



UNIVERSITAS INDONESIA

**RANCANG BANGUN PROTOTIPE SISTEM AKTUATOR
SIRIP ROKET KENDALI MENGGUNAKAN *BRUSHED DC*
*MOTOR DAN PLANETARY GEAR***

SKRIPSI

ARIEF KURNIAWAN

0906602446

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
DEPOK
JANUARI 2012**



UNIVERSITAS INDONESIA

**RANCANG BANGUN PROTOTIPE SISTEM AKTUATOR
SIRIP ROKET KENDALI MENGGUNAKAN *BRUSHED DC*
MOTOR DAN *PLANETARY GEAR***

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik

ARIEF KURNIAWAN

0906602446

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
DEPOK
JANUARI 2012**

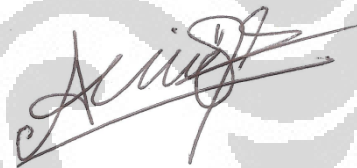
HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

Nama : Arief Kurniawan

NPM : 0906602446

Tanda tangan :



Tanggal : 17 Januari 2012

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh

Nama : Arief Kurniawan
NPM : 0906602446
Program studi : Teknik Elektro
Judul skripsi : Rancang bangun prototipe sistem aktuator sirip roket kendali menggunakan *brushed DC motor* dan *planetary gear*

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia.

DEWAN PENGUJI

Pembimbing : Dr. Abdul Halim M.Eng ()

Penguji : Dr. Abdul Muis ST, M.Eng ()

Penguji : Ir. Aries Subiantoro M.SEE ()

Ditetapkan di : Depok

Tanggal : 17 Januari 2012

KATA PENGANTAR

Puji syukur saya panjatkan kehadirat ALLAH SWT karena atas keridhoan dan karunia-Nya, saya dapat menyelesaikan skripsi ini yang berjudul **”Rancang bangun prototipe sistem aktuator sirip roket kendali menggunakan *brushed DC motor* dan *planetary gear*”** dengan baik.

Penyusunan skripsi ini dibuat untuk memenuhi salah satu persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia.

Skripsi ini dapat tersusun dengan baik dengan dukungan dan bimbingan dari berbagai pihak, maka tidak lupa saya sampaikan terima kasih yang sebesar-besarnya kepada:

1. Bapak Prof. Dr. Ir. Bambang Sugiarto, M.Eng selaku dekan Fakultas Teknik Universitas Indonesia.
2. Bapak Dr. Ir. Dedi Priadi, DEA selaku wakil dekan Fakultas Teknik Universitas Indonesia.
3. Bapak Dr. Ir. Muhamad Asvial, M.Eng selaku Kepala Departemen Teknik Elektro.
4. Bapak Dr. Abdul Halim, M.Eng selaku dosen pembimbing saya yang cukup sabar dalam memberikan arahan dan dukungan pada penyusunan skripsi ini. Terima kasih banyak ya pak....
5. Komaidi sebagai ayah dan Sutarminah sebagai Ibunda tercinta, serta Rahmat Tresnanto selaku adik. Mereka yang senantiasa membantu dan memberikan dukungan dalam berbagai hal hingga selesainya skripsi ini.
6. Anwar Shiddiq, Ariel Yagusandri, dan Ramlan mereka adalah teman-teman satu tim yang sudah memberikan bantuan dalam penyusunan skripsi ini hingga selesai.
7. Teman-teman seperjuangan angkatan 2009 khususnya ekstensi teknik elektro yang telah memberikan semangat dan dukungannya dalam pembuatan skripsi ini.
8. Sahabat-sahabat di luar kampus yang menyediakan waktu dan tenaga untuk membantu dalam penyusunan skripsi ini.

9. Serta pihak lain yang mungkin tidak sempat penulis uraikan satu persatu tanpa mengurangi rasa terima kasih penulis.

Sebuah pepatah mengatakan bahwa *"Tak ada gading yang tak retak"*. Pepatah ini pun sesuai dengan penyusunan skripsi yang telah dilakukan. Kelebihan ataupun kekurangan yang ada di sana-sini menjadi kekuatan yang memotivasi saya supaya lebih baik untuk selanjutnya.

Akhir kata, semoga ALLAH SWT berkenan membalas semua kebaikan pihak-pihak yang membantu selesainya penyusunan skripsi ini. Semoga skripsi ini bermanfaat bagi pengembangan ilmu pengetahuan.

Depok, 17 Januari 2012

Tanda tangan,

Penulis

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS
AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai civitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Arief Kurniawan
NPM : 0906602446
Program Studi : Teknik Elektro
Departemen : Teknik Elektro
Fakultas : Teknik
Jenis karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalty Non Eksklusif (Non-exclusive royalty free right)** atas karya ilmiah saya yang berjudul:

"Rancang bangun prototipe sistem aktuator sirip roket kendali menggunakan *brushed DC motor* dan *planetary gear*"

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non Eksklusif ini Universitas Indonesia berhak menyimpan, mengalih media/formatan, mengelola dalam bentuk pangkalan data (database), merawat, dan mempublikasikan tugas akhir saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : 17 Januari 2012

Yang menyatakan



Arief Kurniawan

ABSTRAK

Nama : Arief Kurniawan

Program studi : Teknik Elektro

Judul : Rancang bangun prototipe sistem aktuator sirip roket kendali menggunakan *brushed DC motor* dan *planetary gear*.

