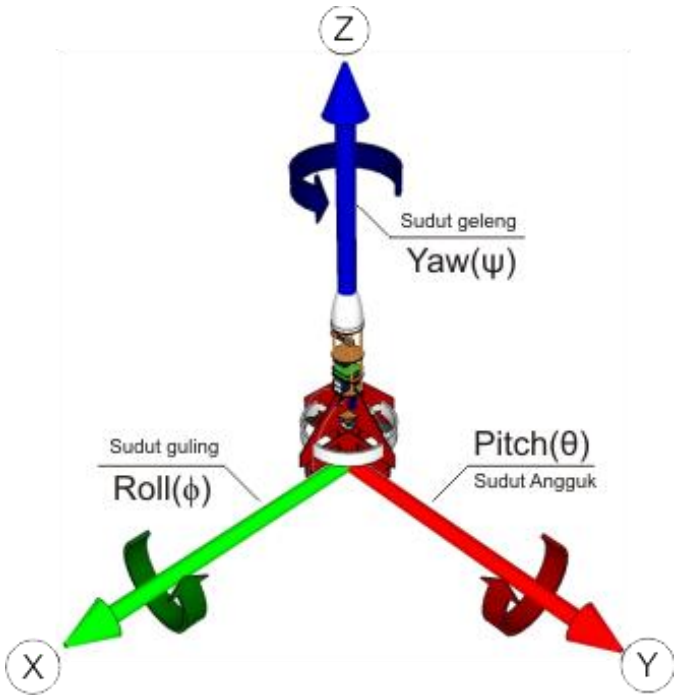
# BAB III
# PERANCANGAN SISTEM

Bab ini akan membahas tentang perancangan sistem untuk mengatasi efek contra-rotating yang terjadi pada roket motor elektrik dengan menggunakan bantuan sensor IMU. Sebelum melanjutkan pada perancangan tiap roket, berikut ini gambar III-1 merupakan ilustrasi dari *attitude* atau perilaku yang akan menjadi bagian keseluruhan dari perancangan sistem.




**Gambar III-1 Sumbu perilaku roket**

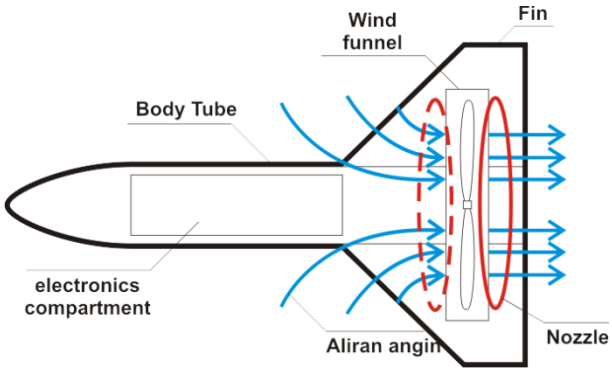
## 3.1 Kontrol mekanisme

Sebelum masuk kepada perancangan algoritma kontrol suatu sistem, proses pengotrolan suatu perangkat perlunya adanya kesinambungan antara fisik alat dengan algoritma kontrolnya. Jika kita ambil contoh pada pengontrolan kecepatan mobil, kita perlu mengetahui bebebrapa hal yakni berapa besar putaran yang dapat dihasilkan dengan sejumlah bahan bakar, berapa besar jari-jari dari roda mobil serta faktor-faktor lainnya.

### 3.1.1 Komponen penyusun dan rancangan mekanik

Proses pengontrolan dapat dilakukan setelah kita mengetahui komponen penyusun dari suatu benda. Pada subbab (2.1.5) telah dijelaskan komponen-

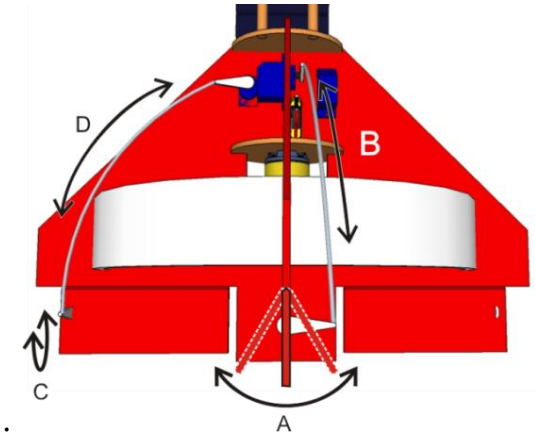
penyusun dari roket propelan. Berikut ini komponen penyusun dari roket motor elektrik.




**Gambar III-2 Komponen mekanik roket motor elektrik**

- *Electronic compartement* berupa tempat penyimpanan sistem elektronik pendukung kontrol roket yang berisikan beberapa perangkat elektronik seperti mikrokontroler, radio komunikasi dan sensor.
- *Igniter* pada roket propelan bertugas untuk pengontrol kecepatan. Pada roket motor elektrik berupa *driver* motor yang mengotrol laju pergerakan pada motor elektrik yakni komponen *electronic speed control(ESC).*
- *Body tube* pada roket motor elektrik akan berupa lembaran styrofoam. Daya dorong dari roket motor elektrik tidak sebesar roket propelan yang dapat melebihi perbandingan massa dan daya dorong hingga 5:1 atau selebihnya. Oleh karena itu, bahan ini dipilih karena massanya yang ringan serta mudah untuk dibentuk.
- *Propellant* pada roket merupakan sumber energi untuk menghasilkan daya dorong. Pada roket motor elektrik, baterai Lithium-ion Polymer(LiPO) akan menjadi sumber energi pendorong roket. Pemilihan jenis baterai ini didasari oleh kemampuan baterai untuk mensuplai arus yang besar dengan ukuran baterai yang kecil.
- *Wind Funnel* merupakan tempat pengumpulan udara yang dipakai sebagai energi pendorong oleh motor elektrik. Pada roket propelan bagian ini memiliki kesamaan dengan *combustion chamber* yang merupakan tempat dimana terjadi pembakaran propelan.
- *Nozzle* pada roket motor elektrik merupakan cerobong pembuangan udara yang telah dihisap pada bagian *Combustion Chamber*.

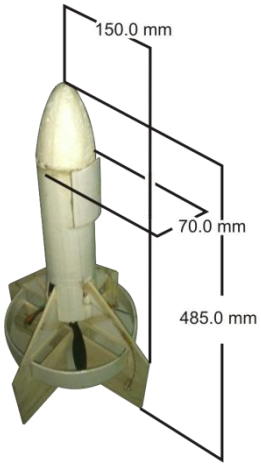
- *Fin* atau ekor roket merupakan bagian terpenting dalam proses kontrol roket yang akan dibuat. Pemilihan terhadap teknik kontrol yang mampu mengatasi masalah putaran pada badan roket menjadi pilihan utama. Pada subbab (2.1.6) terdapat tabel II.3 yang memaparkan tentang beberapa macam teknik pengontrolan roket. Berdasarkan tabel tersebut, teknik pengontrolan roket yang akan dipilih adalah teknik *jet vanes* dan *reaction wheels* dikarenakan kedua teknik tersebut memiliki mekanisme kontrol pada sudut putaran badan roket. Pemilihan *reaction wheels* memang dapat dikatakan lebih baik dari pada *jet vanes* karena penyaluran tekanan udara yang penuh. Namun dikarenakan pertimbangan mengenai daya dorong roket motor elektrik yang kecil maka beban dari roket harus diminimalisir. Oleh karena itu, pilihan teknik kontrol akan dibebankan pada *jet vanes* yang tidak memerlukan tambahan beban yang besar. Selain itu, pertimbangan ini diambil dikarenakan faktor perubahan aliran udara dapat diperhitungkan karena *jet vanes* hanya mendapat aliran udara secara terus menerus dari *nozzle roket* sehingga proses penerapan kontrol akan lebih sederhana.




**Gambar III-3 Penerapan mekanisme jet vanes**

Pada gambar III-2 diperlihatkan ilustrasi penerapan jet vanes dengan memanfaatkan *rotary actuator*. Pada saat rotary actuator berubah maka penghubung mekanik pada poin B menarik atau mendorong ekor roket poin A yang menyebabkan perubahan sudut ekor roket. Begitupula yang terjadi pada poin D dan C.

Berikut ini ilustrasi yang akan memaparkan rancangan mekanik dari spesifikasi roket:

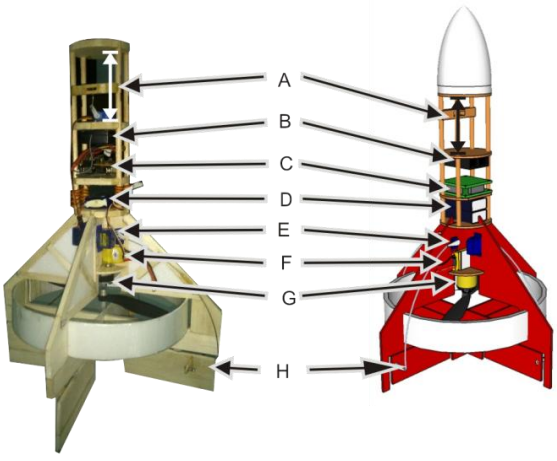



**Gambar III-4 Rancangan dimensi roket**

Dengan spesifikasi sebagai berikut

- Diameter *tube* : 70 mm
- Tinggi : 485 mm
- Lebar : 270 mm
- Massa : 300 gr

Dan penempatan komponen sebagai berikut.




**Gambar III-5 Rancangan penempatan komponen**

Berikut ini adalah keterangan dari gambar III-5:

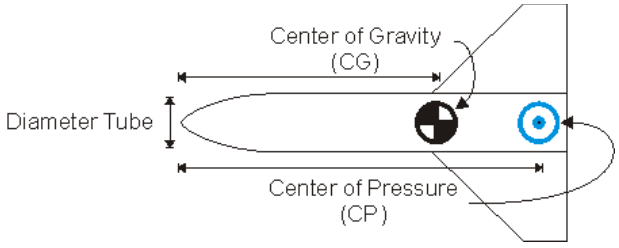
A. Ruang penyimpanan parasut yang akan digunakan dalam proses *recovery* atau penyelamatan badan roket.
B. *Remote receiver* digunakan untuk menangkap komando yang diberikan oleh *remote* kontrol.

