BAB II
TEORI PENUNJANG

Bab ini akan membahas tentang teori dasar dari perancangan kontrol untuk mengatasi efek *contra-rotating* yang terjadi pada roket motor elektrik. Untuk itu, pembahasan dasar teori akan didasarkan terhadap beberapa pendekatan yakni:

1. Pendekatan secara teori kontrol.
2. Pendekatan secara perangkat lunak.
3. Pendekatan secara perangkat keras.

2.1 Pendekatan secara teori kontrol

2.1.1 terminologi

Controlled variable* dan *Manipulated variable. *Controlled variable* adalah suatu kuantitas atau kondisi yang terukur dan terkontrol. Pada topik ini, kecepatan putaran merupakan *Controlled variable*. *Manipulated variable* adalah kuantitas atau kondisi yang diubah oleh pengontrol yang akan mempengaruhi nilai *controlled variable*. Pada topik ini, mekanisme dari teknik pengontrolan roket yang akan menjadi *Manipulated variable*. Kontrol sendiri dapat diartikan pengukuran nilai *controlled variable* sistem dan pemberian sinyal pada sistem untuk mengoreksi nilai deviasi yang terukur menjadi nilai yang diharapkan.

Plants. *Plant* dapat berupa suatu peralatan atau suatu kelengkapan dari perangkat mesin yang bekerja bersama untuk mengerjakan suatu tujuan tertentu.

Processes. *Process* sebagai suatu kelanjutan operasi atau pengembangan yang ditandai dengan serangkaian perubahan bertahap yang menggantikan satu sama lain dengan cara yang relatif tetap dan mengarahkan pada suatu tujuan.

Systems. *System* merupakan kombinasi dari komponen yang bertindak bersama dan melakukan suatu tujuan.

Disturbances. *Disturbance* adalah suatu sinyal yang cenderung mempengaruhi nilai keluaran dari suatu sistem. *Disturbance* yang dihasilkan oleh sistem dinamakan *internal disturbance* sedangkan *disturbance* yang muncul dari luar sistem disebut *external distubance*.

Feedback Control. *Feedback control* mengartikan pada suatu operasi yang memiliki gangguan dan cenderung mengurangi perbedaan diantara keluaran dan referensi sistem.

2.1.2 *Open-loop* dan *Closed-loop*

Open-loop Control Systems. *Open-loop Control Systems* direpresentasikan sebagai sistem dimana nilai keluarannya tidak memiliki efek terhadap aksi dari kontrol. Hal tersebut menyebabkan akurasi nilai keluaran sistem akan bergantung pada kalibrasi. Kehadiran *distubance* pada sistem memberikan hasil yang tidak diharapkan. *Open-loop* biasa digunakan pada sistem *time-based* seperti mesin cuci dimana tidak adanya pengecekan tingkat kebersihan pakaian yang dicuci.

Closed-loop Control Systems. *Closed-loop Control Systems* merupakan representasi terhadap sistem yang nilai keluarannya memiliki efek timbal balik terhadap aksi dari kontrol. Pada sistem *Closed-loop* memiliki sinyal *error* dari hasil pembandingan perbedaan nilai keluaran dan nilai referensi.

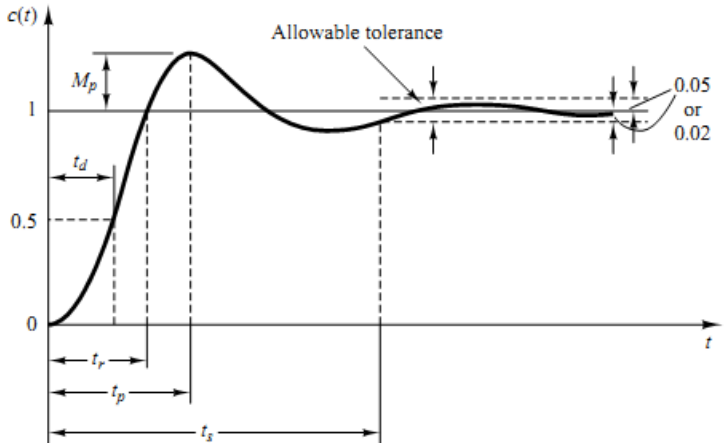
Perbandingan *Closed-loop* dan *Open-loop Control System*. Keuntungan dari penggunaaan *Closed-loop* adalah penggunaan sistem timbal balik yang membuat sistem dapat merespon terhadap gangguan baik dari dalam maupun dari luar sistem yang mustahil dilakukan pada *Open-loop*. Pada sisi stabilitas, *Open-loop* dianggap lebih mudah dirancang karena stabilitas sistem bukan merupakan masalah utama. Sedangkan pada *Closed-loop*, stabilitas sistem kadang kala menjadi masalah karena kelebihan pengkoreksian *error* yang dapat menyebabkan osilasi dari perubahan amplitudo. Oleh karena itu, penggunaan sistem kontrol bergantung pada keberadaan *disturbance*.

2.1.3 Respon sistem

Dalam suatu sistem yang bergerak akan terdapat suatu kontrol yang mengatur pergerakan sistem tersebut. Pada proses kontrol terdapat beberapa hal yang menjadi titik penentu respon dari sistem tersebut, yakni

- *Steady-state error* merupakan perbedaan antara nilai yang dihasilkan terhadap nilai referensi pada saat waktu menuju tak hingga. Sistem yang diharapkan akan memiliki *steady-state error* sekecil mungkin.
- *Rise time*, t_r ; waktu yang diperlukan sistem untuk melakukan kenaikan dari suatu nilai ke nilai lainnya. Untuk sistem yang memiliki redaman cukup(*underdamped*) nilai yang digunakan adalah 0% ke 100%. Sedangkan untuk sistem yang memiliki redaman berlebihan*(overdamped*) digunakan nilai 10% ke 90%. Respon sistem yang dianggap baik memiliki *rise time* yang kecil.

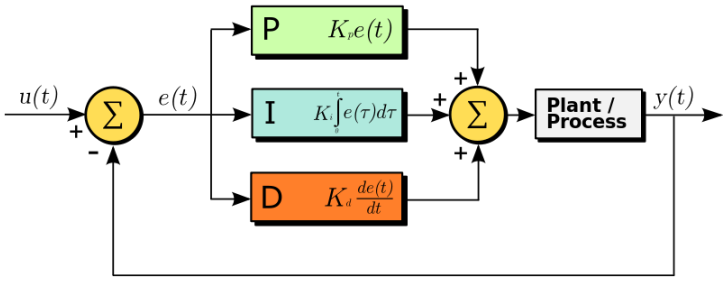
- *Maximum overshoot, M_p* ; *Maximum overshoot* merupakan nilai simpangan maksimum dari nilai puncak respon. Sistem yang stabil tidak memiliki *overshoot* yang besar.
- *Settling time, t_s* ; Settling time merupakan waktu yang diperlukan untuk mencapai dan diam pada nilai akhirnya yang memiliki persentase *steady-state error* sekitar 0,02 atau 0,05. Respon sistem yang dianggap baik memiliki *Settling time* yang kecil.



Gambar II-1 kurva kontrol

2.1.4 Kontrol Proposional Integral Derivatif (PID)

Setelah mengetahui faktor penentu baik buruknya respon sistem maka proses dilanjutkan kearah pengenalan kontrol. Kontrol proporsional integral derivatif (PID) merupakan salah satu jenis kontrol yang telah umum digunakan dalam dunia industri.



Gambar II-2 Diagram blok kontrol PID

Pada kontrol derivatif. Ketiga kontrol tersebut memiliki karakteristik yang terkait satu sama lainnya. kontrol PID terdapat tiga buah komponen utama yaitu kontrol proporsional, kontrol integral dan

Kontrol Proporsional

Pada suatu sistem yang menggunakan kontrol proporsional, Hubungan antara sinyal keluaran u(t) dan sinyal *error* e(t) adalah

$$u(t) = K_p e(t)$$

dalam bentuk diskrit, kontrol proporsional dapat ditulis menjadi

$$\frac{U(k)}{E(k)} = K_p e(k)$$

dimana K_p adalah *gain* proporsional

Persamaan tersebut memperlihatkan bahwa kontrol proporsional merupakan pembesaran/amplifikasi dari nilai sinyal *error*. Kontrol proposional menghasilkan beberapa efek dalam sistem kontrol yakni sebagai berikut.

1. Semakin besar K_p maka semakin kecil rise time yang dihasilkan.
2. Penggunaan K_p yang berlebihan akan menyebabkan sistem tidak stabil atau berosilasi terhadap titik referensinya.
3. Berapapun besarnya nilai K_p, kontrol proporsional tidak dapat menghilangkan *steady-state error*.

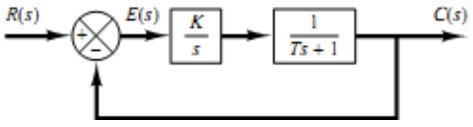
Kontrol Integral

Suatu sistem yang tidak memiliki kontrol integral akan terdapat *steady-state error* atau nilai *offset*. S*teady-state error* adalah selisih dari nilai referensi dan nilai keluaran pada *limit* waktu tak hingga.



Gambar II-3 Unit-step *response* dan *offset*

Nilai offset menyebabkan sistem tidak dapat diam pada titik referensinya. Kontrol integral digunakan untuk mengeliminasi nilai *offset* pada sistem.



Gambar II-4 Kontrol sistem integral

Gambar II-4 menggambarkan suatu sistem yang menggunakan kontrol integral. Kontrol sistem tersebut mengunakan kontroler integral yang dapat

dibentuk kedalam bentuk *closed-loop transfer function* menjadi persamaan berikut ini.

$$\frac{C(s)}{R(s)} = \frac{K}{s(Ts+1)+K}$$

dengan

$$\frac{E(s)}{R(s)} = \frac{R(s)-C(s)}{R(s)} = \frac{s(Ts+1)}{s(Ts+1)+K}$$

Steady-state error untuk *unit-step response* dapat didapatkan dengan memasukan pada persamaan:

$$e_{ss} = \lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s)$$

$$e_{ss} = \lim_{s\to 0} sE(s)$$

$$= \lim_{s\to 0} \frac{s^2(Ts+1)}{Ts^2+s+K}\frac{1}{s}$$

$$= 0$$

Suatu sistem yang menggunakan kontrol integral, nilai keluaran u(t) akan berubah dengan rasio integral terhadap pergerakan sinyal *error* e(t). Hubungan antara sinyal keluaran u(t) dan sinyal *error* e(t) adalah

$$\frac{d}{dt}u(t) = K_i e(t) \;\; atau \;\; u(t) = K_i \int_0^t e(t)dt$$

dalam bentuk diskrit, persamaan ini dapat ditulis menjadi

$$\frac{U(k)}{E(k)} = K_i \sum_{i=1}^{k} e(i)\Delta t$$

Selain menghilangkan *steady-state error*, pada penggunaannya kontrol integral memberikan beberapa efek lain yakni sebagai berikut.

