukan koreksi kesalahan yang kecil, sehingga akan menghasilkan respon sistem yang lambat (menambah *rase- time)*.
b. Respon sistem akan semakin cepat mencapai keadaan mantapnya (mengurangi *rase time*), jika nilai Kp dinaikkan.
c. Namun, 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 *steady state error*.

2. Pengontrol *Integral* berfungsi untuk menghilangkan *error steady*. Ciri-ciri pengontrol *integral*:
   a. Keluaran pengontrol *integral* membutuhkan selang waktu tertentu, sehingga pengontrol *integral* cenderung memperlambat respon.
   b. Keluaran pengontrol akan bertahan pada nilai sebelumnya, ketika sinyal kesalahan berharga nol.
   c. Keluaran akan menunjukkan kenaikkan atau penurunan yang dipengaruhi oleh besarnya sinyal kesalahan dan nilai Ki, jika sinyal kesalahan tidak berharga nol.
   d. Ki yang berharga besar akan mempercepat hilangnya nilai *offset*. Tetapi semakin besar nilai Ki akan mengakibatkan peningkatan osilasi dari sinyal keluaran pengontrol.
3. Pengontrol *derivatif* berfungsi untuk memperbaiki sekaligus mempercepat *respon transient*. Ciri-ciri pengontrol *derivatif*:

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*.

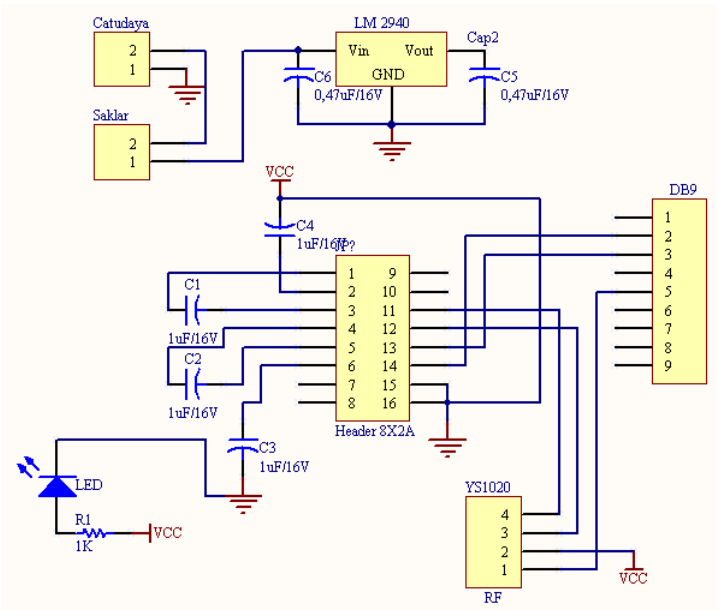
Pada mode dorong (mode 1), untuk mengendalikan arah robot melalui putaran empat motor brushless, maka digunakan sistem kendali Proportional Integral Derivative (PID). Pada perancangan ini tiga pengontrol PID digunakan, dimana diagram sistem kontrolnya ditampilkan pada Gambar 3.10. Pengontrol PID pertama berfungsi mengontrol motor kiri agar mengarahkan robot lebih kekanan dari set point. Sedangkan pengontrol PID ketiga berfungsi mengontrol motor kanan agar mengarahkan robot lebih kekiri dari set point. Akibatnya kedua motor akan saling dorong mendorong sehingga diharapkan resultant dari kedua dorongan motor tersebut mengakibatkan arah robot berada dikisaran set point yang diinginkan. Adapun pengontrol PID kedua berfungsi untuk melakukan intervensi untuk mengoreksi keluaran kendali PID pertama dan ketiga.