C. Mikrokontroler, sensor dan radio komunikasi akan dibuat menjadi suatu sistem *embedded*.
D. Baterai LiPO sebagai catu daya untuk mengaktifkan sistem.
E. Motor Servo sebagai motor aktuator penggerak ekor dari roket.
F. ESC(Electronic Speed Control) berperan sebagai *driver* motor brushless.
G. Motor Brushless berperan untuk memberikan roket daya angkat.
H. Fin aktif sebagai mekanisme dalam proses kontrol stabilisasi roket.

### 3.1.2 Stabilitas roket

Suatu roket dianggap layak uji terbang disaat roket telah memiliki kestabilan yang cukup. Hal tersebut akan bergantung pada penempatan komponen-komponen penyusun roket yang akan mempengaruhi pusat massa dan pusat tekanan pada roket seperti yang dijelaskan pada subbab(2.1.6) tentang stabilitas roket. Oleh karena itu perlu adanya perencanaan terhadap posisi penempatan komponen penyusun roket.

Berdasarkan hasil perancangan pada subbab (3.1.1) tentang perancangan mekanik, *software* Rocksim digunakan untuk mengetahui tingkat stabilitas dari roket. Teknik penghitungan menggunakan metode barrowman dengan hasil sebagai berikut.




**Gambar III-6 *Center of Gravity* dan *Center of Pressure***

*Center of Gravity* : 325,8 mm
*Center of Pressure* : 455,7 mm
Diameter *tube* : 70 mm

$$Static\ Margin = \frac{Center\ Pressure - Center\ Gravity}{Diameter\ tube} = \frac{455{,}7\ mm - 325{,}8\ mm}{70\ mm}$$

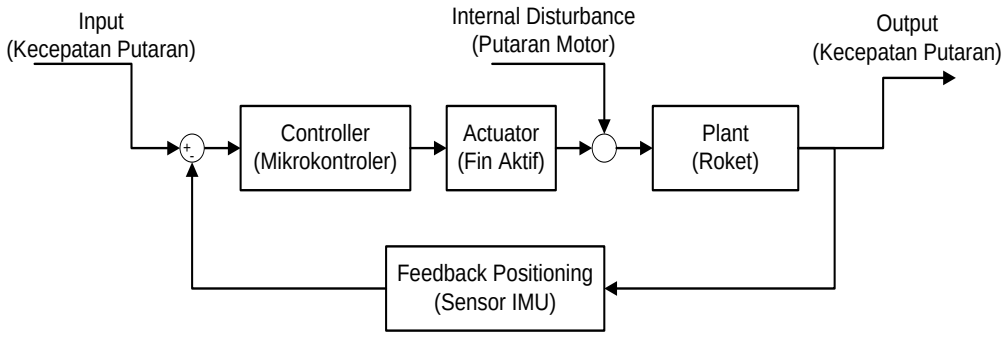
$$= \frac{129{,}9\ mm}{70\ mm} = 1{,}855$$

Berdasarkan dengan tabel II-1.

Analisis: $1 \leq 1{,}855 \leq 2{,}5\ artinya\ roket\ stabil$

### 3.1.3 Kontrol roket

Setelah proses pendesainan mekanik dilakukan maka proses selanjutnya yang akan dilakukan adalah proses kontrol sistem. Pada subbab(2.1.1) telah dijelaskan bahwa sistem kontrol terbagi menjadi 2 macam yakni *closed-loop* dan *open-loop*. Untuk melakukan proses kontrol, kecepatan putaran diperlukan untuk mengatur besar defleksi yang diperlukan untuk menghasilkan momen aerodinamis untuk mengurangi gangguan internal yang berasal dari putaran motor pendorong. Hal tersebut menyebabkan perlu adanya proses timbal balik terhadap besar gangguan yang terjadi dalam pendesainan kontrol sistem. Oleh karena itu, kontrol *closed-loop* dipilih sebagai jenis sistem kontrol yang digunakan karena adanya *feedback* atau timbal balik yang mempengaruhi proses kontrol. Pendesainan sistem dari sudut pandang kontrol memiliki diagram blok seperti berikut ini.

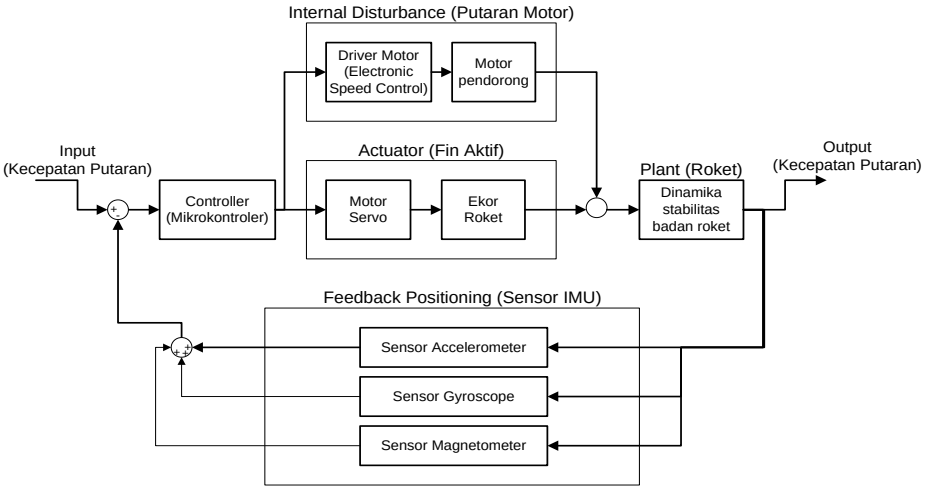



**Gambar III-7 *Closed-loop* roket motor elektrik**

Berikut ini penjelasan diagram blok mengacu pada subbab (2.1.1).

- ***Controlled variable* dan *Manipulated variable***. Pada topik ini, perubahan dari sudut defleksi dari ekor roket yang akan menjadi *Manipulated variable*. Untuk *controlled variable* akan berupa putaran dari badan roket yang menjadi keluaran dari roket.
- ***Plant.*** Pada pembahasan topik tugas akhir ini, *plants* yang akan dikontrol merupakan objek fisik dari badan roket.
- ***Disturbances.*** Pada topik ini, *internal distubance* yang dihasilkan oleh putaran motor pendorong yang akan menjadi konsen pembahasan.
- ***Feedback Control.*** *Feedback control* diartikan sebagai suatu operasi yang memiliki gangguan yang cenderung dikurangi dengan mengukur perbedaan diantara keluaran dan referensi sistem.

Dengan melakukan pendekatan terhadap perangkat keras maka diagram blok keseluruhan pada sistem menjadi seperti berikut ini.




**Gambar III-8 Diagram blok kontrol dengan pendekatan perangkat keras**

***Controller*.** Proses kontrol dilakukan oleh kontrol unit berupa mikrokontroler. Pada mikrokontroler akan ditanamkan algoritma kontrol yang akan mengontrol aktuator. Selain itu, mikrokontroler memiliki tugas lain seperti proses pembacaan dan pengolahan sensor yang akan mempengaruhi kinerja kontrol sistem.

***Actuator***. *Actuator* atau aktuator merupakan suatu mekanisme yang akan memperngaruhi kerja sistem. Sesuai dengan penjelasan pada subbab (3.1.1), aktuator yang akan digunakan berupa *jet vanes*. Oleh karena itu, *rotary actuator* diperlukan untuk mengatur perubahan sudut dari ekor roket berupa motor servo.

***Disturbances.*** *Internal distubance* yang dihasilkan oleh putaran motor pendorong yang akan menjadi konsen pembahasan. Putaran motor tidak dapat dihasilkan bila tidak adanya komponen pengontrol motor atau *driver motor*. Pada subbab (2.1.5) tentang efek *contra-rotating* yang ditimbulkan oleh putaran motor untuk menghasilkan daya dorong. Aksi dan reaksi yang terjadi menimbulkan putaran yang menjadi gangguan pada sistem navigasi dan stabilisasi.

***Plants***. *Plants* pada sistem merupakan badan roket yang telah memiliki kestabilan yang telah hitung pada subbab(3.1.2). Untuk dinamika dari roket tidak akan dijelaskan lebih mendalam sesuai yang tercantum pada batasan masalah.

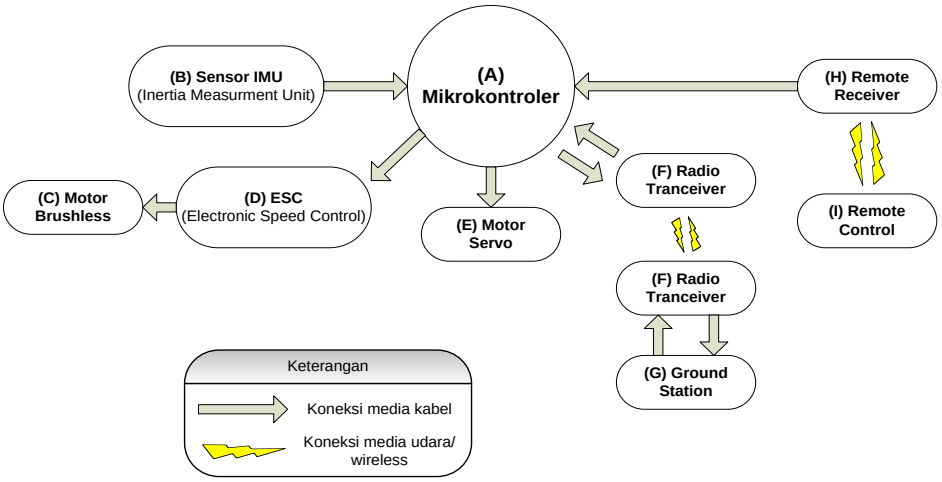
***Feedback.*** *feedback* merupakan nilai timbal balik dari adanya perubahan sistem yang terjadi. Pada kasus kali ini karena faktor perubahan posisi yang diperlukan maka blok *feedback* dapat kita sebut dengan *feedback positioning*.

Namun untuk menghasilkan nilai pembacaan perubahan posisi plant yang baik maka proses pengolahan sensor haruslah dilakukan. Proses pengolahan sensor tidak hanya sebatas pada pembacaan dan konversi kepada nilai. Salah satu contoh lainnya adalah proses filterisasi pada sensor. Proses filter diperlukan karena adanya kelemahan yang dimiliki komponen sensor. Setelah proses filter selesai dilakukan maka hasil keluaran pada tiap sensor akan digabungkan dengan menggunakan *complimentary filter* untuk menghasilkan perilaku roket.