1. Keluaran kontrol integral membutuhkan selang waktu sehingga dapat memperlambat respon sistem.
2. Penggunaan kontrol integral dapat mempercepat hilangnya *offset*. Namun jika penggunaannya terlalu besar, tingkat osilasi dari keluaran kontrol pun akan bertambah besar.

Kontrol Derivatif

Pada saat kontrol derivatif ditambahkan pada sistem yang menggunakan kontrol proporsional akan menghasilkan kontroler yang memiliki sensitifitas tinggi. Selain itu, kontrol derivatif menggunakan laju perubahan dari sinyal *error* sehingga kontroler yang dihasilkan dapat menghasilkan nilai koreksi

yang signifikan sebelum sinyal kontroler menjadi terlalu besar. Pada suatu sistem yang menggunakan kontroler derivatif, Hubungan antara nilai keluaran u(t) dan sinyal *error* e(t) adalah

$$u(t) = K_d \frac{d}{dt} e(t)$$

dalam bentuk diskrit, persamaan ini dapat dituliskan menjadi

$$\frac{U(k)}{E(k)} = Kd \frac{e(k) - e(k-1)}{\Delta t}$$

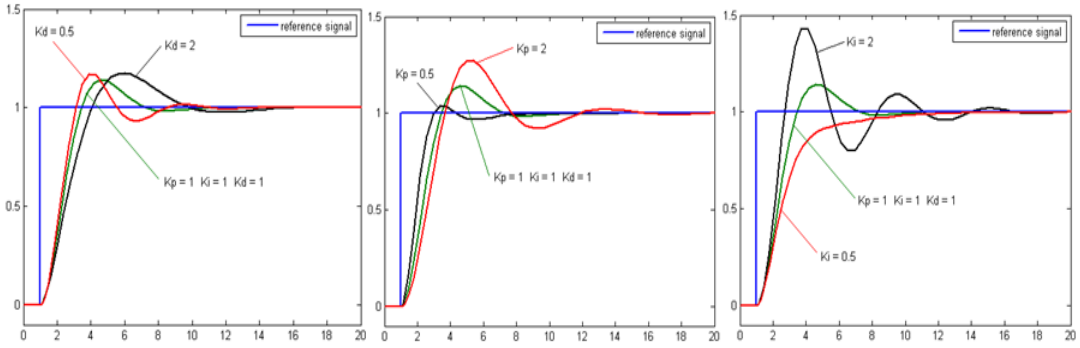
Sinyal kontrol ini memiliki karakter untuk mengurangi *steady-state error* meskipun tidak sebesar yang dilakukan oleh kontrol integral.

Dengan demikian dapat kita simpulkan hasil pembahasan kontrol terhadap sistem yang ada dalam tabel II-1 berikut.

Tabel II-1 Efek kontrol (Proporsional, Integral, Derivatif)

Kontrol	*Rise time*	*Overshoot*	*Settling time*	*Steady-state error*
Proporsional	Menurunkan	Meningkatkan	Perubahan kecil	Mengurangi
Integral	Menurunkan	Meningkatkan	Meningkatkan	Mengeliminasi
Derivatif	Perubahan kecil	Menurunkan	Menurunkan	Perubahan Kecil

Jika tabel II-1 efek kontrol digambarkan dalam grafik menjadi sebagai berikut.



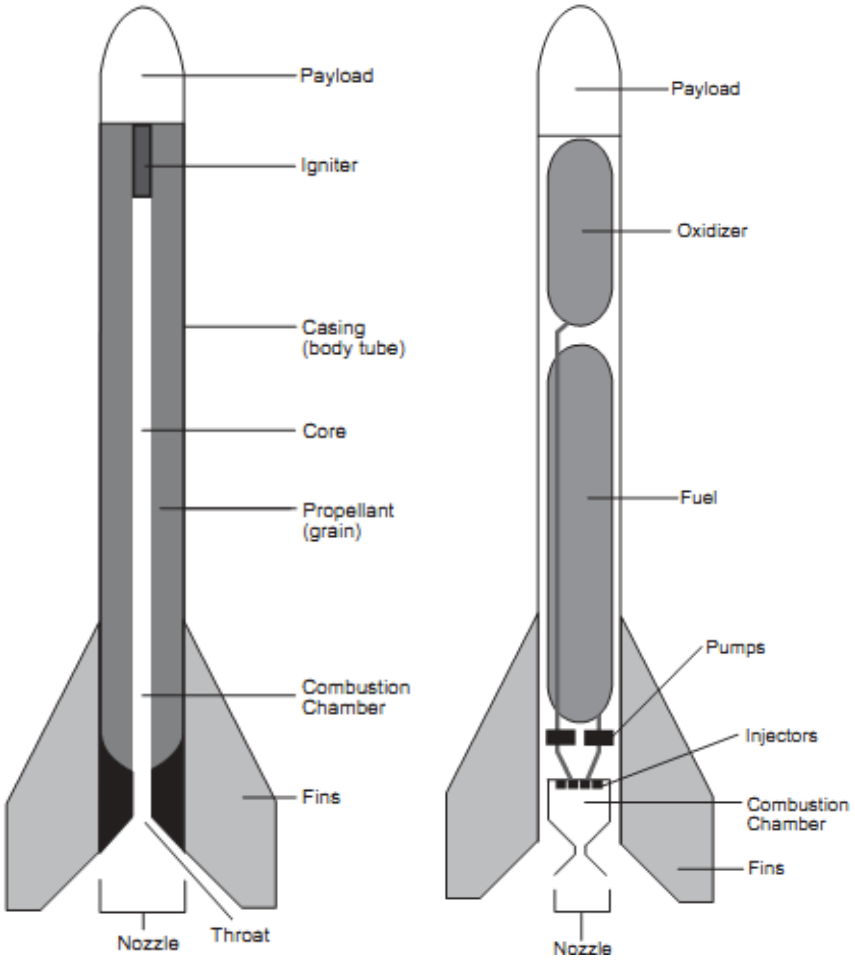
Gambar II-5 Efek kontrol (Proporsional, Integral, Derivatif)

Berdasarkan beberapa karakteristik yang telah dijelaskan sebelumnya, setiap kontrol memiliki kelebihan dan kekurangannya masing-masing. Sistem kontrol yang baik akan tercipta dengan melakukan penggabungan terhadap kontrol proporsional, integral dan derivatif.

2.1.5 Roket

Roket merupakan suatu wahana yang memperoleh daya dorong dari mesin roket. Mesin roket terbentuk sepenuhnya dari propelan yang dibawa dalam roket. Mesin roket bekerja dengan mengandalkan proses aksi dan reaksi. Proses riset yang dilakukan oleh berbagai negara didunia telah menghasil rancang bentuk roket yang telah dapat mencapai luar angkasa. Berikut ini beberapa bagian yang umumnya ada pada roket.

1. *Payload* merupakan tempat penyimpanan beban berbayar yang dibawa oleh roket. Pada roket militer, *payload* biasanya berisikan hulu ledak. Sedangkan pada roket jelajah, *payload* biasanya berisikan manusia ataupun perangkat pemantau.
2. *Igniter* merupakan bagian pemicu aktifnya roket atau pengontrol kecepatan roket. Pada umunnya *igniter* akan berupa suatu pematik atau penghasil api yang digunakan untuk pembakaran *propellant*.
3. *Body tube* merupakan bagian selubung yang melapisi inti roket.
4. *Propellant* merupakan bahan bakar yang digunakan oleh roket untuk mendapatkan gaya angkat. *Propellant* biasanya berupa bahan bakar dalam bentuk padat maupun cair yang dicampur dengan suatu zat pengoksidasi yang menghasilkan udara panas dengan tekanan.
5. *Combustion Chamber* merupakan tempat dimana terjadinya proses kimia pembakaran propellant.
6. *Nozzle* merupakan bagian terbuka pada bawah roket yang membiarkan tekanan udara hasil pembakaran keluar. Tujuan dari *nozzle* adalah untuk meningkatkan akselerasi dari tekanan udara yang keluar dari roket. Perbedaan bentuk dan ukuran nozzle akan menghasilkan perbedaan daya dorong roket.
7. *Fin* merupakan bagian ekor dari roket yang berfungsi sebagai pembentuk titik stabilisasi dari roket. Perbedaan dari bentuk fin akan menghasilkan perubahan dari pusat tekanan(*center of pressure*) pada roket. Jika pusat tekanan semakin mendekati pusat massa dari roket maka kestabilan roket akan semakin berkurang(catatan: pusat tekanan lebih bawah dari pusat massa roket).



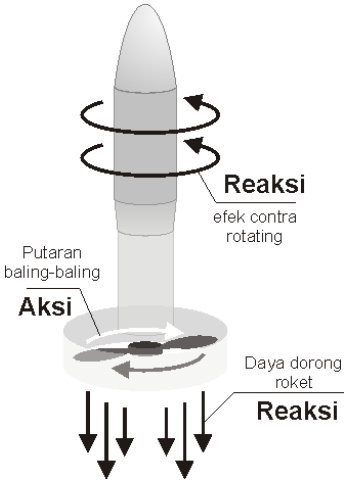
Gambar II-6 (a) roket propelan padat (b) roket propelan cair

Begitupula pada motor dengan roket elektrik, keseluruhan bagian roket menyerupai dengan roket berbahan bakar seperti *payload*, *body tube*, *fin* dan *nozzle*. Namun pada roket motor elektrik terdapat beberapa hal yang berbeda seperti tidak adanya *combustion chamber* karena penggunaan bahan bakar yang digantikan oleh daya dorong motor elektrik yang dihasilkan dari putaran motor dengan baling-baling.