Gambar 3.10 Diagram sistem pengontrol pada robot

3.4 Pengubah Level Digital ke RS232 (MAX232)

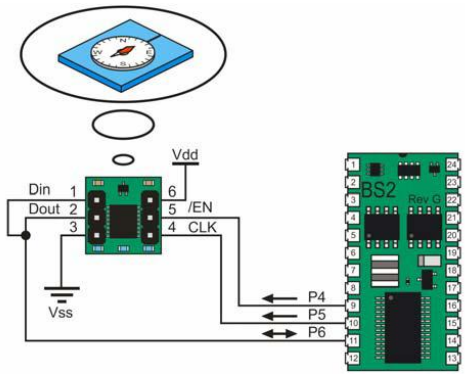
Rangkaian RS232 berfungsi untuk komunikasi antara modem dengan komputer. Komponen utama menggunakan IC MAX232, yaitu sebuah IC yang dapat mengubah format *digital* ke dalam sebuah format atau *level* RS232, dimana pada *level* RS232, tegangan *high* diwakili dengan tegangan +3 sampai +25 V. Diantara -3 dan +3 merupakan tegangan *invalid*. Rangkaian RS232 ditunjukkan pada Gambar 3.11.



Gambar 3.11Rangkaian RS232

3.4 Sensor Kompas (Hitachi HM55B)

Pada perancangan robot ini menggunakan hitachi HM55B sebagai sensor kompas. Pemilihan sensor ini didasari kemampuannya yang cukup handal dan kemampuan pembacaan data yang cepat, hanya 30 sampai 40 ms antara pengukuran awal dan data siap dibaca kembali. Tentunya pembacaan data yang cepat ini sangat dibutuhkan agar pencarian arah gerak *payload* dapat tercapai dengan baik. Penggunaan modul hitachi HM55B ini juga sangat sederhana karena tidak memerlukan rangkaian tambahan yang rumit. Rangkaian komunikasi modul hitachi HM55B dengan *Basic Stamp*, diperlihatkan pada gambar 3.12.

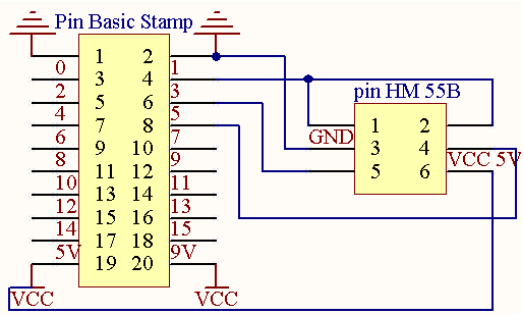


Gambar 3.12 Rangkaian komunikasi modul hitachi HM55B dengan Basic Stamp 2p40

Adapun maksud dari gambar 3.12 adalah cara pemasangan kaki kaki dari sensor kompas ke mikrokontroller basic stamp, agar tidak terdi kesalah maka pemasangan jangan sampai salah, terutama pada kaki Vdd dan Vss. Modul hitachi HM55B akan mengirimkan data axis x dan axis y, data kedua axis ini akan diolah oleh mikrokontroller *slave* yang kemudian diubah dalam satuan derajat dengan menggunakan persamaan di bawah ini.

$$Sudut\ \theta = \arctan\left(-(\text{axis y})\Big/(\text{axis x})\right) \qquad (3.1)$$

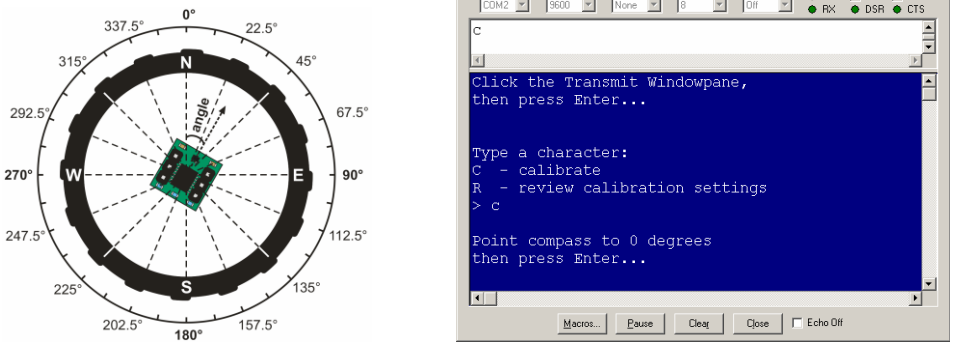
Berikut adalah adalah rangkaian yang digunakan untuk menghubungkan pin pada basic pada sensor kompas HM55B ke mikrokontroler basic stamp.