## 3.2 Perancangan Perangkat Keras

### 3.2.1 Diagram blok perangkat keras dan pertimbangannya

Pada perancangan perangkat keras ini diperlukan untuk menjabarkan korelasi atau keterkaitan tiap-tiap perangkat yang akan menunjang sistem yang akan dibuat. Berikut ini diagram blok sistem perangkat keras yang akan dibangun.




**Gambar III-9 Diagram blok perangkat keras**

Berikut ini penjelasan dari blok-blok pada gambar III-9

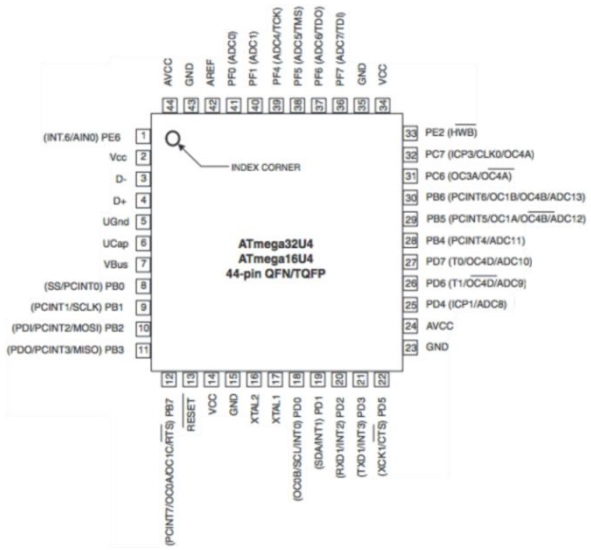
a) Mikrokontroler

Mikrokontroler ini akan digunakan untuk menanamkan algoritma kontrol untuk memenuhi tujuan dari topik ini. Selain berisikan algoritma kontrol, mikrokontroler akan digunakan untuk melakukan beberapa tugas lain seperti menerima data dari *remote* kontrol, mengolah data sensor dari sensor IMU, mengirimkan data sensor kepada *ground control station* dan mengatur kecepatan aktuator. Mikrokontroler ATmega merupakan salah satu mikrokontroler keluaran perusahaan ATMEL yang menggunakan beberapa bahasa pemrograman contohnya Basic, C, dan assembler.

Mikrokontroler yang digunakan pada perancangan alat ini adalah mikrokontroler jenis ATmega32u4 yang menggunakan pemrograman bahasa C. Mikrokontroler ini menggunakan arsitektur RISC sehingga tiap instruksi hanya membutuhkan waktu 1 *clock*. Pemilihan mikrokontroler jenis ini didasari kemampuannya yang cukup handal dan memiliki spesifikasi sebagai berikut.

- 32 Kbyte *programmable flash* yang ukurannya dianggap cukup dan tidak terlalu berlebih untuk algoritma yang akan ditanamkan.
- 1 Kbyte EEPROM yang dapat digunakan untuk menyimpan hasil kalibrasi dari beberapa sensor.
- Tersedia komunikasi UART RS-232 yang mendukung beberapa jenis radio komunikasi.
- Mendukung protokol I2C yang digunakan untuk berkomunikasi terhadap sensor-sensor.
- Memiliki fitur interupsi yang akan digunakan sebagai media untuk membaca data *remote* yang berupa pulsa dan menghasilkan PWM untuk memenuhi kebutuhan pin PWM untuk motor aktuator.
- Jalur I/O sebanyak 26 port yang dianggap telah memenuhi kebutuhan jumlah pin I/O yang digunakan.
- Tegangan 2,7-5,5 VDC.
- Kecepatan prosesor dapat mencapai 16 MHz.

Berikut ini alokasi pin yang dimiliki mikrokontroler ATmega32u4




**Gambar III-10 Alokasi pin ATmega32u4**

b) Sensor *Inertia Measurment Unit*(IMU)

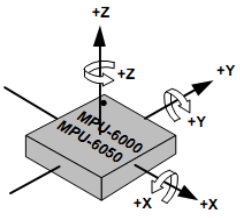
Sensor *Inertia Measurment Unit*(IMU) adalah sensor yang digunakan untuk mengukur gaya inersia yang terjadi pada benda. Sensor IMU terdiri dari beberapa gabungan sensor yaitu sensor *accelerometer*, *gyroscope* dan *magnetometer*. Pada sistem yang dirancang, sensor IMU akan memberikan hasil pengukuran sensor yang diolah menjadi data perilaku roket kepada perangkat mikrokontroler. Sensor yang digunakan dalam perancangan ini adalah sensor MPU6050 dan HMC5883L. Sensor MPU6050 merupakan gabungan dari sensor *accelerometer* dan *gyroscope*. Beberapa hal yang menjadi pertimbangan dalam penggunaan sensor ini adalah telah tertanamnya *low-pass filter* yang diperlukan untuk mengatasi *noise* yang telah dijelaskan pada bagian sensor pada bab 2. Selain itu, sensor MPU6050 mendukung protokol komunikasi I2C. Kedua faktor tersebut dianggap dapat membantu meringankan beban kerja mikrokontroler sebagai pusat kendali. Berikut ini spesifikasi yang dimiliki oleh sensor MPU6050 sebagai *gyroscope*.

- Skala pembacaan mulai dari ± 250º/detik hingga ± 2000º /detik.
- Memiliki resolusi sebesar 16 bit untuk meningkatkan keakuratan data untuk skala pembacaan tertinggi dapat menghasilkan keakuratan 0,61 º/detik untuk tiap bit-nya.
- Dapat menghasilkan data hingga 8000 sampel perdetik.

Berikut ini spesifikasi yang dimiliki oleh sensor MPU6050 sebagai *accelerometer*.

- Memiliki skala pembacaan mulai dari ± 2g, ± 4g, ± 8g, ± 16g,
- Memiliki resolusi sebesar 16 bit untuk skala pembacaan tertinggi dapat menghasilkan keakuratan 0,0191 $m/s^2$ untuk tiap bit-nya.

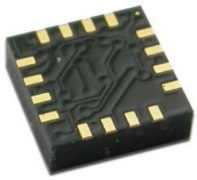
Sumber tegangan yang diperlukan oleh sensor ini relatif tidak terlalu besar yaitu 2.375V-3.46V.




**Gambar III-11 Orientasi sensor *gyroscope* MPU6050**

Sensor HMC5883 merupakan sensor *magnetometer* yang akan membantu sistem dalam menghasilkan nilai perilaku dari roket. HMC5883 merupakan sensor digital *magnetometer* 3 sumbu. Keluaran dari sensor *magnetometer* akan berupa gauss yang akan diformulasikan untuk menghasilkan arah kutub. Berikut ini spesifikasi dari sensor HMC5883.

- Memiliki skala pembacaan hingga ± 8 gauss.
- Memiliki resolusi sebesar 12 bit untuk meningkatkan keakuratan data untuk skala pembacaan tertinggi dapat menghasilkan keakuratan 19,53125 miligauss untuk tiap bit-nya.

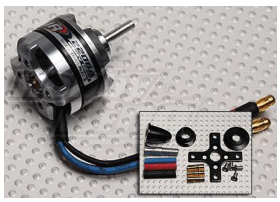


**Gambar III-12 Bentuk fisik sensor HMC5883**

c) Motor Brushless

Motor Brushless memiliki peran sebagai media penyedia daya dorong dari roket motor elektrik. Motor brushless menghasilkan daya dorong dari putaran baling-baling. Daya dorong yang dihasilkan harus lebih besar dari beban roket. Pada sistem ini, daya dorong yang diharapkan adalah sebesar minimal 2 kali dari beban roket. Motor pendorong yang digunakan bersifat motor brushless dengan spesifikasi sebagai berikut:

- Tegangan: 11.1V / 3 sel LiPO
- Tenaga keluaran: 130 Watt
- Arus maksimum: 17,8 Ampere
- RPM/V: 1650 kv
- Baling-Baling berdiameter 8 inch
- Kekuatan dorongan: 750 gram



**Gambar III-13 Turnigy L2206A-1650 Brushless Motor**

d) *Electronic Speed Control*(ESC)

Perangkat ESC digunakan untuk membantu peran mikrokontroler dalam mengatur kecepatan putaran dari motor *brushless*. Oleh karena itu, ESC juga bisa dikatakan sebagai *driver* untuk mengendalikan motor *brushless* tanpa membebani kerja dari mikrokontroler. Berdasarkan spesifikasi motor, motor *brushless* yang digunakan memerlukan arus hingga 17,8 Ampere maka dibutuhkan ESC yang memiliki kemampuan menyuplai yang lebih dari 17,8 Ampere. Maka dipilih Turnigy Multistar 30A Multi-rotor Brushless. ESC ini dipilih dengan beberapa pertimbangan yakni:

- Suplai arus maksimal 30 Ampere. Oleh karena itu, proses suplai arus pada motor tidak akan terganggu karena motor brushless sendiri hanya memerlukan tegangan hingga arus 17,8 Ampere.
- Penggunaan baterai LiPO 3 sel mengharuskan adanya kemampuan dari ESC untuk menerima tegangan baterai. ESC ini dapat menerima tegangan pada 2-4 sel sebagai ESC yang memenuhi standar keamanan dari tegangan ESC.
- Berdasarkan hasil perhitungan putaran motor tanpa baling-baling , driver motor harus dapat menghasilkan putaran maksimal sebesar 20.790 RPM. ESC ini memiliki kemampuan untuk menghasilkan putaran hingga 240.000 RPM yang telah memenuhi syarat maksimal putaran motor.
- Adanya sistem BEC yang menjadi sistem regulator pada sistem. Tegangan yang dihasilkan BEC sebesar 5,5 Volt dengan arus sebesar 4 Ampere.