2.1.6 Efek *contra-rotating*

Mesin roket bekerja dengan mengandalkan proses aksi dan reaksi. Sama halnya seperti pada roket motor elektrik yang menggunakan motor elektrik berbaling-baling untuk mendapatkan daya dorong motor. Namun dikarenakan roket motor elektrik menggunakan motor berbaling-baling untuk mendapatkan daya dorong maka terjadi pula aksi reaksi putaran motor terhadap poros tegak dari

badan roket yang menyebabkan adanya pergerakan berlawanan poros badan roket terhadap putaran motor roket yang disebut efek *contra-rotating*.



Gambar II-7 Efek *contra-rotating* pada roket

Efek *contra-rotating* yang terjadi pada roket dapat menyebabkan beberapa hal seperti penurunan daya dorong motor karena pernurunan kecepatan putaran motor dan kesalahan pembacaan data perilaku roket karena adanya pergerakan pada poros badan roket yang terus menerus.

2.1.7 Stabilitas dan kontrol roket

Suatu roket diterbangkan harus dalam keadaan stabil. Jika roket yang tidak stabil diterbangkan akan sulit memprediksi arah geraknya. Stabilitas dari roket sendiri bergantung pada pusat massa*(center of gravity)* dan pusat tekanan*(center of pressure).* Pusat massa roket adalah posisi dimana seluruh massa pada roket seimbang pada suatu titik. Jika roket diberikan suatu benang penyangga pada titik pusat massa maka roket tidak akan jatuh ke salah satu sisi badan roket. Pusat tekanan merupakan rata-rata dari lokasi terjadinya tekanan. Pusat tekanan pada roket muncul ketika adanya aliran udara pada roket. Dengan adanya pusat tekanan yang berada dibelakang pusat massa maka roket dianggap stabil. *Static margin* merupakan suatu nilai yang digunakan dalam suatu perancangan stabilisasi pada suatu wahana terbang. *Static margin* terdapat dua kondisi utama yakni:

- *Static margin* negatif dimana pusat massa berada dibelakang pusat tekanan. Sistem yang memiliki *static margin* bernilai negatif akan merespon gangguan dengan menghasilkan momen aerodinamis yang akan memperbesar gangguan yang masuk. Jika suatu wahana memiliki *static*

margin negatif maka disaat adanya gangguan yang masuk wahana terbang akan semakin menjauh titik aliran tekanan yang datang.

- *Static margin* positif dimana pusat massa berada didepan pusat tekanan. Sistem yang memiliki *static margin* bernilai positif akan merespon gangguan dengan menghasilkan momen aerodinamis yang mengembalikan arah wahana terbang kearah titik aliran tekanan yang datang. Hal tersebut dapat diartikan sebagai bentuk redaman pada saat sistem terkena gangguan. Nilai *static margin* yang semakin besar akan menyebabkan peningkatan kestabilan pada wahana terbang namun akan semakin sulit dikontrol karena wahana lebih terpengaruh oleh aliran tekanan yang datang.

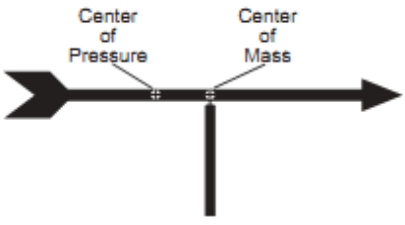
Tabel II-2 Stabilitas roket

Nilai *Static margin*	Stabilitas
0 < *static margin* < 1	Marginal Stabil
1 < *static margin* < 2,5	Stabil
2,5 < *static margin*	Sangat Stabil

Berikut ini merupakan persamaan untuk menentukan *static margin* dalam suatu roket.

$$\text{Static Margin} = \frac{Pusat\ massa - Pusat\ tekanan}{Diameter\ badan\ roket}$$

Penunjuk arah angin merupakan salah satu contoh konktrit pada konsep roket. Pada saat adanya aliran angin maka penunjuk arah angin akan mengarah kearah angin mengalir dengan kepala dibagian depan. Begitupula dengan roket yang stabil, arah pergerakan roket dapat diprediksi.

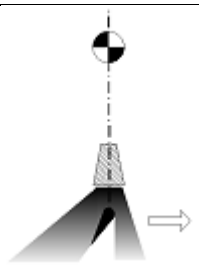
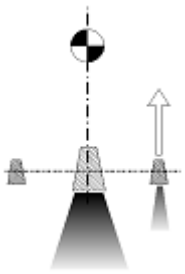


Gambar II-8 Penunjuk arah angin

Pada proses peluncuran satelit menggunakan roket, proses kontrol dari roket sangatlah penting. Hal tersebut dikarenakan jika daya dorong terlalu banyak atau

sedikit akan menyebabkan kesalahan orbit pada satelit. Pengaruh lainnya yang terhadap proses kontrol daya dorong roket adalah arah trayektori atau pergerakan dari roket. Berikut ini merupakan beberapa jenis kontrol stabilisasi telah banyak diterapkan pada sistem penerbangan.

Tabel II-3 Konsep konsep non-aerodinamik

	Gimballed engine	Jet vanes	Attitude thrusters	Reaction wheels
Diagram				
Deskripsi	Mesin utama diarahkan kearah yang ditentukan untuk memproduksi momen kontrol.	Sebuah *vane* ditempatkan pada tempat pembuangan tekanan untuk mengarahkan tekanan kearah tertentu.	Menempatkan sebuah pendorong kecil yang tidak segaris dengan pusat massa untuk menghasilkan momen kontrol .	Menempatkan suatu roda berat yang melakukan akselerasi dan deakselerasi untuk menghasilkan momen kontrol.
Kelebihan	Tekanan dari udara pendorong tersalurkan secara penuh.	Mudah dalam penerapan karena proses desain yang sederhana.	Tekanan udara pendorong tersalurkan secara merata.	Tidak merubah bentuk aerodinamis dari badan roket dan tekanan udara tersalurakan secara penuh
Kekurangan	Sulit diterapkan karena proses desain yang sulit dan struktur yang harus handal. Tidak terdapat mekanisme pengontrol putaran pada badan roket.	Adanya tambahan daya hambat karena pemasangan *vane.*	Adanya biaya tambahan untuk membuat pendorong kecil. Tidak terdapat mekanisme pengontrol putaran pada badan roket.	Tambahan beban roda yang akan mengurangi daya dorong roket.
Contoh	Saturn IB/V, Space shuttle, Apollo Lunar Module	Roket Goddard, misil V2	Satelit, Space shuttle, Apollo Lunar Module	Satelit, robot Murata boy balancing

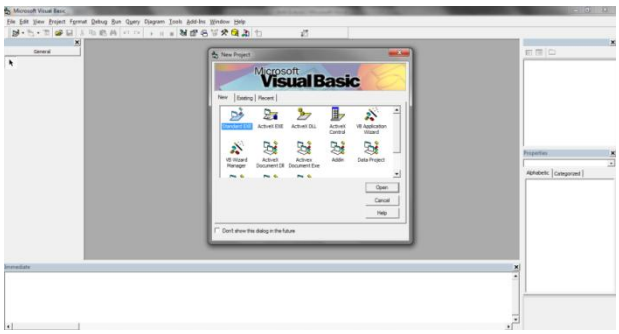
Beberapa konsep pada tabel II-3 merupakan konsep yang telah diterapkan pada roket diberbagai negara. Pemilihan konsep akan bergantung terhadap faktor-faktor seperti kemudahan penerapan, efisiensi daya dorong, massa roket, biaya dan lainnya.

2.2 Pendekatan secara perangkat lunak

Pada bagian ini akan memaparkan beberapa perangkat lunak yang digunakan dalam pembuatan topik tugas akhir ini.

2.2.1 Microsoft Visual Basic 6.0

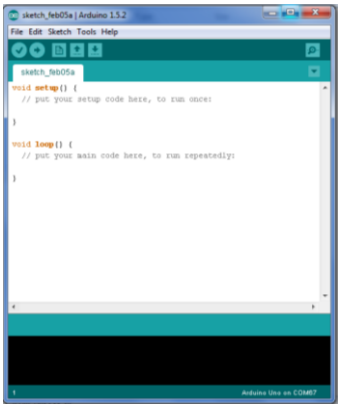
Microsoft Visual Basic merupakan salah satu produk buatan perusahaan Microsoft. Perangkat lunak ini pertama kali diluncurkan pada tahun 1991. Pada tahun 1998, Microsoft mengeluarkan versi stabil dari perangkat ini yaitu Microsoft Visual Basic 6.0. Visual Basic merupakan perangkat yang digunakan dalam pembuatan perangkat lunak yang berhubungan langsung dengan pengguna.



Gambar II-9 *Graphical User Interface*(GUI) Visual Studio 6.0

2.2.2 Arduino Sketch 1.5.2

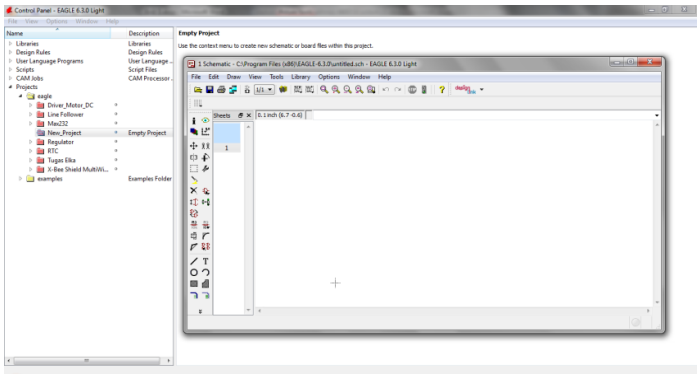
Arduino Sketch adalah perangkat lunak yang digunakan untuk merancang algoritma yang akan ditanamkam pada mikrokontroler. Perangkat ini menggunakan pemrograman berbahasa C untuk menghasilkan suatu algoritma. Perangkat ini juga dilengkapi dengan terminal (pada windows xp merupakan Hyperterminal) yang dapat digunakan untuk menerima data dari mikrokontroler melalui port serial.