Gambar 3.13 Pemasangan pin sensor kompas ke *basic stamp*

3.4.1 Cara Kalibrasi Sensor Kompas

Sebelum sensor kompas di gunakan untuk pengukuran maka terlebih dahulu dilakukan pengkalibrasian agar tidak terjadi kesalahan pembacaan arah pada saat *seeking* (pencariaan arah) berikut adalah cara yang dilakukan untuk mengkalibrasi sensor kompas HM 55B.



(a) (b)

Gambar 3.14 (a) segment 16 arah kompas (b) tampilan program saat dilakukan pengkalibrasian pada sensor

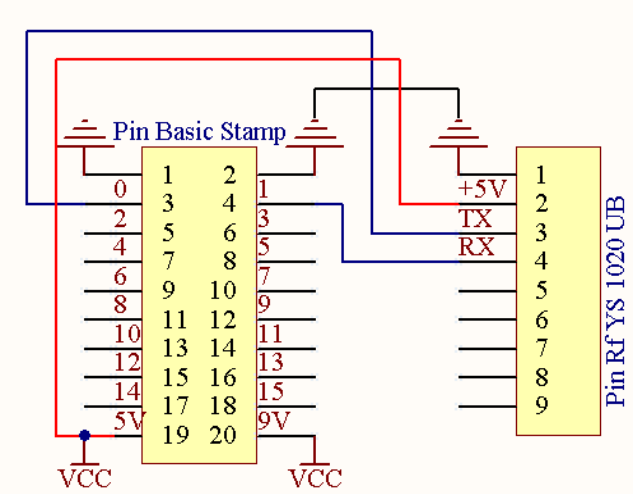
Berikut adalah langkah-langkah pengkalibrasian pada sensor kompas HM55B

1. Rangkai modul HM55B dengan *Basic Stamp* seperti pada gambar 3.13 pastikan catu daya terpasang sesuai dengan nilai *range* dari Vin modul HM55B.
2. Membuat cetakan gambar seperti pada gambar 310 (a).
3. Letakkan cetakan pada bidang datar dan sejajarkan ke utara magnetik dengan bantuan kompas magnetik.
4. Pastikan untuk mengatur magnetik kompas jauh dari hasil cetak sebelum melanjutkan poin berikutnya.
5. Sejajarkan modul HM55B ke utara magnetik dengan berbasis tepipapan operator anda dengan garis putus-putus.
6. Jalankan program yang terdapat pada lampiran A (diperoleh dari data sheet).
7. Pastikan klik kaca cendela transmit, seperti yang ditunjukkan pada gambar 3.14 (b) sebelum mengikuti perintah yang ditunjukkan pada terminal *debug prompt*.Program akan meminta anda untuk mengambil seri pengukuran pertama pada arah 90^{o} kenaikkan untuk mengumpulkan kompas *offset* . kemudian anda akan mengambil seri kedua pengukuran pada arah $22{,}5^{o}$ kenaikan untuk mengumpilkan informasi *interpolasi linier*. Kedua set pengukuran akan disimpan ke dalam *Basic Stamp* EEPROM.
8. Bila anda telah menyelesaikan proses kalibrasi, jalankan program yang terdapat pada lampiran B (diperoleh dari data sheet).

3.5 Radio Frekuensi (RF) Huawey YS1020UB

Radio YS1020 UB pada perancangan robot ini berfungsi sebagai media pengirim dan penerimaan data dari hasil pengolahan mikrokontroler yang bersumber dari hasil pembacaan sensor-sensor yang digunakan, selanjutnya radio YS1020UB akan mengirimkan hasil pembacaan sensor ke ground segment, dan dari ground segment kita dapat mengontrol robot sesuai keinginan dengan

mengirim perintah dari ground segment ke robot melalui radio YS 2020UB, adapun cara pemasangan radio ke mikrokontoler dapat dilihat pada Gambar 3.15 dan program yang dirancang untuk komunikasi dari mikro kontroler ke Radio YS1020 dapat di lihat pada Gambar 3.16 untuk menerima data, dan untuk mengirimkan data dapat di lihat pada Gambar 3.17.