**Gambar III-14 Turnigy Multistar 30A Multi-rotor ESC**

e) Motor Servo

Pada sistem ini motor servo memiliki peran untuk mengendalikan tingkat perubahan sudut defleksi ekor aktif yang terdapat pada roket. Ekor aktif tersebut yang akan dipakai untuk mengurangi efek *contra-rotating* yang terjadi pada roket. Penggunaan motor brushless pada sistem roket akan mempengaruhi seberapa besar torsi yang harus dimiliki oleh motor servo yang digunakan. Menurut *datasheet* Turnigy L2206A-1650 Brushless Motor, motor pendorong ini memiliki daya dorong sebesar 750 gram yang langsung diarahkan kearah aktuator ekor aktif (Sesuai dengan yang digambarkan pada perancangan mekanik). Oleh karena itu, motor servo yang digunakan harus memiliki torsi untuk menahan beban yang lebih besar dari 750 gr. Untuk itu dipilih motor servo HXT900 dengan beberapa pertimbangan sebagai berikut

- Torsi yang dimiliki torsi sebesar 1,6kg. Hal tersebut telah dapat memenuhi torsi yang diperlukan yakni lebih besar dari 750 gr.
- Tegangan kerja 3V~6V. Dengan adanya regulator yang telah tertanam pada *Electronic Speed Control (ESC)* sebesar 5V/4A maka tegangan kerja untuk servo telah terpenuhi.




**Gambar III-15 Motor Servo HXT500**

f) Radio Komunikasi

Radio komunikasi pada sistem ini berfungsi sebagai media perantara pengiriman data perilaku roket antara roket dan *ground station*. Radio komunikasi yang akan digunakan bersifat *full-duplex*. Pemilihan komunikasi *full-duplex* dikarenakan adanya proses permintaan dan pengiriman data yang akan dilakukan secara bersamaan. XBee-PRO merupakan pilihan untuk melakukan komunikasi pengiriman data jarak

jauh. XBee-PRO dibuat berdasarkan standar IEEE 802.15.4 bekerja frekuensi 2,4GHz. Frekuensi 2,4GHz dipilih karena tidak adanya keharusan dalam pendaftaran kepada badan pemerintahan terhadap perangkat dengan komunikasi tersebut. XBee-PRO memiliki jarak jangkauan hingga 1 *mile* (sekitar 1600 meter) dalam keadaan *outdoor Line-of-Sight*(LOS).




**Gambar III-16 Radio Modem XBee-PRO**

*g) Ground Station*

Untuk menerima data perilaku dari roket maka dibutuhkan suatu media penerima yang disebut *Ground Station*. Perangkat *Ground Station* yang akan digunakan berupa perangkat komputer yang berisikan perangkat lunak penerima data.

h) Radio Penerima

Radio penerima adalah perangkat yang digunakan untuk menerima sinyal kontrol yang berasal dari perangkat *remote* kontrol. Pada sistem ini, radio penerima memiliki peran untuk mengontrol laju kecepatan dan memberi perintah mengaktifkan motor pada roket. Radio penerima ini bersifat *simplex* dimana peran pengirim dan penerima data telah baku. Keluaran tiap kanal pada radio ini bersifat PWM sehingga dapat dibaca oleh mikrokontroler.




**Gambar III-17 Receiver *remote* turnigy9x**

i) *Remote* Kontrol

Perangkat *remote* kontrol digunakan untuk mengontrol kecepatan dan aktivasi motor *brushless*. *Remote* kontrol yang digunakan dalam perancangan roket ini menggunakan turnigy9x dengan 8 kanal kontrol. Pemilihan *remote* ini didasari karena jumlah kanal yang digunakan dalam sistem roket berjumlah 5 kanal yakni

- kanal 1 untuk *pitch*
- kanal 2 untuk *roll*
- kanal 3 untuk *throttle*
- kanal 4 untuk *yaw* dan
- kanal 5 untuk komando separasi.

Selain itu, *remote* ini dilengkapi dengan *Frequency-hopping spread spectrum* (FHSS) sehingga sistem komunikasi tidak terganggu meskipun ada pengguna lain yang menggunakan *remote* yang sama disekitarnya. Jarak jangkauan *remote* ini mencapai 2 Km *line-of-sight*(LOS).



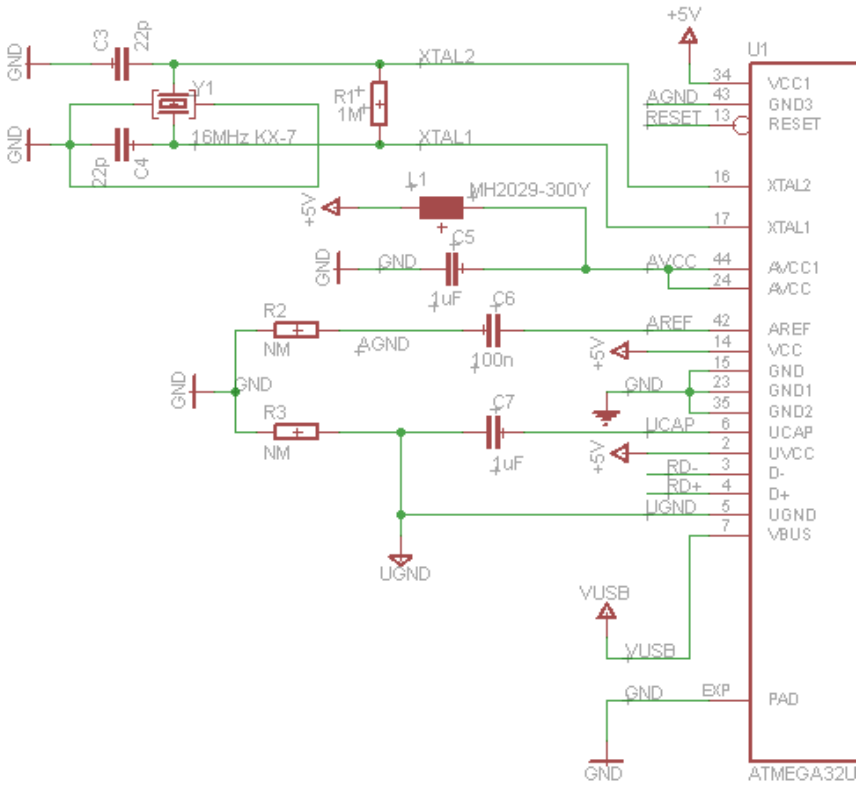

**Gambar III-18 *Remote* turnigy9x**

### 3.2.2 *Wiring Diagram* Sistem

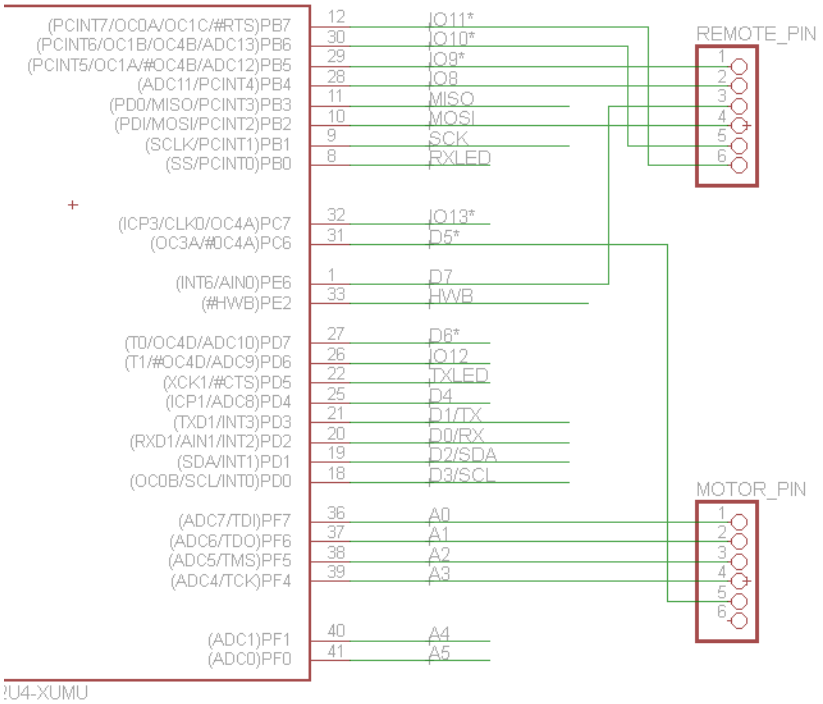
#### 3.2.2.1 Mikrokontroler dan komponen pendukung

Untuk melakukan proses perancangan algoritma pada mikrokontroler, kita perlu mengetahui bagaimana konfigurasi dan hubungan antara unit kontrol dan perangkat pendukungnnya. Proses pemetaan konfigurasi konektivitas seluruh komponen elektronika akan dilakukan dengan menggunakan perangkat lunak Eagle CadSoft 6.3.0. Hasil dari proses pemetaan tersebut dapat dicetak menjadi

sebuah perangkat PCB yang didalamnya berisikan komponen elektronika yang digunakan dalam pengontrolan roket.




**Gambar III-19 Sistem minimum dari mikrokontroler**




**Gambar III-20 Wiring diagram koneksi mikrokontroler, *remote*, dan pin motor**

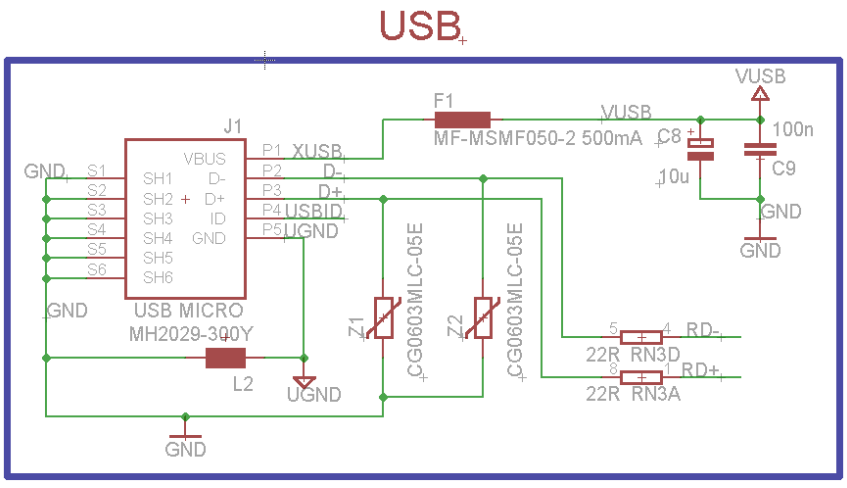
Pada perangkat lunak eagle, unit kontrol yang berupa mikrokontroler akan terhubung dengan beberapa perangkat menggunakan jalur penghubung baik menggunakan *Net*(Nama alias dari jalur).