Gambar II-10. *Graphical User Interface*(GUI) pada Arduino Sketch 1.5.2

2.2.3 Eagle CadSoft 6.3.0

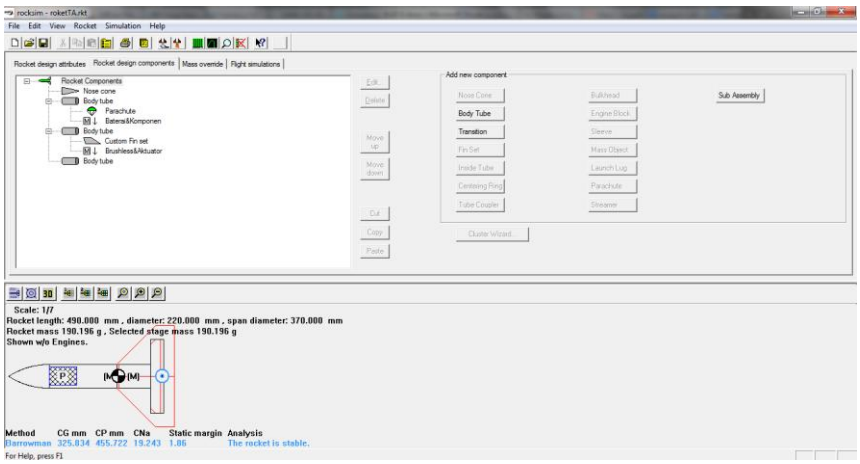
Eagle CadSoft 6.3.0 merupakan perangkat lunak untuk melakukan proses pendesainan skema dari rangkaian elektronika pada PCB(Printable Circuit Board). PCB sendiri akan berisikan plot penyimpanan komponen beserta jalur-jalur yang menghubungkan tiap komponen elektronika. PCB dapat ditemukan pada berbagai macam alat elektronika contohnya televisi, radio FM, komputer dan sebagainya.



Gambar II-11 *Graphical User Interface*(GUI) Eagle Cadsoft 6.3.0

2.2.4 RockSim

RockSim merupakan perangkat lunak yang digunakan untuk merancang badan roket. Dari hasil perancangan badan roket akan dihasilkan berbagai data seperti pusat massa, pusat tekanan, drag, analisis kestabilan, ketinggian dan sebagainya. Pada perangkat lunak proses peluncuran dapat disimulasikan tetapi roket hanya dapat menggunakan pendorong berbahan bakar propelan sehingga dilakukan pendekatan dengan memilih jenis propelan yang mendekati spesifikasi daya dorong baik pada roket air ataupun roket motor elektrik.



Gambar II-12 *Graphical User Interface*(GUI) pada Rocksim

2.3 Pendekatan secara perangkat keras

2.3.1 Mikrokontroler

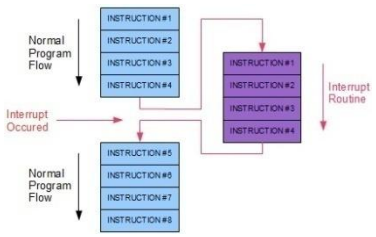
Mikrokontroler adalah sebuah komputer kecil pada suatu *integrated ciruit* yang berfungsi untuk membaca data, melakukan kalkulasi terbatas pada data dan mengendalikan lingkungannya berdasarkan kalkulasi tersebut. Penggunaan utama mikrokontroler adalah mengontrol operasi sebuah mesin mengunakan program yang disimpan dalam ROM dan tidak berubah sepanjang umur sistem tersebut.

Arsitektur pendesaian mikrokontroler yang populer ada 2 jenis yakni *Complex Instruction Set Computing* (CISC) dan *Reduce Instruction Set Computing* (RISC). Pada CISC terdapat efisiensi dalam mengurangi ukuran program yang ditanamkan dengan cara memasukan sejumlah instruksi kompleks. Instruksi kompleks ekivalen dari tiga atau empat instruksi sederhana. Namun diperlukan compiler efisiensi tinggi untuk menggunakan instruksi kompleks. Pada RISC, instruksi yang digunakan adalah instruksi sederhana dimana tiap panjang instruksinya sama untuk semua instruksi. Penggunaan instruksi sederhana pada RISC mempermudah proses pendeteksian suatu kegagalan pada pekerjaan sistem.

Pada sistem mikrokontroler terdapat beberapa fitur atau fungsi-fungsi yang menyertainya antara lain interupsi, *Timer/Counter*, *Pulse Width Modulation* (PWM), *Universal synchronous asynchronous receiver and transmitter*(USART), *Inter Integrated Circuit*(I2C) dan lainnya.

2.3.1.1 Interupsi

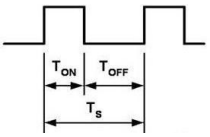
Interupsi adalah suatu kejadian dalam sistem komputer yang meminta pelayanan khusus pada CPU ketika CPU sedang melakukan pemrosesan. Dalam merespon interupsi, CPU menunda eksekusi program saat itu dan melakukan percabangan ke *Interrupt Service Routine*(ISR) atau disebut juga vektor interupsi. ISR adalah sebuah program yang melayani interupsi. Setelah eksekusi ISR, CPU akan kembali kepada programnya semula.



Gambar II-13 *Interrupt Service Routine*(ISR)

2.3.1.2 *Timer/Counter*

Timer pada dasarnya hanya menghitung pulsa *clock*. Frekuensi pulsa *clock* akan bergantung pada penggunaan kristal pada sistem mikroprosesor. Pada mikroprosesor terdapat beberapa hal yang berhubungan dengan *Timer* yakni frekuensi, perioda dan *Duty Cycle*. Frekuensi adalah jumlah pulsa yang terjadi dalam satuan waktu. Perioda adalah kebalikan dari frekuensi. *Duty Cycle* adalah perbandingan antara waktu pulsa *High* dan perioda yang dinyatakan dalam persen.



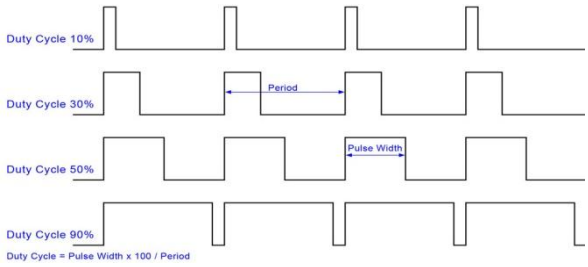
Gambar II-14 *Duty Cycle*

Dengan persamaan sebagai berikut

$$Duty\ Cycle = \frac{T_{ON}}{T} x100\% = \frac{T_{ON}}{T_{ON} + T_{OFF}} x100\%$$

2.3.1.3 *Pulse Width Modulation*(PWM)

Pulse Width Modulation(PWM) adalah suatu fitur yang dimiliki oleh mikrokontroler. PWM bekerja dengan cara melakukan perubahan level tegangan pada frekuensi tertentu. Konsep PWM dapat digunakan dalam beberapa hal seperti pengatur kecepatan motor, digital to analog converter dan sebagainya.

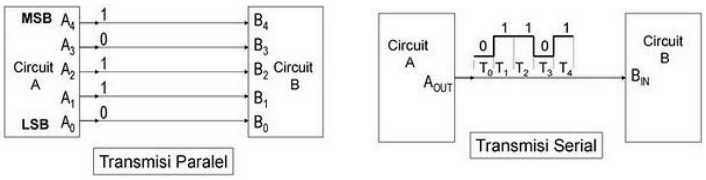


Gambar II-15 Modulasi Lebar Pulsa dengan Berbagai *Duty Cycle*

Berdasarkan gambar II-15, bila kita katakan sumber tegangan bernilai 5V. Maka pada *duty cycle* 10%, keluaran PWM bernilai 0,5V. Demikian juga, pada *duty cycle* 30% keluaran PWM pernilai 1,5V, *duty cycle* 50% keluaran PWM bernilai 2,5V, dan pada *duty cycle* 90% keluaran PWM bernilai 4V. Oleh karena itu, kita dapat simpulkan dengan adanya manipulasi terhadap perubahan level tegangan maka terjadi perubahan tegangan. Hal tersebutlah yang dimanfaatkan untuk pengaturan kecepatan gerak motor.

2.3.1.4 Universal asynchronous receiver and transmitter(UART)

Komunikasi serial merupakan salah satu alternatif yang telah banyak digunakan sebagai pengganti komunikasi paralel pada beberapa perangkat. Meskipun pemrosesan data pada CPU terjadi dalam bentuk paralel, pengiriman data serial tetap dapat dilakukan karena adanya proses konversi paralel ke serial. Proses konversi tersebut dilakukan oleh *register* geser dengan cara melakukan pergeseran bit-bit data setelah jumlah *clock* yang ditentukan terpenuhi.

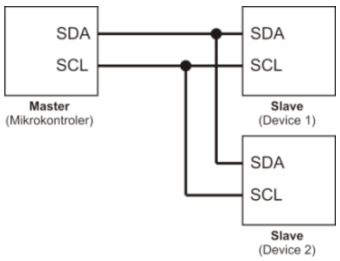


Gambar II-16 Perbandingan komunikasi paralel dan serial

Komunikasi *Universal synchronous asynchronous receiver and transmitter* (USART) adalah fasilitas komunikasi serial yang dimiliki beberapa jenis mikrokontroler. Komunikasi ini bekerja baik secara sinkron maupun asinkron. komunikasi sinkron adalah komunikasi antara perangkat dimana sinyal *clock* yang digunakan pengirim dan penerima berasal dari satu sumber *clock*. Sedangkan komunikasi asinkron memiliki sumber *clock* masing-masing.

2.3.1.5 Inter Integrated Circuit (I2C)

I2C merupakan salah satu fitur baru dalam teknologi komunikasi. I2C memiliki kelebihan pada penggunaan port dibandingkan komunikasi UART. Pada UART tiap perangkat memerlukan satu jalur untuk menerima data dan satu jalur mengirim data sedangkan fitur I2C hanya menggunakan 2 buah *port* untuk semua perangkat sebagai media komunikasi yakni jalur data(menerima/mengirim) dan sinkronisasi *clock*. Berdasarkan protokol I2C, jumlah maksimum perangkat yang terhubung pada sepasang *port* SDA dan SCL adalah sebanyak 128 perangkat.



Gambar II-17 *Wiring Diagram I2C*

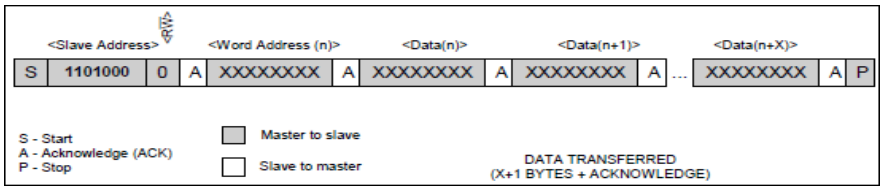
Metode transfer data yang dimiliki I2C ada 2 macam yakni

1. *Write Register*

   Metode ini dapat dikatakan sebagai proses penulisan data oleh *master* terhadap alamat *register slave*. Berikut ini langkah-langkah proses penulisan *register slave* pada I2C.