Gambar 3.15 Rangkaian basic ke radio YS1020 UB

Contoh sintak program yang di gunakan adalah sebagai berikut.

```
' {$STAMP BS2p}
' {$PBASIC 2.5}
angle VAR Word   'Variable
angle=300        'variable
 AUXIO           '0 berada di pin auxio basic stamp
  SEROUT 0,240,[DEC3 angle]     'mengirim data ke pin 0 sebanyak 3 angka desimal
  MAINIO                        'selanjutnya kembali ke pin mainio
  RETURN
```

Gambar 3.16 Contoh program pengiriman data

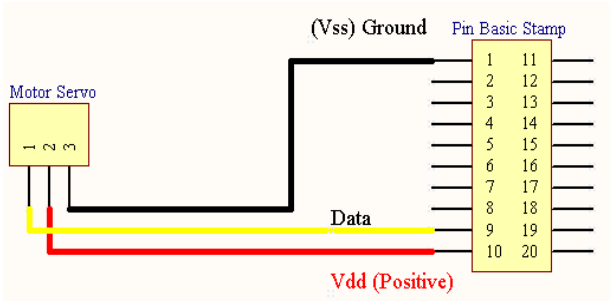
Contoh program penerimaan data dari bs yang akan di tampilkan ke ground segment adalah sebagai berikut:

```
' {$STAMP BS2p}
' {$PBASIC 2.5}
commad VAR  Word
command VAR Word
standby:
  DEBUG "AYE"
  'pada kondisi stanby saat di program muncul kata AYE
  SERIN 1,240,[WAIT("R0500")]
 'terima data dari pin 1 tunggu R500 dengan boudrate 240
```

Gambar 3.17 Contoh program terima data dari mikro melalui radio YS1020 UB

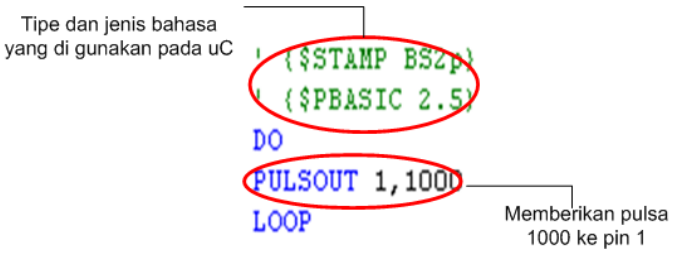
3.8 Motor Servo

Pada perancancangan model robot terbang sebagai media pendistribusian paket bantuan ke daerah terisolir terdapat dua buah motor servo yang di letakkan di masing-masing lengan robot, yang fungsinya adalah sebagai penubah mode robot dari mode dorong ke mode angkat (terbang), motor servo yang di gunakan adalah motor servo digital HS5645MG yang memiliki torsi ±10kg. berikut adalah contoh berupa gambar cara penghubungan motor servo ke mikrokontroller Basic Stamp.



Gambar 3.18 Contoh rangkaian motor servo

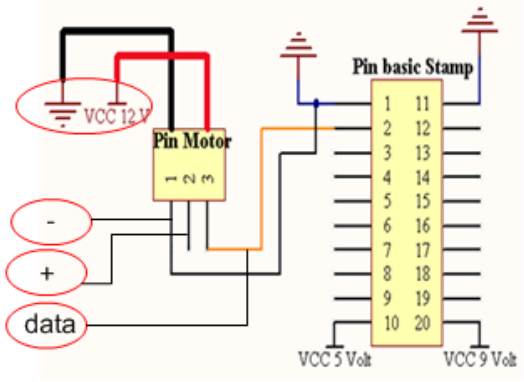
Agar motor servo dapat di atur sesuai kebutuhan maka perlu dilakukan pemograman, berikut adalah contoh program sederhanya untuk mengatur sudut pada motor servo. Di mana Pada motor servo pulsa minimal di mulai dari 1000 sampai 2800.