Berikut ini merupakan konfigurasi pin data yang digunakan mikrokontroler yang terhubung kebeberapa perangkat lainnya.

**Tabel III-1 Konfigurasi pin yang digunakan pada mikrokontroler**

| Nama Pin | Nomor Pin | Fungsi | Keterangan |
|---|---|---|---|
| **Pin C** | 6 | Output | Pengendali Motor Brushless |
| **Pin F** | 7 | Output | Pengendali Motor Servo 1 |
| **Pin F** | 6 | Output | Pengendali Motor Servo 2 |
| **Pin F** | 5 | Output | Pengendali Motor Servo 3 |
| **Pin F** | 4 | Output | Pengendali Motor Servo 4 |
| **Pin D** | 2 | Input | Penerima data perilaku roket kepada radio komunikasi |
| **Pin D** | 3 | Output | Pengirim data perilaku roket kepada radio komunikasi |
| **Pin D** | 0 | Output | Pemberi sinyal *clock* untuk sinkronisasi perangkat sensor |
| **Pin D** | 1 | In/Out | Pengirim atau penerima data terhadap perangkat sensor |
| **Pin E** | 6 | Input | Penerima data masukan dari kanal *throttle* pada *remote* |
| **Pin B** | 4 | Input | Penerima data masukan dari kanal *roll* pada *remote* |
| **Pin B** | 5 | Input | Penerima data masukan dari kanal *pitch* pada *remote* |
| **Pin B** | 2 | Input | Penerima data masukan dari kanal *yaw* pada *remote* |

Selain beberapa fitur yang telah dijelaskan atmega32u4 memiliki fitur USB yang memudahkan proses penanaman algoritma serta pembacaan data keluaran mikrokontroler. Berikut ini merupakan koneksi rangkaian miniUSB yang terhubung dengan mikrokontroler atmega32u4.



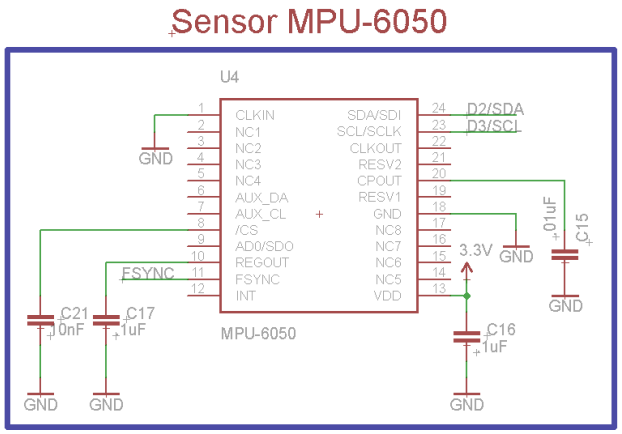

**Gambar III-21 Koneksi rangkaian miniUSB**

Dengan konfigurasi sebagai berikut.

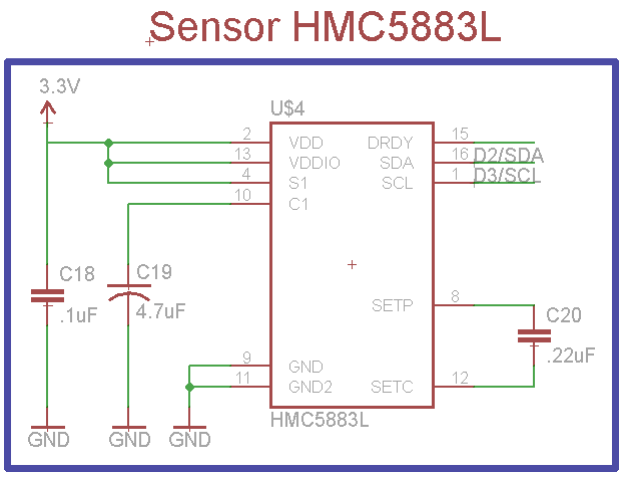
**Tabel III-2 Konfigurasi koneksi mini USB terhadap mikrokontroler**

| Nama Pin | Fungsi | Keterangan |
|---|---|---|
| **Pin D-** | Output/Input | Pin D merupakan pin yang berperan sebagai jalur komunikasi data. Logika 1 akan didapat ketika pin D- ada pada tegangan dibawah 0,3V dan pin D+ berada pada tegangan diatas 2,8V. Logika 0 akan didapat ketika pin D- ada pada tegangan diatas 2, 8V dan pin D+ ada pada tegangan dibawah 0,3V. |
| **Pin D+** | Output/Input | |
| **VBUS** | Output | Sumber tegangan dari perangkat komputer |

Selanjutnya adalah konektivitas antara unit pemroses dan komponen sensorik. Komponen sensorik yang digunakan akan melakukan proses transmisi data menggunakan protokol I2C yang telah dijelaskan pada subbab(2.3.1.5) tentang protokol I2C. Sensor akan terhubung dengan mikrokontroler melalui jalur SDA pada Pin D1 dan jalur SCL pada Pin D0.




**Gambar III-22 Koneksi sensor *accelerometer* dan *gyroscope* MPU-6050**

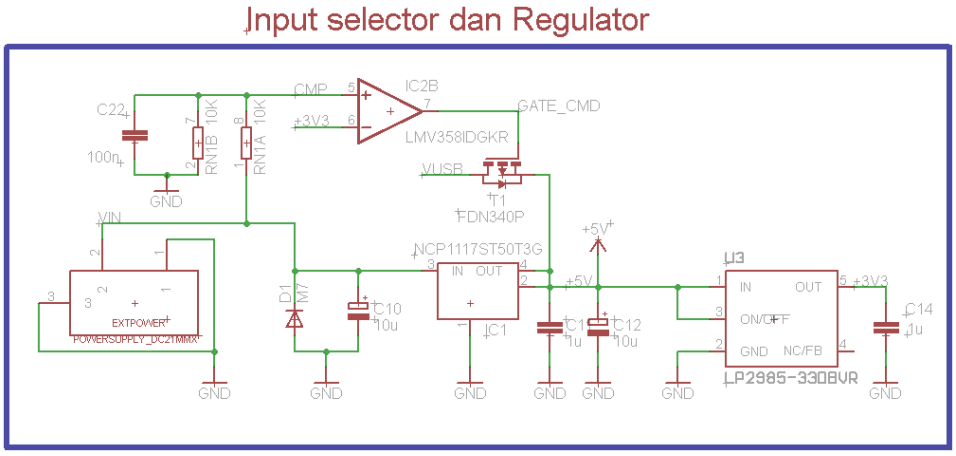



**Gambar III-23 Koneksi sensor *Magnetometer* HMC5883L**

### 3.2.2.2 Input selector dan Regulator

Atmega32u4 memiliki fitur USB port yang membuat proses pengujian data menjadi lebih mudah. Fungsi USB port pada atmega32u4 juga dapat dimanfaatkan sebagai sumber catu daya yang berasal dari perangkat komputer yang digunakan sebagai perangkat untuk menanamkan algoritma kontrol. Namun dalam penggunaan catu daya perlu pengaturan yang baik agar tidak terjadi bentrok dengan sumber daya lain seperti UBEC atau regulator yang dimiliki oleh *Electronic Speed Control*(ESC). Untuk itu digunakan suatu rangkaian komparator untuk membandingkan apakah ada masukan sumber tegangan dari arah yang berbeda. Disaat penggunaan tegangan dari USB *port* digunakan bersamaan dengan penggunaan sumber tenaga dari UBEC maka prioritas utama yang akan dihasilkan oleh rangkaian regulator adalah penggunaan tegangan dari UBEC.

Pada penggunaan sumber daya UBEC, sumber tegangan yang digunakan lebih besar dari 5 Volt. Untuk itu, perlu adanya suatu tambahan rangkaian regulator. NCP1117ST5 merupakan IC regulator tegangan yang mampu meregulasi tegangan masukannya menjadi bernilai 5 Volt. Selain tegangan 5V diperlukan juga tegangan 3,3 Volt untuk kebutuhan komponen sensor. LP2985 digunakan sebagai IC regulator untuk menghasilkan tegangan 3,3 Volt.

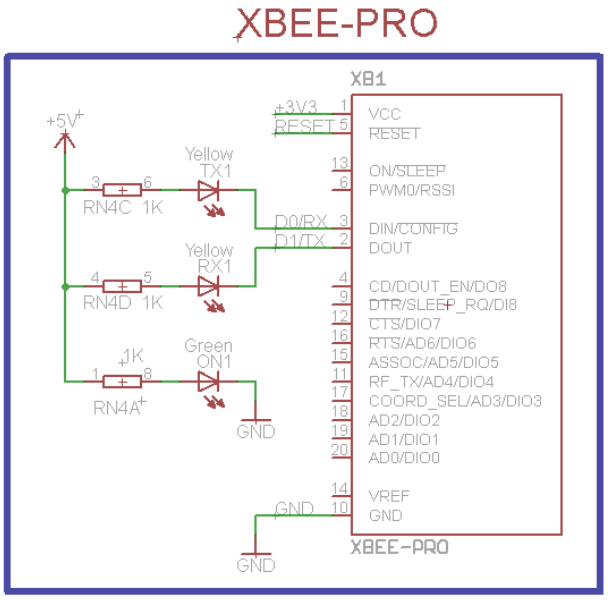



**Gambar III-24 Rangkaian *selector* masukan dan *regulator***

### 3.2.2.3 Komunikasi data

Komunikasi data yang akan digunakan dalam perancangan roket ini adalah *simplex* dan *full-duplex*. Protokol *simplex* akan digunakan untuk komunikasi kontrol *remote* terhadap roket. Protokol *simplex* dipilih karena *remote* hanya akan berperan untuk mengirimkan sinyal data kepada penerimanya. Sedangkan

protokol *full-duplex* digunakan untuk komunikasi data perilaku roket dari hasil pengolahan data sensor menggunakan perangkat Xbee-Pro. Protokol *full-duplex* dipilih untuk menghindari adanya *collision* pada pengirim permintaan *ground station* dan pengirim data perilaku roket. Berikut ini adalah gambar rangkaian komunikasi data yang terdapat pada roket.