   1. Byte pertama yang dikirimkan oleh *master* berupa alamat *slave* yang dituju. *Slave* akan mengirimkan balasan berupa 1-bit *acknowledge* dimana jika *acknowledge* bernilai 1 maka *slave* dapat diakses.
   2. Byte kedua yang dikirimkan oleh *master* berupa alamat *register slave* yang dituju. *Slave* akan mengirimkan balasan berupa 1-bit *acknowledge* dimana jika *acknowledge* bernilai 1 maka *register slave* dapat ditulis.
   3. Byte ketiga yang dikirimkan berupa oleh *master* berupa data yang akan ditulis. *Slave* akan mengirimkan balasan berupa 1-bit *acknowledge* dimana jika *acknowledge* bernilai 1 maka *register slave* berhasil ditulis. Pada Byte keempat dan selanjutnya yang dikirimkan berupa data yang akan ditulis.

   Proses komunikasi data akan berakhir disaat tidak adanya data yang dikirimkan oleh *master* kepada *slave.*



Gambar II-18 Protokol penulisan data pada I2C

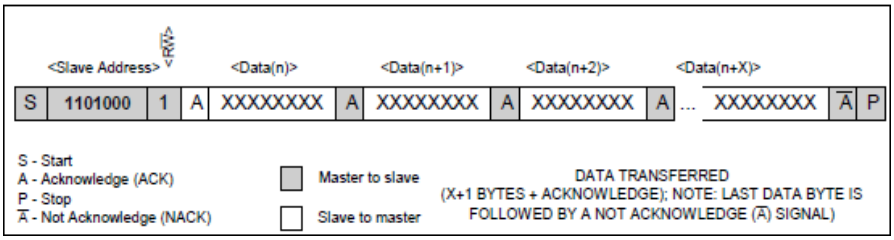
2. *Read Register*

   Metode ini dapat dikatakan sebagai proses permintaan data oleh *master* kepada *slave*. Berikut ini langkah-langkah proses pembacaan *register slave* pada I2C.

   1. Byte pertama yang dikirimkan oleh *master* berupa alamat *slave* yang dituju. *Slave* akan mengirimkan balasan berupa 1-bit *acknowledge* dimana jika *acknowledge* bernilai 1 maka *slave* dapat diakses.
   2. Byte kedua dan selanjutnya adalah proses pengiriman balik oleh *slave* kepada *master* yang didalamnya berupa data isi tiap *register*. *Master*

akan mengirimkan balasan berupa 1-bit *acknowledge* dimana jika *acknowledge* bernilai 1 maka *slave* akan melanjutkan pengiriman data yang ada pada *register* berikutnya.

Proses *Read register* akan berakhir disaat master mengirimkan bit *acknowledge* bernilai 0 atau dapat disebut bit *STOP*



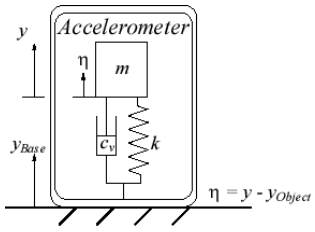
Gambar II-19 Protokol pembacaan data pada I2C

2.3.2 Sensor *Inertia Measurment Unit(IMU)*

Inertia Measurment Unit (IMU) merupakan gabungan beberapa sensor yang digunakan untuk mengukur perubahan inersia yang terjadi pada suatu benda. Kombinasi yang digunakan berupa sensor *accelerometer* dan *gyroscope* yang kadangkala ditambahkan beberapa sensor lain seperti *magnetometer* dan *barometer*. Sensor ini telah banyak digunakan pada beberapa perangkat navigasi pada kendaraan maupun *mobile device*.

2.3.2.1 Sensor *Accelerometer*

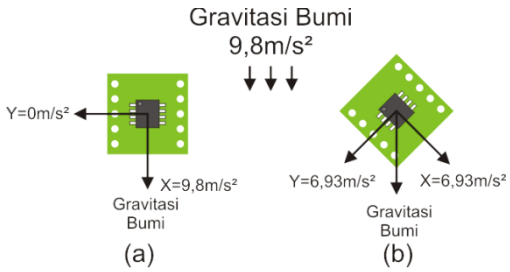
Sensor *accelerometer* adalah sensor yang mengukur akselerasi pada lingkungannya. Beberapa jenis accelerometer menggunakan konsep kerja pegas. Perubahan panjang yang terjadi pada pegas akan menjadi nilai akselerasi yang dihasilkan.



Gambar II-20 Konsep sensor *accelerometer*

Selain mengukur akselerasi, *accelerometer* mampu mengukur kemiringan suatu benda dengan melakukan proses perbandingan terhadap gaya gravitasi bumi

yang dapat dirasakan oleh sensor ini. Berikut ini ilustrasi pembacaan gravitasi bumi.



Gambar II-21 Pembacaan gravitasi bumi oleh sensor *accelerometer*

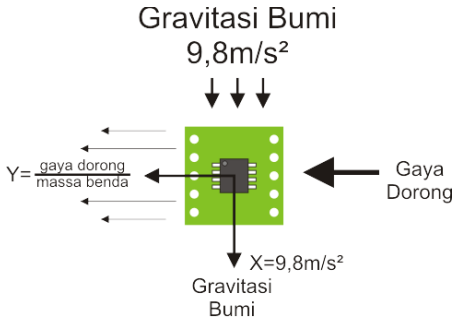
Pada gambar II-21 tersebut terlihat adanya perubahan pembacaan percepatan gaya gravitasi bumi pada saat sensor *accelerometer* dimiringkan. Pada kondisi (a), sumbu x membaca keseluruhan gaya gravitasi bumi. Sedangkan pada kondisi (b), sumbu x dan sumbu y memiliki besar masing-masing pembacaan gaya gravitasi bumi. Dengan memanfaatkan persamaan dibawah ini, perubahan sudut yang terjadi pada sensor *accelerometer* dapat diketahui.

$$Degree = \tan^{-1}\left(\frac{Accelerometer\ X}{Accelerometer\ Y}\right)$$

Pada kondisi (a) : $Degree = \tan^{-1}\left(\frac{9{,}18\ m/s^2}{0\ m/s^2}\right) = \tan^{-1}(\infty) = 90°$

Pada kondisi (b) : $Degree = \tan^{-1}\left(\frac{6{,}93\ m/s^2}{6{,}93\ m/s^2}\right) = \tan^{-1}(1) = 45°$

Perhitungan tersebut membuktikan bahwa dengan memanfaatkan gaya gravitasi bumi, sensor *accelerometer* dapat mengukur sudut benda. Namun proses pembacaan akan terganggu disaat sensor mendapat *noise* berupa getaran atau akselerasi dari luar.

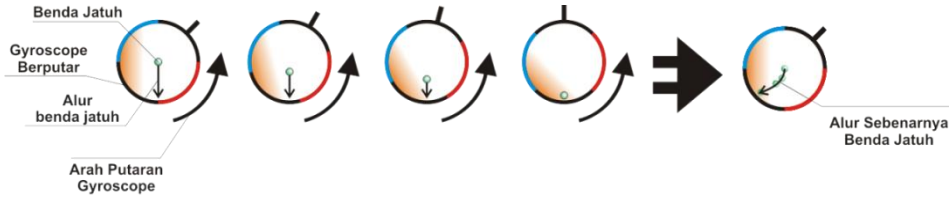


Gambar II-22 Noise *Accelerometer*

Noise tersebut merupakan kelemahan dari sensor *accelerometer*. Penggunaan low-pass filter dan pembatasan besar percepatan merupakan salah satu solusi untuk mengatasi kelemahaan *accelerometer* untuk pembacaan sudut.

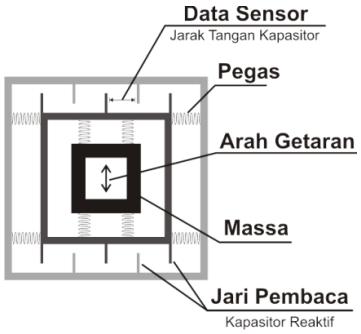
2.3.2.2 Sensor *Gyroscope*

Gyroscope adalah alat yang digunakan untuk kecepatan putar suatu benda. Pada fisika terdapat suatu efek yang dinamakan efek coriolis. Efek coriolis merupakan defleksi benda yang bergerak ketika dilihat dari sudut pandang rotasi.



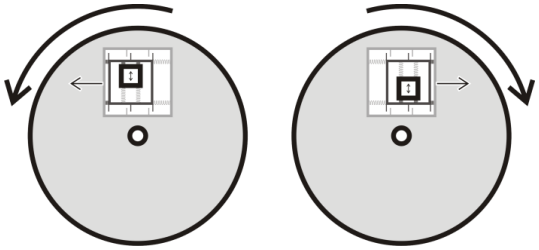
Gambar II-23 efek coriolis

Beberapa jenis sensor gyroscope menggunakan efek ini untuk membaca kecepatan putaran.



Gambar II-24 Rangkaian elektronik *gyroscope*

Pada saat sensor gyroscope yang mengalami pergerakan memutar pada sumbu-sumbunya, efek Coriolis akan menyebabkan getaran yang akan dideteksi oleh suatu rangkaian elektronik.



Gambar II-25 Rangkaian elektronik *gyroscope* saat berputar

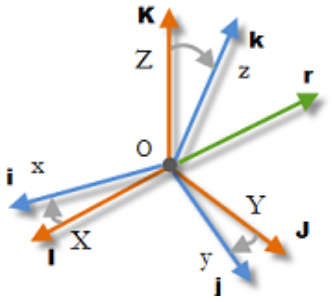
Namun dikarenakan sensitifitas pembacaan getaran sensor *gyroscope* yang sangat sensitif maka pada saat terjadi getaran yang sangat kecil, sensor *gyroscope* membaca getaran kecil tersebut sebagai proses perputaran benda. Penggunaan *low-pass filter* dapat menjadi solusi kelemahan sensor *gyroscope*

2.3.2.3 Sensor Magnetometer

Sensor magnetometer adalah sensor yang mendeteksi kekuatan medan magnet pada suatu lingkungan. Magnetometer juga dapat digunakan untuk mendeteksi logam metal. Namun pada tugas akhir ini, magnetometer digunakan untuk mendeteksi arah utara. Nilai keluaran dari sensor ini berupa nilai Gauss atau nilai kemagnetan pada suatu medan. Sensor ini memiliki kelemahan yakni sensitif terhadap keberadaan benda logam. Oleh karena itu, penempatan sensor ini diharapkan agak menjauh dari benda logam dan benda yang menghasilkan elektromagnetik.