Gambar 3.19 contoh program pada motor servo

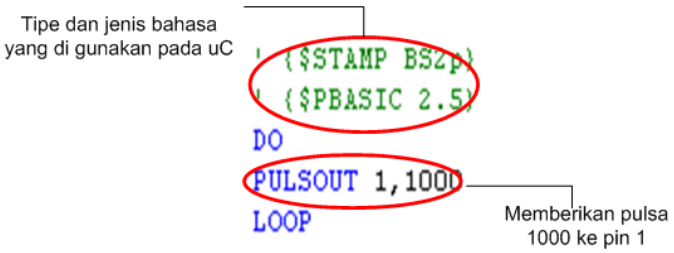
3.9 Motor Brushless

Pada perancangan model robot terbang ini sistem navigasinya menggunakan empat motor brusless yang fungsinya adalah untuk menarik parasut dan paket bantuan *(mode1)* dan mode terbang yang berfungsi setelah melakukan separasi *(mode2)* . Adapun cara pemansangan motor brushless ke mikrokontroller adalah sebagai berikut



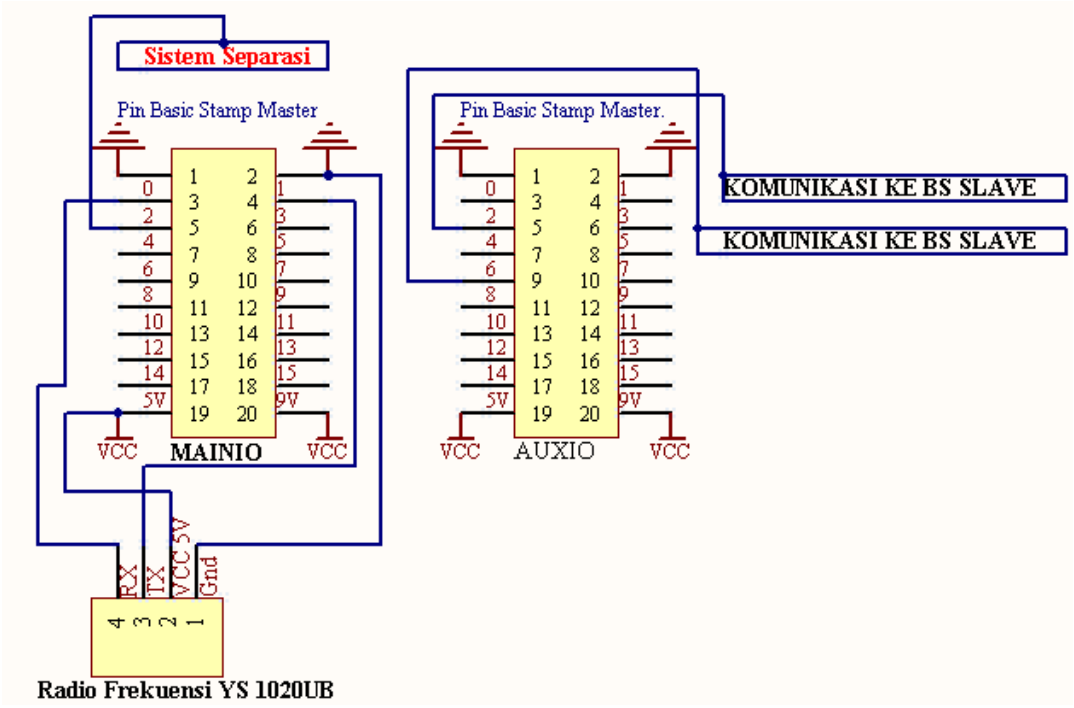
Gambar 3.20 Pemasangan pin motor servo ke basic stamp

Berikut adalah sintak program yang digunakan untuk pemograman motor brushless, di mana pulsa minimal mulai dari 1000 sampai 2000.