**Gambar III-25 Koneksi rangkaian XBEE-Pro**

## 3.3 Perancangan Perangkat Lunak

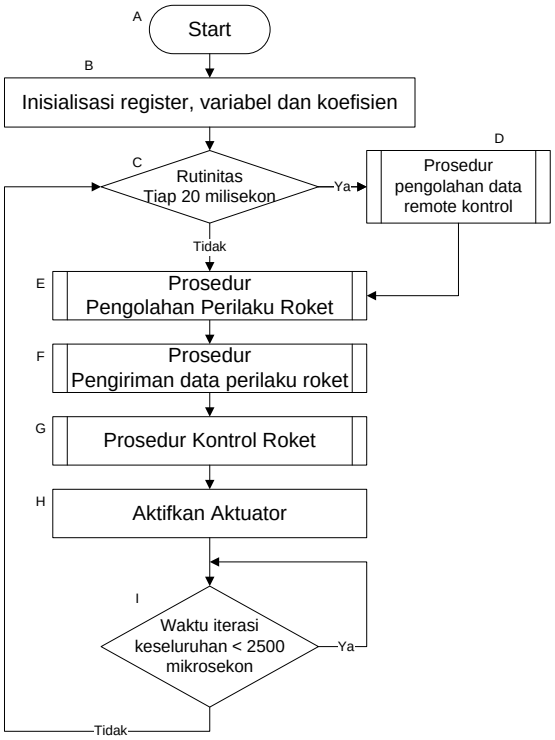
Pada perancangan perangkat lunak merupakan bagian utama dari pokok pembahasan topik ini. Dibagian perancangan ini akan dilakukan implementasi algoritma kontrol yang dibuat kedalam perangkat mikrokontroler.

### 3.3.1 Diagram Alir Program Utama

Pada diagram alir program utama ini berisikan tentang garis besar hal yang dikerjakan mikrokontroler sebagai pusat pengontrol. Pada diagram ini, tugas dibagi menjadi beberapa jenis tugas yaitu.

1. tugas yang dilakukan tiap iterasi yakni prosedur pembacaan data perubahan perilaku, pengiriman data dan kontrol roket.
2. tugas yang dilakukan tiap interval waktu yakni proses pengolahan data *remote* yang dilakukan setiap 20 ms.

Hal tersebut dilakukan untuk meminimalkan waktu tunda yang dihasilkan tiap iterasi untuk pengerjaan logika kontrol dengan frekuensi yang lebih tinggi.




**Gambar III-26 Diagram Alir Program Utama**

Berikut ini penjelasan diagram alir program utama.

A. Memulai Program.
B. Inisialisasi register, variabel dan koefisien.

C. Pemeriksaan apakah algoritma memasuki rutinitas yang dilakukan setiap 20 milidetik yakni pengecekan pengolahan data *remote*. Hal tersebut dilakukan agar proses pengolahan data *remote* tidak mengganggu proses aktuator pada sistem. Jika ya lanjutkan ke poin D dan jika tidak lanjutkan ke poin E.
D. Panggil prosedur "pembacaan dan kontrol *remote*".
E. Proses pembacaan data perilaku roket pada sumbu *Roll*, *Pitch* dan *Yaw*
F. Panggil prosedur "pengiriman data perilaku roket".
G. Panggil prosedur "Kontrol Roket".
H. Aktifkan aktuator.
I. Proses menunggu hingga selama waktu iterasi program keseluruhan belum mencapai 2500 mikrosekon. Hal ini dimaksudkan agar proses pengerjaan keseluruhan program akan tetap pada 2500 mikrosekon.

Berdasarkan hasil pemetaan pada subbab 3.2.2 mengenai "wiring diagram sistem". Pada tabel III-1 terlihat beberapa pin yang digunakan sebagai masukan atau keluaran suatu fitur. Proses inisialisasi register dilakukan untuk mengaktifkan fitur-fitur yang dimiliki oleh mikrokontroler. Terdapat beberapa inisialisasi register pada mikrokontroler yakni

**Inisialisasi I2C**

Pada proses inisialisasi I2C yang akan digunakan untuk berkomunikasi dengan perangkat sensor terdapat beberapa register yang memiliki peran untuk pengaktifan fitur I2C yakni

- Pengaturan *prescaler* pada register TWSR.
- Pengaturan *clock rate* pada register TWBR.
- Pengaktifan I2C dengan cara memberikan logika *High* pada register TWCR bit TWEN.

**Inisialisasi port**

Pada proses inisialisasi *port* terdapat register yang harus diatur yakni

- Pengaturan port F sebagai *output* untuk dimanipulasi menjadi *Software PWM* melalui register DDRF (pin 4,5,6,7).
- Pengaktifan *pull-up* resistor dengan memberikan nilai High pada register PORTF.

**Inisialisasi PWM**

Pada proses inisialisasi PWM yang digunakan dalam pengaktifan aktuator terdapat beberapa register yang berperan dalam pengaktifan fitur PWM yakni

- Pengaturan arah keluaran PWM pada pin melalui register TCCR1A pada bit COM1A1, COM1B1 atau COM1C1.
- Pengaturan mode operasi PWM pada register TCCR1A (WGM10 dan WGM11) dan TCCR1B (WGM12 dan WGM13).
- Pengaturan frekuensi PWM pada register ICR1.
- Pengaturan *prescaler* pada register TCCR1A bit CS12, CS11 dan CS10.

**Inisialisasi aktif *Interrupt* dan *PCInterrupt***

Pada proses inisialisasi *interrupt* dan *PCInterrupt* yang digunakan untuk pembacaan data *remote* kontrol diperlukan pengaturan beberapa register yakni

- Mengatur *interrupt sense control* untuk mendeteksi *logical change* pada pin *interrupt* 6 dengan memberikan logika *High* register EICRB (*External Interrupt Control Register B*) pada bit ISC60.
- Mengaktifkan *interrupt* eksternal 6 dengan memberikan logika *High* register EIMSK (*External Interrupt Mask Register*) pada bit INT6.
- Mengaktifkan *PCinterrupt* dengan cara memberikan logika High pada register PCICR pada bit PCIE0

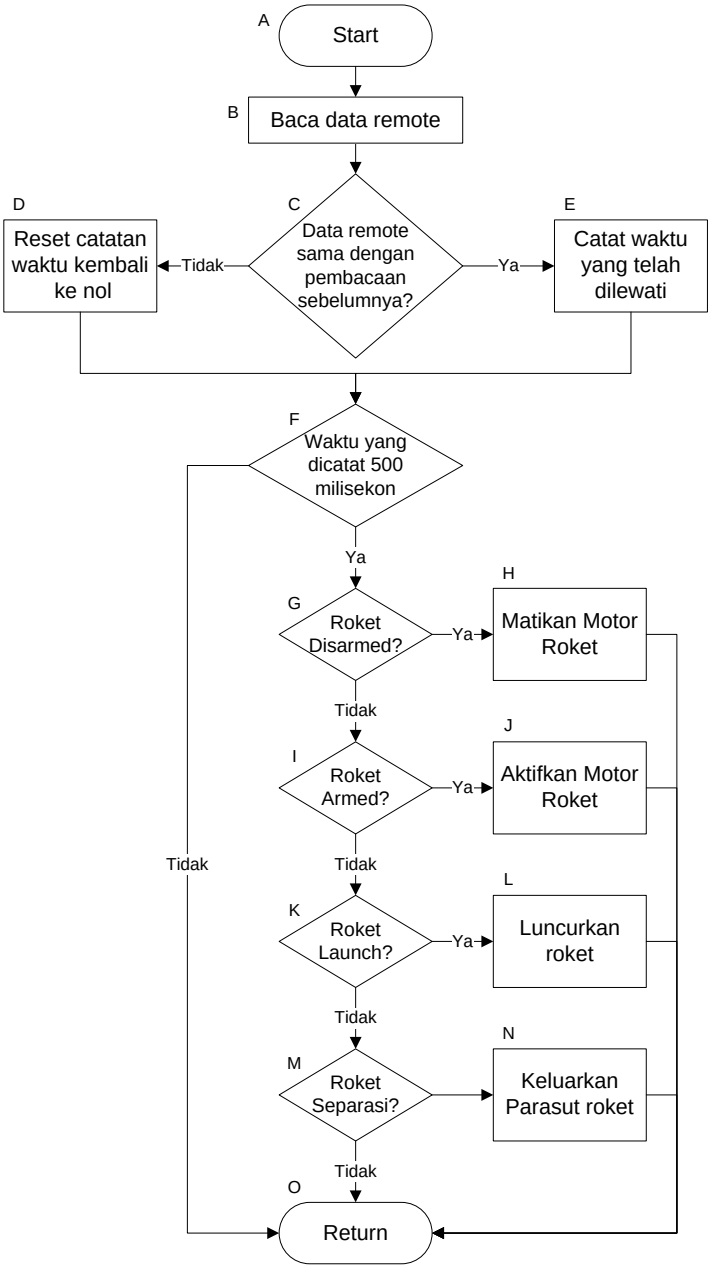
**Inisialisasi Komunikasi UART**

Pada proses inisialisasi komunikasi UART yang digunakan untuk mentransmisikan data terdapat beberapa register yang memiliki peran untuk pengaktifan fitur komunikasi UART yakni

- Mengkatifkan fungsi RX dan TX kanal 1 dengan mengatur register (RXEN1 dan TXEN1).
- Mengatur *Baudrate* yang digunakan dengan mengatur register UBRROH dan UBRROL.

### 3.3.2 Diagram Alir Prosedur Pengolahan Data *Remote*

Pada diagram alir prosedur pengolahan data *remote* berisikan tentang pengolahan terhadap data *remote* yang dibaca untuk dibandingkan dengan suatu kondisi. Hasil perbandingan tersebut akan menjadi kontrol terhadap kondisi roket.




**Gambar III-27 Diagram Alir Prosedur Pengolahan Data *Remote***

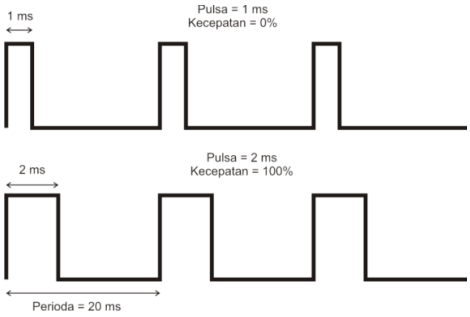
Berikut ini penjelasan dari Diagram Alir Prosedur Pengolahan Data *Remote*

A. Memulai Prosedur
B. Membaca data *remote* untuk keperluan proses pengecekan terhadap kombinasi data *remote* agar kondisi roket dapat diubah.