2.3.2.4 DCM atau Rotation Matrix

Berbicara soal kinematik orientasi dengan menghitung orientasi benda terhadap koordinat global sistem. Jika kita sebut bahwa sumbu xyz adalah koordinat benda dan sumbu XYZ adalah koordinat global maka kita definisikan i, j, k sebagai kesatuan dari vektor co-directional dari benda yang dengan kata lain merupakan versor(normalisasi vektor) dari xyz dan vektor I,J,K merupakan versor dari koordinat global XYZ.



Gambar II-26 vektor benda(biru) dan vektor globalnya(merah)

Berdasarkan gambar II-26, vektor $\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}$ dan $\boldsymbol{I}, \boldsymbol{J}, \boldsymbol{K}$ dapat dimisalkan menjadi

$i^B = \begin{bmatrix}\mathbf{1}\\\mathbf{0}\\\mathbf{0}\end{bmatrix}$, $j^B = \begin{bmatrix}\mathbf{0}\\\mathbf{1}\\\mathbf{0}\end{bmatrix}$, $k^B = \begin{bmatrix}\mathbf{0}\\\mathbf{0}\\\mathbf{1}\end{bmatrix}$ dan $I^G = \begin{bmatrix}\mathbf{1}\\\mathbf{0}\\\mathbf{0}\end{bmatrix}$, $J^G = \begin{bmatrix}\mathbf{0}\\\mathbf{1}\\\mathbf{0}\end{bmatrix}$, $K^G = \begin{bmatrix}\mathbf{0}\\\mathbf{0}\\\mathbf{1}\end{bmatrix}$

Ambilah contoh proyeksi panjang vektor $\boldsymbol{i}$ terhadap sumbu global X menjadi

$i_x^G = |\boldsymbol{i}|cos(X, i) = cos(I, i)$ dimana

- $|\boldsymbol{i}|$ merupakan nilai normalisasi dari vektor $\boldsymbol{i}$
- $cos(I, i)$ merupakan cosinus dari sudut yang dibentuk oleh vektor $\boldsymbol{I}$ dan $\boldsymbol{i}$.

Dengan menggunakan $|\boldsymbol{I}| = 1$ dan $|\boldsymbol{i}|$ =1, persamaan tersebut dapat dituliskan.

$i_x^G = cos(I, i) = \frac{\boldsymbol{I}.\boldsymbol{i}}{|\boldsymbol{I}||\boldsymbol{i}|}$ **ingatlah bahwa** $cos(a, b) = \frac{\boldsymbol{a}.\boldsymbol{b}}{|\boldsymbol{a}||\boldsymbol{b}|}$

$i_x^G = |\boldsymbol{I}||\boldsymbol{i}|cos(I, i) = \boldsymbol{I}.\boldsymbol{i}$

proyeksi vektor $\boldsymbol{i}$ terhadap sumbu global Y dan Z menjadi

$$\boldsymbol{i_y^G} = |i|\boldsymbol{cos}(\boldsymbol{Y},\boldsymbol{i}) = \boldsymbol{cos}(\boldsymbol{J},\boldsymbol{i}) \qquad \text{dan} \qquad \boldsymbol{i_z^G} = |i|\boldsymbol{cos}(\boldsymbol{Z},\boldsymbol{i}) = \boldsymbol{cos}(\boldsymbol{K},\boldsymbol{i})$$
$$\boldsymbol{i_y^G} = \boldsymbol{cos}(\boldsymbol{J},\boldsymbol{i}) \qquad\qquad\qquad\qquad\quad \boldsymbol{i_z^G} = \boldsymbol{cos}(\boldsymbol{I},\boldsymbol{i})$$
$$= |J||i|\boldsymbol{cos}(\boldsymbol{J},\boldsymbol{i}) = J.i \qquad\qquad\qquad\qquad\quad = |K||i|\boldsymbol{cos}(\boldsymbol{K},\boldsymbol{i}) = K.i$$

maka $\boldsymbol{i^G} = \begin{bmatrix} i_x^G \\ i_y^G \\ i_z^G \end{bmatrix} = \begin{bmatrix} \boldsymbol{I.i} \\ \boldsymbol{J.i} \\ \boldsymbol{K.i} \end{bmatrix}$, $\boldsymbol{j^G} = \begin{bmatrix} j_x^G \\ j_y^G \\ j_z^G \end{bmatrix} = \begin{bmatrix} \boldsymbol{I.j} \\ \boldsymbol{J.j} \\ \boldsymbol{K.j} \end{bmatrix}$, $\boldsymbol{k^G} = \begin{bmatrix} k_x^G \\ k_y^G \\ k_z^G \end{bmatrix} = \begin{bmatrix} \boldsymbol{I.k} \\ \boldsymbol{J.k} \\ \boldsymbol{K.k} \end{bmatrix}$

Semua proyeksi vektor $\boldsymbol{i},\boldsymbol{j},\boldsymbol{k}$ terhadap sumbu global dirangkum dalam *Direction Cosine Matrix* (DCM). DCM merupakan cosinus semua sudut kombinasi yang terjadi pada versor benda dan versor global.

Vektor $\boldsymbol{i},\boldsymbol{j},\boldsymbol{k}$ dalam koordinat global menjadi

$$[\boldsymbol{i^G} \quad \boldsymbol{j^G} \quad \boldsymbol{k^G}] = \begin{bmatrix} \boldsymbol{I.i} & \boldsymbol{I.j} & \boldsymbol{I.k} \\ \boldsymbol{J.i} & \boldsymbol{J.j} & \boldsymbol{J.k} \\ \boldsymbol{K.i} & \boldsymbol{K.j} & \boldsymbol{K.k} \end{bmatrix} = \begin{bmatrix} cos(I,i) & cos(I,j) & cos(I,k) \\ cos(J,i) & cos(J,j) & cos(J,k) \\ cos(K,i) & cos(K.j) & cos(K,k) \end{bmatrix} = \boldsymbol{DCM^G}$$ **...(persamaan 1.1)**

Vektor $\boldsymbol{I},\boldsymbol{J},\boldsymbol{K}$ dalam kordinat benda menjadi

$$[\boldsymbol{I^B} \quad \boldsymbol{J^B} \quad \boldsymbol{K^B}] = \begin{bmatrix} \boldsymbol{i.I} & \boldsymbol{i.J} & \boldsymbol{i.K} \\ \boldsymbol{j.I} & \boldsymbol{j.J} & \boldsymbol{j.K} \\ \boldsymbol{k.I} & \boldsymbol{k.J} & \boldsymbol{k.K} \end{bmatrix} = \begin{bmatrix} cos(i,I) & cos(i,J) & cos(i,K) \\ cos(j,I) & cos(j,J) & cos(j,K) \\ cos(k,I) & cos(k,J) & cos(k,K) \end{bmatrix} = \boldsymbol{DCM^B}$$ **...(persamaan 1.2)**

Hubungan persamaan 1.1 dan persamaan 1.2 akan memunculkan persamaan

$$\boldsymbol{DCM^G} = (\boldsymbol{DCM^B})^T \; atau \; \boldsymbol{DCM^B} = (\boldsymbol{DCM^G})^T$$

Saat persamaan 1.1 dan persamaan 1.2 digabungkan maka didapatkan matriks.

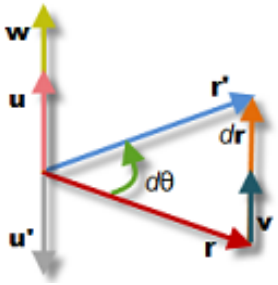
$$\boldsymbol{DCM^B DCM^G} = (\boldsymbol{DCM^G})^T \boldsymbol{DCM^G} = \begin{bmatrix} \boldsymbol{i^{GT}} \\ \boldsymbol{j^{GT}} \\ \boldsymbol{k^{GT}} \end{bmatrix} [\boldsymbol{i^G} \quad \boldsymbol{j^G} \quad \boldsymbol{k^G}] = \begin{bmatrix} \boldsymbol{i^{GT}.i^G} & \boldsymbol{i^{GT}.j^G} & \boldsymbol{i^{GT}.k^G} \\ \boldsymbol{j^{GT}.i^G} & \boldsymbol{j^{GT}.j^G} & \boldsymbol{j^{GT}.k^G} \\ \boldsymbol{k^{GT}.i^G} & \boldsymbol{k^{GT}.j^G} & \boldsymbol{k^{GT}.k^G} \end{bmatrix}$$

dimana $\boldsymbol{i^{GT}.i^G} = |\boldsymbol{i^G}||\boldsymbol{i^G}|\cos(0) = 1$

vektor $\boldsymbol{i}$ dan vektor $\boldsymbol{j}$ adalah tegak lurus maka $\boldsymbol{i^{GT}.j^G} = 0$

$$\boldsymbol{DCM^B DCM^G} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$ **...(persamaan 1.3)**

DCM mendefinisikan rotasi dari benda relatif terhadap sesuatu yang lain. DCM juga dapat digunakan untuk menentukan koordinat global dari perubahan vektor jika kita mengetahui koordinat pada benda (begitupula sebaliknya).



Gambar II-27 Perputaran vektor pada sumbu

$$\boldsymbol{r} = r(t), \qquad \boldsymbol{r}' = \boldsymbol{r}(t+dt), \qquad dan \qquad d\boldsymbol{r} = \boldsymbol{r}' - \boldsymbol{r}$$

vektor **r'** adalah perputaran vektor **r** dengan sudut vektor d**θ** pada interval dt pada sumbu vektor **u** maka vektor **u** adalah *cross product* dari **r** dan **r'**

$$\boldsymbol{u} = \frac{\boldsymbol{r}x\boldsymbol{r}'}{|\boldsymbol{r}x\boldsymbol{r}'|} = \frac{\boldsymbol{r}x\boldsymbol{r}'}{|\boldsymbol{r}|.|\boldsymbol{r}'|\sin(d\theta)} = \frac{\boldsymbol{r}x\boldsymbol{r}'}{|\boldsymbol{r}|^2\sin(d\theta)} \qquad \textbf{...(persamaan 2.1)}$$

dimana $|\boldsymbol{r}| = |\boldsymbol{r}'|$ karena perputaran tidak mengubah panjang vektor.