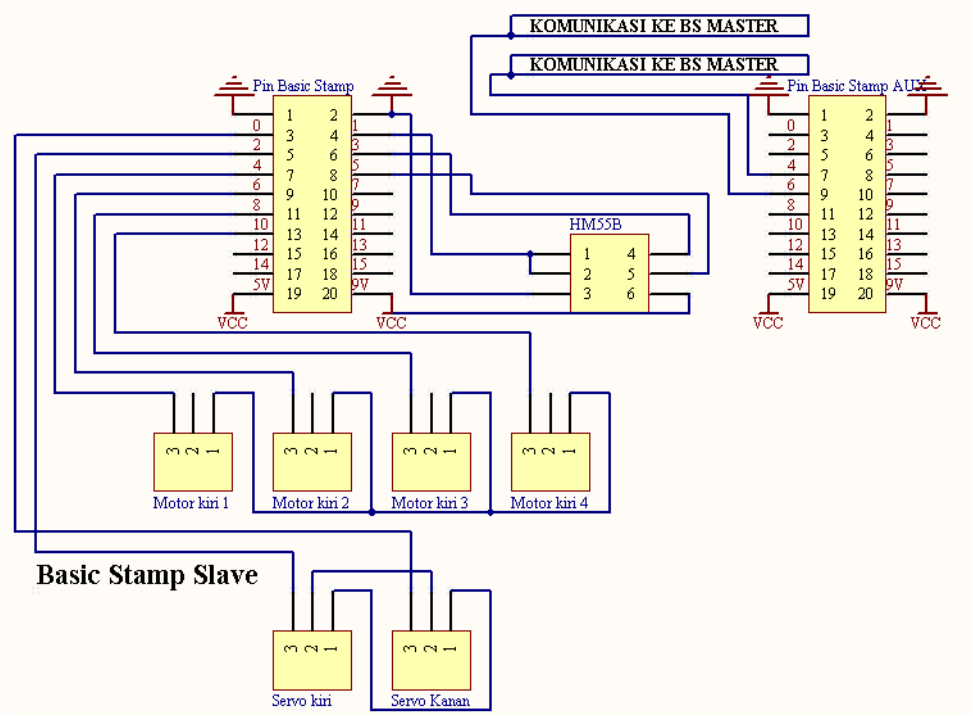


Gambar 3.21 contoh program untuk mengaktifkan motor

3.5 Blok diagaram Keseluruhan Rangkaian



Gambar 3.22 Skema Rangkaian pada Basic Stamp Master

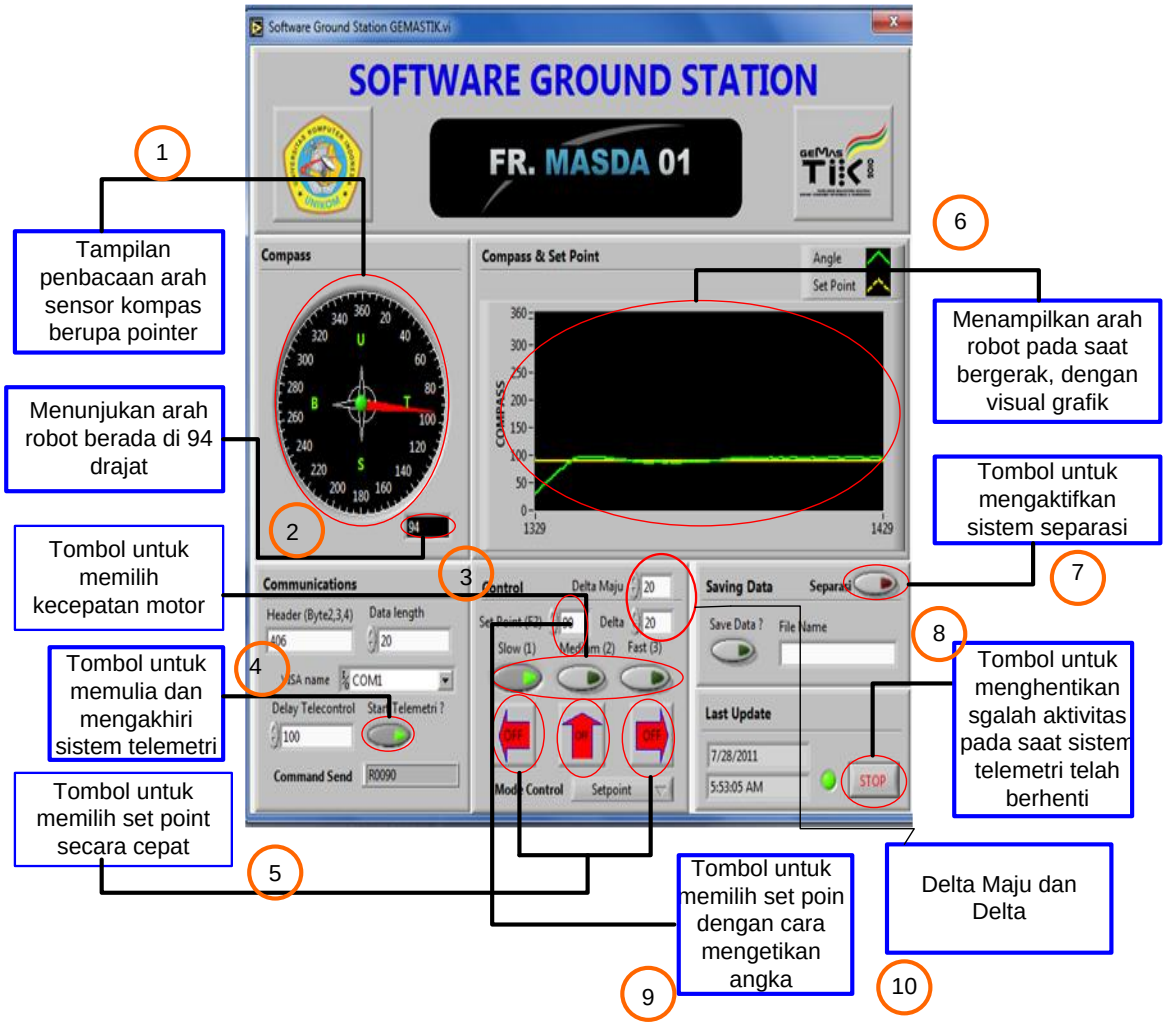


Gambar 3.23 Skema Rangkaian pada Basic Stamp Slave

3.7 Perancangan Sofware

Adapun sofware yang digunakan untuk monitoring robot adalah Labview. LabVIEW adalah salah satu bahasa pemograman komputer grafik yang menggunakan icon-icon sebagai pengganti teks dalam membuat aplikasi. Program ini adalah salah satu produk keluaran National Instrument.

Ground segment berfungsi sebagai monitoring keadaan robot pada saat diaktifkan, dimana robot juga dapat dikontrol dari ground segment serta hasil pembacaan sensor dapat di tampilka ecara real time pada ground segment. Berikut adalah tampilan dari ground segment di lengkapi dengan pilihan tombol yang berfungsi untuk mengaktifkan dan mengontrol robot.



Gambar 3.24 Tampilan dari gound station

Berikut ini adalah penjelasan dari masing- masing fungsi tombol (fitur) yang terdapat pada ground segment yang akan digunakan:

Tabel 3.2 Keterangan dan Fungsi Tombol (Fitur) dari *ground segment*

NO	Fitur	Keterangan
1	Tampilan Kompas	Berfungsi untuk melihat pergerakan robot secara pointer berada pada posisi berapa derajat.