C. Periksa apakah hasil pembacaan data *remote* sama dengan pembacaan data *remote* pada iterasi sebelumnya. Jika ya lanjutkan ke poin E dan jika tidak lanjutkan ke poin D.
D. Ulangi proses pencatatan waktu. Program dilanjutkan ke poin F
E. Catat waktu yang telah dilewati pada saat kondisi *remote* sama.
F. Periksa apakah catatan waktu yang telah dilewati mencapai 500 milisekon. Hal ini dimaksudkan agar status keadaan roket tidak drastis berubah jika ada kesalahan pada penggunaan *remote* kontrol. Jika kondisi ya lanjutkan kepoin G dan jika kondisi tidak lanjutkan kepoin O.
G. Periksa apakah kombinasi data *remote* dalam keadaan *disarmed* (baling-baling tidak aktif). jika ya lanjutkan kepoin H dan jika tidak lanjutkan kepoin I.
H. Roket diubah kedalam kondisi *disarmed*. Program dilanjutkan ke poin O.
I. Periksa apakah kombinasi data *remote* dalam keadaan *armed* (baling-baling aktif). jika ya lanjutkan kepoin J dan jika tidak lanjutkan kepoin K.
J. Roket diubah kedalam kondisi *armed*. Program dilanjutkan ke poin O.
K. Periksa apakah kombinasi data *remote* dalam keadaan *launch* (aktifkan motor pendorong). jika ya lanjutkan kepoin L dan jika tidak lanjutkan kepoin M.
L. Motor roket diaktifkan dalam keadaan putaran penuh. Program dilanjutkan ke poin O.
M. Periksa apakah kombinasi data *remote* dalam keadaan separasi (proses penyelamatan badan roket dengan bantuan parasut). jika ya lanjutkan kepoin N dan jika tidak lanjutkan kepoin O.
N. Roket mengeluarkan parasutnya.
O. Kembali ke program utama.

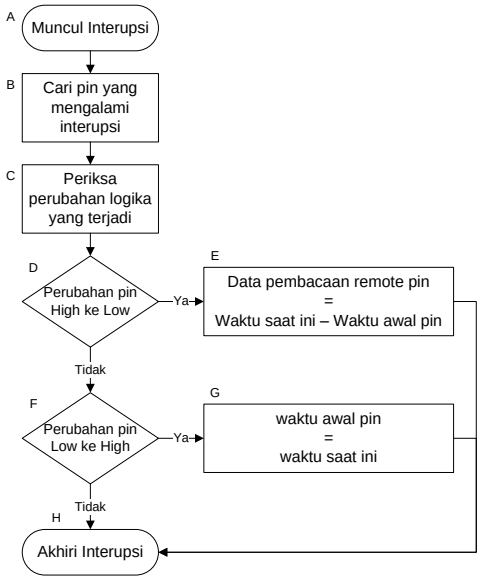
### 3.3.3 Diagram Alir Interupsi Penerimaan Data *Remote*

Data yang akan diterima oleh mikrokontroler berupa sinyal PWM dimana adanya perubahan lebar pulsa atau *duty cycle*. Pada konsep PWM untuk pembacaan *remote* kontrol, data yang dibaca akan berupa pulsa 1-2 milisekon. Pada subbab (2.3.1.1) mengenai interupsi dan subbab (2.3.1.2) mengenai *Timer/counter*, proses penerimaan data *remote* akan berupa proses interupsi terhadap adanya perubahan logika dalam suatu pin dimana lama waktu suatu pin mengalami logika HIGH akan dicatat oleh *Timer/counter*.




**Gambar III-28 Pulsa pembacaan data *remote***

Hasil pembacaan data *remote* berupa pulsa yang dapat diartikan sebagai lamanya waktu pada saat data berlogika HIGH. Logika HIGH akan terjadi pada 1-2 milidetik.




**Gambar III-29 Diagram Alir Interupsi Penerimaan Data *Remote***

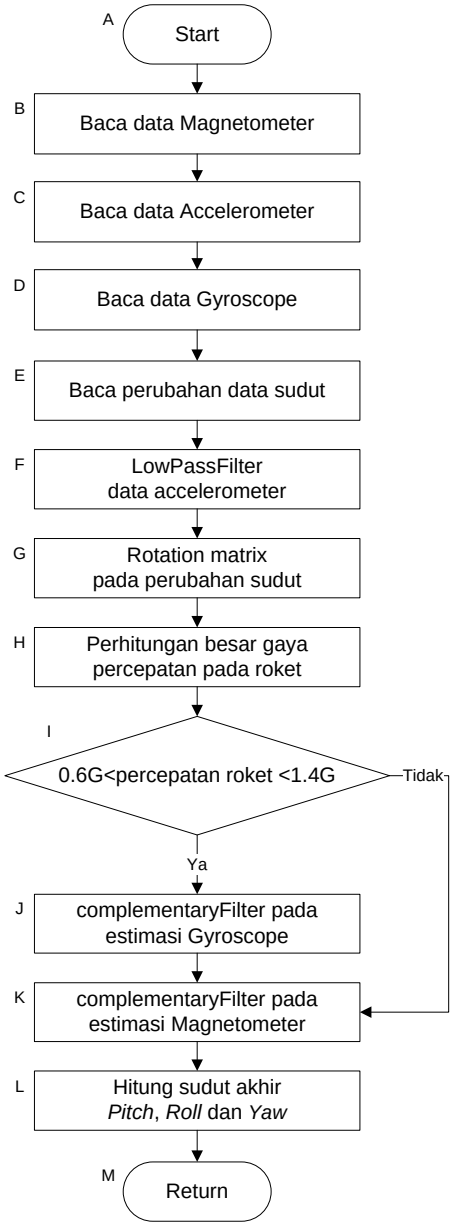
Berikut ini penjelasan dari diagram alir prosedur penerimaan data remote.

A. Muncul perubahan logika pada pin interupsi yang menyebabkan interupsi.

B. Mikrokontroler memeriksa kanal pin penerima yang mengalami interupsi.
C. Lakukan pengecekan terhadap pin mana yang berubah dan logika perubahannya.
D. Memeriksa apakah perubahan yang terjadi adalah perubahan logika *High* menjadi logika *Low* atau *falling edge*. Jika ya, lanjutkan ke poin E. Jika tidak, lanjutkan ke poin F.
E. Baca lama waktu pin tersebut dalam keadaan logika *High* dengan cara mengurangi waktu saat ini dengan waktu awal logika highdan simpan catatan waktu kedalam suatu variabel. Lanjutkan ke poin H.
F. Memeriksa apakah perubahan yang terjadi adalah perubahan logika *Low* menjadi logika *High* atau *rising edge*. Jika ya, lanjutkan ke poin G. Jika tidak, lanjutkan ke poin H.
G. Catat waktu awal dimulainya logika *High* suatu pin simpan catatan waktu kedalam suatu variabel.
H. Akhiri interupsi.

### 3.3.4 Diagram Alir Prosedur Pengolahan Perilaku Roket

Pada proses pembacaan data sensor digunakan beberapa proses *filter* agar hasil data sensor yang didapatkan mencapai hasil yang akurat. Selain itu, adanya proses *rotation matrix* yang digunakan untuk mendapatkan data orientasi roket.




**Gambar III-30 Diagram Alir Prosedur Pembacaan Data Sensor**

Berikut ini penjelasan dari diagram alir prosedur pembacaan data sensor.

A. Memulai prosedur
B. Lakukan proses pembacaan data pada *magnetometer*.
C. Lakukan proses pembacaan data pada *accelerometer*.
D. Lakukan proses pembacaan data pada *gyroscope*.

E. Hitung besar perubahan sudut pada seluruh sumbu yang terjadi dengan mengkalikan nilai *gyroscope* berupa kecepatan sudut dan perubahan waktu lalu simpan pada Δsudut.
F. Berikan *Low pass filter* pada data *accelerometer* untuk mengurangi efek guncangan yang dihasilkan motor pendorong.
G. Hitung estimasi sudut akhir dengan proses *rotation matrix* menggunakan perubahan sudut yang dihasilkan oleh sensor *gyroscope*.
H. Hitung berapa besar percepatan yang terjadi pada sistem. Hal ini dilakukan karena kelemahan data *accelerometer* tidak akurat digunakan dalam penghitungan rotasi jika ada pada percepatan yang tinggi.
I. Memeriksa apakah faktor *accelerometer* akan digunakan untuk proses complementary pada poin L dengan cara mengecek apakah percepatan yang terjadi pada sistem lebih besar dari 0,6G atau lebih kecil dari 1,4G. Jika ya, lanjutkan ke poin J. Jika tidak, lanjutkan ke poin K.
J. Gunakan faktor *accelerometer* pada *complementary filter* untuk membantu menghitung estimasi sudut akhir dengan sensor *gyroscope*.
K. Gunakan faktor *magnetometer* pada *complementary filter* untuk membantu menghitung estimasi sudut akhir dengan sensor *gyroscope*.
L. Hitung nilai akhir dari sudut dengan persamaan

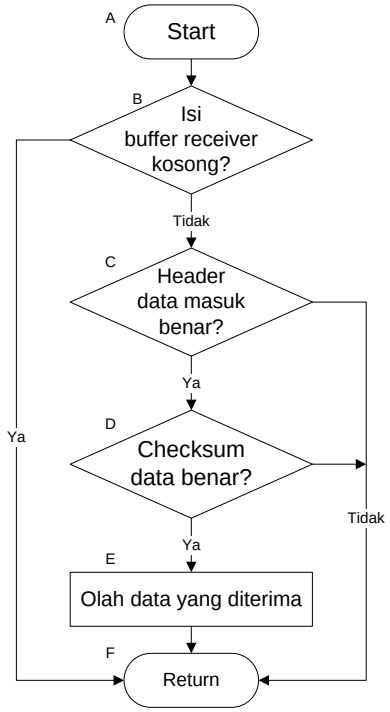
$$\text{Roll} = \arctan\left(\frac{\text{EstGyro [X]}}{\text{EstGyro [Z]}}\right) \text{ dan } \text{Pitch} = \arctan\left(\frac{\text{EstGyro [Y]}}{\text{EstGyro [Z]}}\right).$$

$$\text{Yaw} = \arctan\left(\frac{\text{EstGyro [X]} * \text{EstMag [Z]} - \text{EstGyro [Z]} * \text{EstMag [X]}}{\text{EstGyro [Y]} * \text{EstMag [Z]} - \text{EstGyro [Z]} * \text{EstMag [Y]}}\right)$$

M. Kembali ke program utama

### 3.3.5 Diagram Alir Prosedur Pengiriman Data

Diagram alir prosedur ini menjelaskan tentang bagaimana proses komunikasi data yang dilakukan oleh pengguna dan roket. Proses komunikasi data dilakukan dengan mengirimkan baris perintah. Baris perintah tersebut akan dicek apakah memiliki *header* yang sama. Selain itu, adanya *checksum* yang membantu komunikasi data untuk mengetahui kelengkapan data yang dikomunikasikan.