Kecepatan dari perpindahan vektor r dapat didefinisikan dalam persamaan berikut.

$$\boldsymbol{v} = \frac{d\boldsymbol{r}}{dt} = \frac{\boldsymbol{r}'-\boldsymbol{r}}{dt} \qquad \textbf{...(persamaan 2.2)}$$

Kecepatan putar dari perpindahan vektor r didefinisikan dalam persamaan berikut.

$$\boldsymbol{w} = \frac{d\boldsymbol{\theta}}{dt}\boldsymbol{u} \qquad \textbf{...(persamaan 2.3)}$$

Penggabungan persamaan 2.3 dan persamaan 2.1 didapatkan persamaan berikut.

$\boldsymbol{w} = \frac{d\boldsymbol{\theta}}{dt}\boldsymbol{u} = \frac{d\boldsymbol{\theta}}{dt}\frac{\boldsymbol{r}x\boldsymbol{r}'}{|\boldsymbol{r}|^2\sin(d\theta)}$ saat d**θ** mendekati 0 maka sin(dθ) mendekati 0

$$\boldsymbol{w} = \frac{\boldsymbol{r}x\boldsymbol{r}'}{|\boldsymbol{r}|^2 dt} \qquad \textbf{...(persamaan 2.4)}$$

$\boldsymbol{w} = \frac{\boldsymbol{r}x(\boldsymbol{r}+d\boldsymbol{r})}{|\boldsymbol{r}|^2 dt} = \frac{\boldsymbol{r}x\boldsymbol{r}+\boldsymbol{r}xd\boldsymbol{r}}{|\boldsymbol{r}|^2 dt} = \frac{\boldsymbol{r}x\boldsymbol{r}}{|\boldsymbol{r}|^2 dt} + \frac{\boldsymbol{r}xd\boldsymbol{r}}{|\boldsymbol{r}|^2 dt}$ dimana *cross-product* dari $\boldsymbol{r}x\boldsymbol{r} = \boldsymbol{0}$

$$\boldsymbol{w} = \frac{\boldsymbol{r}xd\boldsymbol{r}}{|\boldsymbol{r}|^2 dt} = \boldsymbol{r}x\frac{d\boldsymbol{r}}{|\boldsymbol{r}|^2 dt} = \boldsymbol{r}x\left(\frac{d\boldsymbol{r}}{dt}\right)/|\boldsymbol{r}|^2 = \boldsymbol{r}x\boldsymbol{v}/|\boldsymbol{r}|^2 = \frac{\boldsymbol{r}x\boldsymbol{v}}{|\boldsymbol{r}|^2} \qquad \textbf{...(persamaan 2.5)}$$

$\boldsymbol{w}x\boldsymbol{r} = \frac{\boldsymbol{r}x\boldsymbol{v}}{|\boldsymbol{r}|^2}x\boldsymbol{r}$ menggunakan aturan *triple product* $(axb)xc = (a.c)b - (b.c)a$

$$\boldsymbol{w}x\boldsymbol{r} = \frac{(\boldsymbol{r}.\boldsymbol{r})\boldsymbol{v}+(\boldsymbol{v}.\boldsymbol{r})\boldsymbol{r}}{|\boldsymbol{r}|^2} = \frac{|\boldsymbol{r}|^2.\boldsymbol{v}}{|\boldsymbol{r}|^2} + \frac{0}{|\boldsymbol{r}|^2} = \frac{|\boldsymbol{r}|^2.\boldsymbol{v}}{|\boldsymbol{r}|^2} = \boldsymbol{v} \qquad \textbf{...(persamaan 2.6)}$$

$$d\boldsymbol{r_1} = \boldsymbol{dt}\, v_1 = \boldsymbol{dt}\,(w_x.r); \qquad d\boldsymbol{r_2} = \boldsymbol{dt}\, v_2 = \boldsymbol{dt}\,(w_y.r); \qquad d\boldsymbol{r_3} = \boldsymbol{dt}\, v_3 = \boldsymbol{dt}\,(w_z.r)$$

jika $d\boldsymbol{r}_1, d\boldsymbol{r}_2, d\boldsymbol{r}_3$ digabungkan maka akan menghasilkan persamaan berikut.

$d\boldsymbol{r} = d\boldsymbol{r}_1 + d\boldsymbol{r}_2 + d\boldsymbol{r}_3 = dt\,(w_x + w_y + w_z)x\boldsymbol{r}$ maka $\boldsymbol{v} = \frac{d\boldsymbol{r}}{dt} = (w_x + w_y + w_z)x\boldsymbol{r}$

Keluaran *accelerometer* dicantumkan dalam vektor $\boldsymbol{A} = \{A_x, A_y, A_z\}$

Keluaran *magnetometer* dicantumkan dalam vektor $\boldsymbol{M} = \{M_x, M_y, M_z\}$

$\boldsymbol{DCM^G} = (\boldsymbol{DCM^B})^T = [\boldsymbol{I^B} \quad \boldsymbol{J^B} \quad \boldsymbol{K^B}]^T$

$\boldsymbol{K^B} = -\boldsymbol{A}$ karena sifat *accelerometer* yang mendeteksi adanya gravitasi bumi.

$\boldsymbol{I^B} = \boldsymbol{M}$ karena *magnetometer* mengetahui arah utara seperti pada sumbu $\boldsymbol{I^B}$ yang merupakan penunjuk arah kompas.

Dengan mengetahui $\boldsymbol{K^B}$ dan $\boldsymbol{I^B}$, $\boldsymbol{J^B}$ dapat diketahui dengan $\boldsymbol{J^B} = \boldsymbol{K^B} x \boldsymbol{I^B}$ dengan aturan tangan kanan. Jika komponen DCM terpenuhi, mengapa *gyroscope* dibutuhkan. *Gyroscope* digunakan untuk "*fine-tune*" DCM karena *gyroscope* lebih presisi dibanding *accelerometer* dan *magnetometer*. Keluaran *gyroscope* berupa kecepatan putaran yang dibentuk dalam matriks $\boldsymbol{w} = \{w_x, w_y, w_z\}$. Berikut ini contoh penggunaan gyroscope pada sumbu Z.

$\boldsymbol{K^B_{1G}} = \boldsymbol{K^B_0} + dt\, \boldsymbol{v} = \boldsymbol{K^B_0} + dt\, (\boldsymbol{w_g}\, x\, \boldsymbol{K^B_0}) = \boldsymbol{K^B_0} + (d\boldsymbol{\theta}_g\, x\, \boldsymbol{K^B_0})$ dimana $d\boldsymbol{\theta}_g = \boldsymbol{w_g}\, dt$ karena $\boldsymbol{w_g}$ merupakan kecepatan putaran yang dihitung oleh gyroscope.

$\boldsymbol{w_a} = \frac{\boldsymbol{K^B_0}\, x\, \boldsymbol{v_a}}{|\boldsymbol{K^B_0}|^2}$ karena $\boldsymbol{v_a} = \frac{d\boldsymbol{K^B}}{dt} = \frac{(\boldsymbol{K^B_{1A}} - \boldsymbol{K^B_0})}{dt}$ merupakan kecepatan linier dari $\boldsymbol{K^B_0}$ dan $|\boldsymbol{K^B_0}|^2 = 1$. Jika digabungkan dapat dituliskan menjadi sebagai berikut.

$d\boldsymbol{\theta}_{acc} = dt\, \boldsymbol{w_a} = dt(\boldsymbol{K^B_0}\, x\, \boldsymbol{v_a}) = dt\, \left(\boldsymbol{K^B_0}\, x \frac{(\boldsymbol{K^B_{1A}} - \boldsymbol{K^B_0})}{dt}\right) = \boldsymbol{K^B_0}\, x\, (\boldsymbol{K^B_{1A}} - \boldsymbol{K^B_0})$

Penggabungkan *accelerometer* dan *gyroscope* menggunakan persamaan berikut.

$d\boldsymbol{\theta} = \frac{(s_a\, d\boldsymbol{\theta}_a + s_g\, d\boldsymbol{\theta}_g)}{s_a + s_g}$ dimana s adalah beban untuk menentukan berapa besar sensor berpengaruh pada perhitungan perubahan sudut yang terjadi. Persamaan tersebut merupakan *complementary filter*. Dengan cara yang sama pada *accelerometer*, perubahan sudut oleh *magnetometer* dapat dituliskan dalam persamaan berikut.

$d\boldsymbol{\theta}_{magnet} = dt\, \boldsymbol{w_{magnet}} = \boldsymbol{I^B_0}\, x\, (\boldsymbol{I^B_{1M}} - \boldsymbol{I^B_0})$

Jadi kesimpulan pada perhitungan perubahan sudut yang terjadi.

$\boldsymbol{K^B_1} = \boldsymbol{K^B_0} + (d\boldsymbol{\theta}\, x\, \boldsymbol{K^B_0})$ hal tersebut berlaku pada sumbu lainnya sehingga

$\boldsymbol{I^B_1} = \boldsymbol{I^B_0} + (d\boldsymbol{\theta}\, x\, \boldsymbol{I^B_0})$

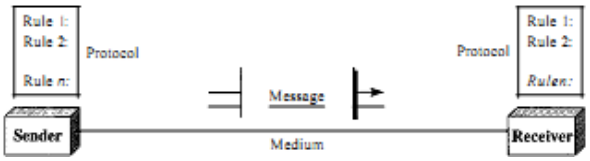
$\boldsymbol{J^B_1} = \boldsymbol{J^B_0} + (d\boldsymbol{\theta}\, x\, \boldsymbol{J^B_0})$

2.3.3 Komunikasi data

Komunikasi data adalah merupakan bagian dari telekomunikasi yang secara khusus berkenaan dengan transmisi atau pemindahan data dan informasi diantara suatu perangkat ke perangkat lainnya yang dikirimkan melalui media transmisi data. Komunikasi data merupakan bagian vital karena sistem ini menyediakan informasi sebagai upaya penentuan keputusan yang akan dilakukan. Untuk mendapatkan data perilaku roket saat roket terbang maka diperlukan suatu media komunikasi pengiriman data jarak jauh yaitu modul radio.

2.3.3.1 Komponen Komunikasi Data

Komunikasi data memiliki tujuan untuk melakukan pertukaran informasi diantara beberapa perangkat yang saling berhubungan seperti yang ditunjukan dalam gambar II-28.