2	94	Arah robot berada di 94 drajat, bisa langsung di lihat pada saat robot diaktifkan dan tapil angka desimal pembacaan sensor.
3	Tombol slow,medium,fast	Tombol yang berfungsi untuk mengatur kecepatan pada motor brushless
4	Tombol Star Telemetri	Untuk mengaktifkan, dan memutuskan proses pengiriman dan kontrol robot dari ground segment.
5	Tombol panah kiri dan kanan	Berfungsi untuk mengubah arah set poin secara tobol, tanpa perlu mengetikan arah
6	Display grafik	Berfungsi menampilkan arah robot dalam brntuk grafik.
7	Tombol Separasi	Tombol yang digunakan untuk mengaktifkan sistem separasi, yang akan memutuskan tali parasut
8	Tombol Stop	Tombol yang berfungsi untuk mengakhiri segala proses telemetri ataupun proses untuk mengirim data.
9	Setpoint	Tempat untuk mengetikan arah robot yang diinginkan .
10	Delta 20 dan Delta Maju 20	Delta 20 berfungsi untuk mengubah arah setpoint pada saat tombol panah kiri dan kanan di tekan maka setpoin akan berubah dengan kelipatan 20. Delta maju 20 berfungsi untuk memberikan pulsa terhadapap motor pada saat kondisi low, medium, fast dan saat tombol panah atas ditekan maka robot akan maju dalam 3 tahap dan pertahap pulsa yang diberikan akan naik 20 ke masing- masing motor.