**Gambar III-31 Diagram Alir Prosedur Pengiriman Data**

Berikut ini penjelasan dari diagram alir prosedur pengiriman data

A. Memulai Prosedur.
B. Memeriksa apakah ada data masuk kedalam sistem dengan cara memeriksa apakah *Buffer* penerima kosong. Jika ya, lanjutkan kepoin E. Jika tidak, lanjutkan ke poin C.
C. Memeriksa apakah *header* dari data yang masuk memiliki nilai yang sama dengan *header* komunikasi yang telah ditentukan. Hal tersebut dimaksudkan untuk mencegah adanya data asing yang masuk kedalam *buffer* penerima pada sistem. Proses dilakukan dengan cara meng-XOR-kan *header* yang diterima dan *header* yang disimpan dalam roket. Jika ya, lanjutkan kepoin D. Jika tidak, lanjutkan ke poin F.

D. Memeriksa apakah bagian *checksum* dari data yang masuk memiliki nilai yang sama dengan *checksum* yang dihitung oleh mikrokontroler. Hal tersebut dimakdsukan untuk mencegah adanya data yang hilang pada saat pengiriman data. *checksum* didapat dengan cara meng-XOR-kan data non-*header* yang dikirimkan oleh GCS(*Ground Control Station*). Jika ya, lanjutkan kepoin E. Jika tidak, lanjutkan ke poin F.

E. Olah data yang diterima.

F. Kembali ke program utama

Pada proses pengiriman dan penerimaan data diperlukan suatu format transmisi data. Hal tersebut dimaksudkan untuk mempermudah perangkat dalam mengidentifikasi data yang ditransmisikan. Berikut ini format data yang dibuat untuk proses pengiriman data dari *ground station* ke roket.

| Byte ke | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | .. | .. | 6+ukuran payload | 7+ukuran payload |
| Header | | | Ukuran Payload | Jenis Paket Data | Payload Byte awal | | | Payload Byte akhir | Checksum |

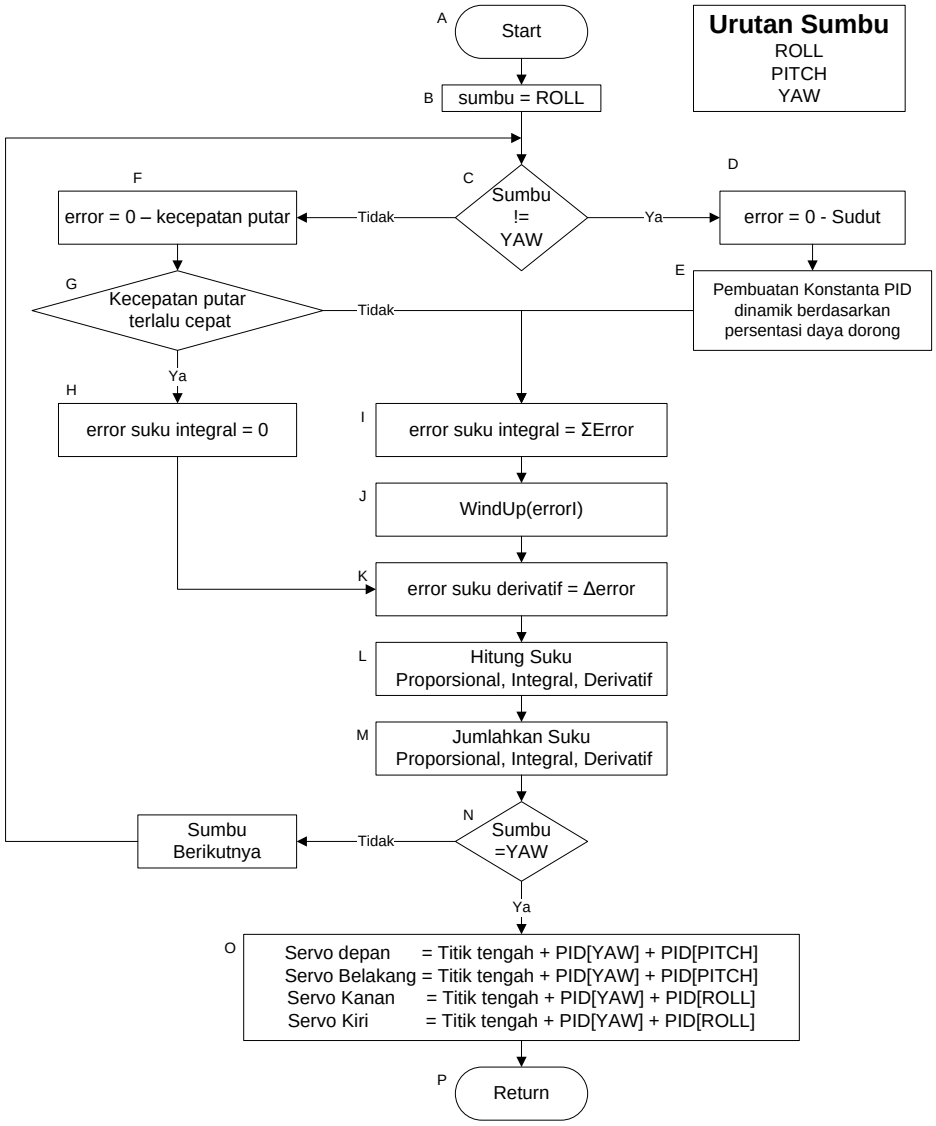
**Tabel III-3 Format komunikasi data**

Keterangan

- Byte ke 1,2,3 atau "Header" berisi *byte* yang menjadi identitas paket data untuk mengenali apakah data yang masuk merupakan data yang seharusnya.
- Byte ke 4 atau "Ukuran Payload" berisi *byte* yang memberitahukan berapa Byte data non-pengenal yang dikirim.
- Byte ke 5 atau "Jenis Paket Data" berisi *byte* yang dapat menjadi identitas jenis paket untuk proses pengolahan data yang diterima. Contoh jenis paket.
    - "A" : Komunikasi data perilaku roket
    - "B" : Komunikasi data kontrol roket
- Byte ke 6 sampai "6+ukuran payload" berisi *byte* yang digunakan dalam pengolahan sistem bisa berupa data sensor ataupun data yang lainnya.
- Byte ke "7+ukuran payload" berisi *byte* checksum yang digunakan untuk mengecek apakah data diterima secara lengkap. *checksum* ini merupakan hasil dari proses XOR dari data yang diterima kecuali *header*

### 3.3.6 Diagram Alir Prosedur Kontrol Roket

Pada diagram alir ini terdapat 3 sumbu yang memiliki masing-masing kontrol. Sumbu yang diolah akan berurutan mulai dari *Roll*, *Pitch* dan *Yaw*. Pada sumbu *Yaw*, nilai *error* didapatkan dari sensor *gyroscope*(kecepatan sudut) karena sistem harus menghentikan putaran efek *contra-rotating* motor pendorong sedangkan pada sumbu lainnya, nilai *error* didapatkan dari perubahan sudut karena sudut yang diinginkan untuk terbang vertikal merupakan 0 derajat. Hasil prosedur ini merupakan nilai PID yang dimasukan kedalam kontrol servo.




**Gambar III-32 Diagram Alir Prosedur Kontrol Roket**

Berikut ini penjelasan diagram alir prosedur kontrol roket.

A. Memulai Prosedur

B. sumbu *Roll* adalah sumbu pertama yang diolah pada iterasi kontrol PID.

C. Memeriksa apakah sumbu merupakan sumbu *Yaw*. Hal ini untuk membedakan penggunaan nilai *error*. Jika ya, lanjutkan ke poin D. Jika tidak, lanjutkan ke poin F.
D. Gunakan nilai (0 - sudut) untuk *error* pada sumbu *Roll* atau *Pitch*.
E. Memanipulasi koefisien PID dengan mengkalikan persentasi daya dorong dan koefisien PID yang digunakan untuk memperhalus respon dari roket. lanjutkan ke poin I.
F. Gunakan nilai (0 - kecepatan putar) sebagai *error* pada sumbu *Yaw*.
G. Periksa apakah kecepatan putaran badan roket melebihi batas putaran yang ditentukan. Jika ya lanjutkan kepoin H dan jika tidak lanjutkan kepoin I.
H. Jadikan *error* suku integral menjadi bernilai 0. Dikarenakan kontroler *integral* memiliki sifat meningkatkan *overshoot* maka penggunaannya harus dikurangi karena dapat menyebabkan sistem tidak stabil. Lanjutkan ke poin K.
I. Proses menghitung *error* suku integral dengan menjumlahkan nilai *error* untuk *error* suku integral terhadap iterasi sebelumnya.
J. Lakukan proses WindUp atau pembatasan nilai maksimum terhadap *error* suku integral.
K. Proses menghitung *error* suku derivatif dengan menggunakan selisih *error* pada iterasi terhadap iterasi sebelumnya.
L. Kalikan tiap *error* dengan konstanta yang ditentukan untuk mendapatkan nilai suku pada kontroler proporsional, integral dan derivatif.
M. Jumlahkan semua suku kontroler.
N. Memeriksa apakah sumbu merupakan sumbu *Yaw*. Jika ya, lanjutkan ke poin P. Jika tidak, kembali kepoin C dengan urutan sumbu berikutnya.
O. Isikan nilai hasil penghitungan PID kedalam tiap servo yang digunakan.
P. Kembali ke program utama.

Seluruh Algoritma yang telah dibuat akan di implementasikan kedalam mikrokontroler ATMEGA32u4 dengan menggunakan bahasa C pada kompiler Arduino.