Gambar II-28 Komunikasi Data

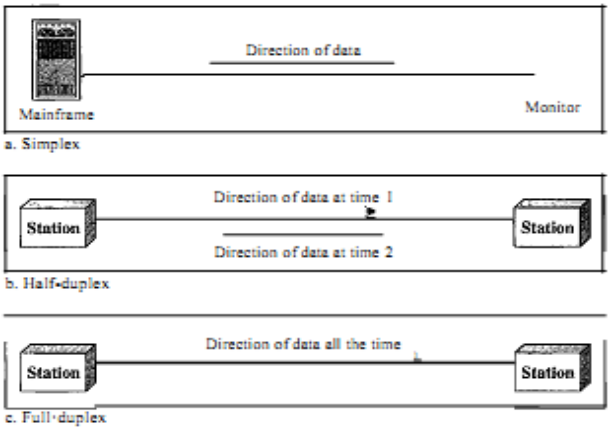
Beberapa hal yang akan menjadi komponen dalam melakukan komunikasi data:

1. *Message*. *Message* atau pesan merupakan informasi yang ditransmisikan.
2. *Sender*. *Sender* atau pengirim merupakan suatu perangkat yang memiliki tugas untuk mengirimkan *Message.*
3. *Receiver*. *Receiver* atau penerima merupakan suatu perangkat yang memiliki tugas untuk menerima *Message.*
4. *Transmission medium*. *Transmission Medium* atau media transmisi merupakan media fisik yang menjadi tempat *Message* untuk berpindah dari *Sender* ke *Receiver*.
5. *Protocol*. *Protocol* atau protokol merupakan suatu set aturan yang mengatur jalannya komunikasi data.

Dengan memenuhi kriteria tersebut, kita dapat melakukan suatu komunikasi data yang akan menjadi sistem vital pada sistem pemantauan perilaku roket.

2.3.3.2 *Data Flow* atau Aliran Data

Komunikasi antar perangkat memiliki arah aliran data yang menjadi suatu penentu bagaimana dan kemana data akan ditransmisikan. Berikut ini merupakan beberapa ilustrasi dari pengiriman aliran data.



Gambar II-29 Aliran data

Berikut ini merupakan penjelasan dari ilustrasi Gambar II-29

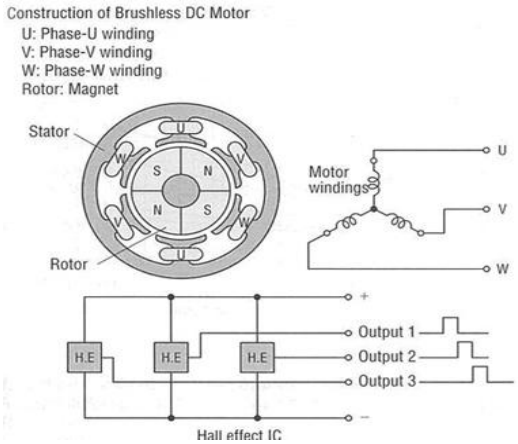
a. Simplex merupakan metode pengiriman aliran data dimana setiap pihak hanya memiliki 1 peran, dimana pihak satu akan menjadi pengirim dan pihak lainnya adalah penerima.
b. Half-Duplex merupakan metode pengiriman aliran data dimana setiap pihak mampu memiliki peran ganda sebagai pengirim atau penerima tapi tidak disaat bersamaan.
c. Full-Duplex merupakan metode pengiriman aliran data dimana setiap pihak mampu menjadi pengirim dan penerima secara bersamaan.

2.3.4 Aktuator

2.3.4.1 Motor Brushless

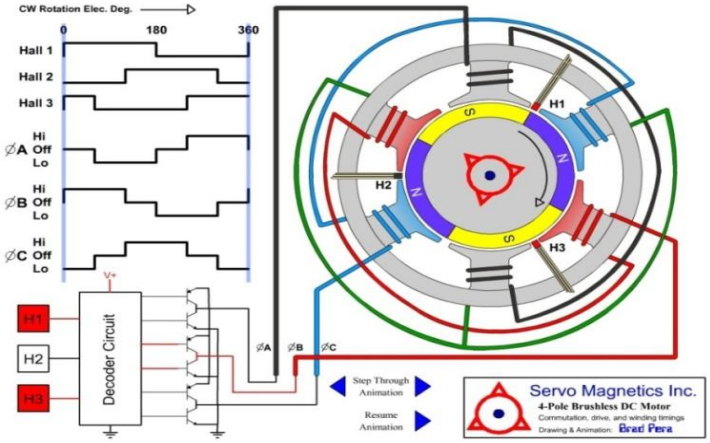
Sesuai dengan pokok bahasan yang telah dibuat, mesin pendorong roket akan menggunakan motor elektrik. Motor *Brushless* dikenal juga dengan *electronically commutated motors* (ECMs) adalah motor yang tersinkronisasi yang ditenagai oleh tegangan DC. Perbedaan motor DC *Brush* dan *Brushless* terlihat pada penggunaan *brush* atau lempengan penghubung untuk mengalirkan tegangan DC pada motor brush yang tidak dimiliki motor brushless. *Brushless* motor juga kadangkala dianggap sebagai motor stepper karena bentuk fisiknya yang sama. Namun motor stepper didesain untuk lebih ditekankan pada

perpindahan sudut putaran sedangkan motor brushless pada topik tugas akhiri ini digunakan untuk kecepatan perubahan putaran.



Gambar II-30 Rangkaian fisik pada Motor Brushless

Berdasarkan gambar II-30, motor brushless memiliki dua bagian utama yakni stator berupa logam yang dililitkan kabel dan rotor berupa magnet berkutub. Stator pada motor *brushless* merupakan bagian yang bertugas memanipulasi kemagnetan yang akan memutarkan rotor. pada motor Motor tipe ini memiliki 3 bidirectional output yang artinya dikontrol dengan 3 fasa. Berikut ini ilustrasi tentang cara kerja dari motor brushless.



Gambar II-31 Cara kerja motor brushless

Berdasarkan gambar II-31, pemanfaatan 3 fasa terdapat pada proses perubahan arah pergerakan kemagnetan pada stator. Disaat motor berputar pada sudut 0 derajat maka arus mengalir dari jalur B ke C. Disaat motor berputar pada sudut 60 derajat maka arus mengalir dari jalur B ke A. Disaat motor berputar

pada sudut 120 derajat maka arus mengalir dari jalur C ke A. Berikut ini tabel II-4 yang merupakan pergerakan arah arus yang terjadi untuk gambar II-31.

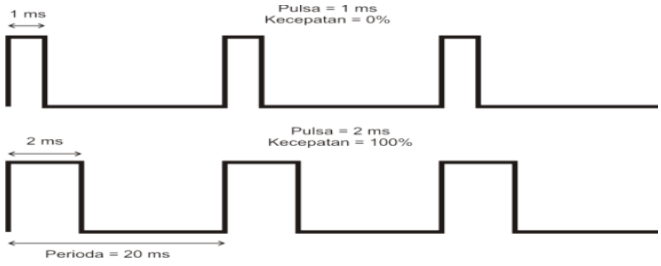
Tabel II-4 Sudut pergerakan motor brushless

No	Sudut Motor (Derajat)	Jalur sumber arus	Jalur tujuan arus
1	0	B	C
2	60	B	A
3	120	C	A
4	180	C	B
5	240	A	B
6	300	A	C

Proses kontrol pada motor ini biasanya dikontrol oleh suatu IC *comparator* untuk mengatur fasa yang dijalankan. Namun untuk pengontrolan pada akselerasi, kecepatan dan efisiensi biasanya digunakan suatu mikrokontroler.

2.3.4.2 Electronic Speed Control(ESC)

Electronic Speed Control(ESC) adalah sirkuit elektronik yang bertujuan untuk mengatur perubahan kecepatan motor, arah putaran motor bahkan dapat sebagai pengerem. Penggunaan motor *Brushless* sebagai motor pendorong mengharuskan penggunaan *driver* pengontrol kecepatan motor berupa ESC. ESC mengandalkan kecepatan *switch* pada transistor untuk mengatur kecepatan motor *brushless*. Jenis transistor yang biasa digunakan pada ESC berupa *field effect transistor*(FET). Sinyal masukan pada ESC pada umumnya berupa PWM 50 Hz dengan varias pulsa 1 milisekon hingga 2 milisekon.



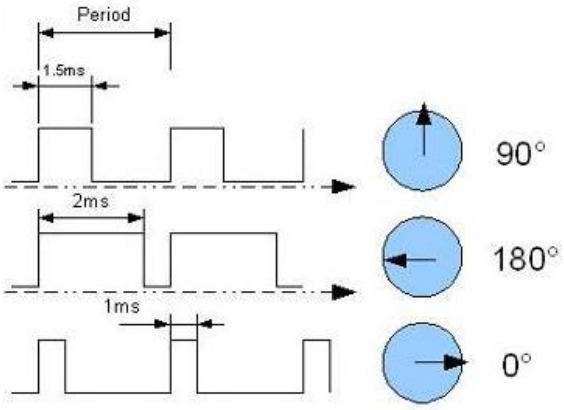
Gambar II-32 Lebar pulsa sinyal masukan ESC

Pada gambar II-32 terlihat bahwa perubahan lebar pulsa yang diberikan kepada perangkat ESC akan menyebabkan perubahan kecepatan pada motor brushless. Berikut ini persamaan yang dihasilkan dari gambar II-32.

$$output = \frac{lebar\ pulsa - 1ms}{1ms} x100\%$$

2.3.4.3 Motor Servo

Motor servo merupakan suatu *rotary actuator* yang berarti suatu aktuator yang menghasilkan pergerakan sudut. Motor servo terdiri dari beberapa bagian rangkaian sederhana yakni motor DC, komparator, *Analog to Digital Converter* (ADC) dan Potensiometer. Cara kerja dari motor servo bergantung pada nilai yang dihasilkan oleh komponen potensiometer yang akan diubah oleh komponen ADC menjadi data digital. Data tersebut akan dibandingkan dengan sinyal masukan oleh komparator. Hasil perbandingan komparator akan menjadi penentu arah pergerakan motor. Sinyal masukan pada motor servo hampir berkerja sama seperti pada *Electronic Speed Control(ESC)*. Hanya saja, pada ESC terjadi perubahan kecepatan motor sedangkan pada motor servo terjadi perubahan sudut seperti pada Gambar II-33 ini.



Gambar II-33 Lebar pulsa motor servo