Pada saat ini perkembangan teknologi roket di Indonesia telah memasuki tahap yang pesat. Salah satu tema yang penting pada perkembangan roket adalah sistem pengendalian pada aktuator roket. Sistem pengendalian pada roket sudah memasuki tahap kendali aktif, dimana pergerakan roket dapat diatur saat roket sedang terbang. Dalam skripsi ini akan dirancang prototipe sistem aktuator sirip roket kendali yang diwujudkan dengan menggabungkan beberapa sistem yaitu mikrokontroler ATmega16 sebagai unit pemroses, driver motor DC, *brushed DC motor*, *planetary gear*, *rotation sensor*, komunikasi serial, dan *power supply*. Perancangan perangkat lunak pengendali PID sebagai pengendali program pada mikrokontroler ATmega16 menggunakan bahasa basic dan software AvrOsp sebagai *compiler*-nya. Tujuan pembuatan prototipe ini adalah agar dapat merancang perangkat keras, perangkat lunak dan mengetahui kinerja sistem kendali PID untuk pengendalian putaran motor DC yang mengatur pergerakan dari sudut putaran sirip sehingga menentukan arah dari tujuan roket. Untuk menentukan koefisien-koefisien pengendali PID, digunakan metode penalaan Ziegler-Nichols. Dari hasil pengujian, dapat disimpulkan bahwa dengan menggunakan pengontrol PID dengan nilai $K_p=11$ $K_i=60,5$ dan $K_d=0,50$ didapatkan bahwa tanggapan sistem dapat mencapai kestabilan dan tidak mengalami lonjakan yang berarti, artinya kestabilan dan performansi (kinerja sistem) sesuai yang diinginkan. Dari hasil pengujian didapatkan settling time sebesar 1,05 detik, overshoot tereduksi lebih kecil dari 12%, dan kesalahan keadaan tunak mendekati nol.

Kata kunci:

Brushed DC Motor, *planetary gear*, kendali PID, mikrokontroler ATmega16, sirip roket kendali.

ABSTRACT

Name : Arief Kurniawan

Study Program: Electrical Engineering

Title : Prototype development of guided missile fin actuator system using brushed DC motor and planetary gear.

In this time development of rocket technology has been growing rapidly. One important theme in development of the rocket is how to make control system on rocket actuator. Control system on rocket has entered the stage of active control, which movement of rocket can be set when rocket in flight. This final project will develop a prototype of guided missile fin actuator system which is realized by combining several sub-systems such as microcontroller ATmega16 as a processing unit, DC motor driver, brushed DC motor, planetary gear, rotation sensor, serial communication, and power supply. Basic language is used to program microcontroller ATmega16 and AvrOsp as compiler. The purpose of this final project is to be able to design hardware, software and know the performance of PID control system for controlling DC motor rotation. The rotation of DC motor then regulates the movement of fin so the rocket can be directed to desired destination. To determine the parameters of PID controllers, Ziegler-Nichols tuning method is utilized. By using parameters of PID $K_p=11$ $K_i=60.5$ and $K_d=0.50$ the system response has shown good stability and performance. Its mean that the design has achieved the desired performance. From test results, the system has 1.05 second settling time, 12% overshoot and zero steady-state error.

Key words:

Brushed DC motor, planetary gear, PID control, microcontroller ATmega16, guided missile fin.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN	iii
KATA PENGANTAR	iv
LEMBAR PERSETUJUAN PUBLIKASI UNTUK AKADEMIS	vi
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI	ix
DAFTAR TABEL	xi
DAFTAR GAMBAR	xii
BAB 1 PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Perumusan Masalah.....	2
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian.....	3
1.5 Batasan Penelitian	4
1.6 Metode Penelitian.....	4
1.7 Sistematika Penulisan.....	5
BAB 2 TINJAUAN PUSTAKA	
2.1 Sistem sirip roket	7
2.2 Motor DC sikat magnet permanen.....	8
2.2.1 Bagian-bagian <i>brushed</i> DC motor.....	9
2.2.2 Karakteristik <i>brushed</i> DC motor	10
2.2.3 Motor DC <i>Driver</i>	12
2.3 <i>Rotation</i> Sensor	14
2.4 <i>Gear</i>	16
2.4.1 Klasifikasi rodagigi.....	16
2.4.1.1 Menurut letak poros.....	17
2.4.1.2 Menurut arah putaran	17
2.4.1.3 Menurut bentuk jalur gigi	18
2.4.2 Perbandingan putaran dan perbandingan rodagigi	20
2.4.3 Penggunaan rodagigi kerucut (<i>Bevel</i>) dan <i>planetary gear</i>	20
2.4.3.1 Penggunaan <i>bevel gear</i> (kerucut).....	20
2.4.3.2 Penggunaan <i>planetary gear</i>	21
2.5 Mikrokontroler AVR ATmega16.....	22
2.5.1 Arsitektur ATmega16	23
2.5.2 Fitur ATmega16	24
2.5.3 Komunikasi serial mikrokontroler ATmega16.....	25
2.5.4 Analog to Digital Converter (ADC)	26
2.6 Pengendali PID	27
2.6.1 Pengendali Proporsional.....	30
2.6.2 Pengendali Integral	31
2.6.3 Pengendali Derivatif	32
2.6.4 Pengendali Proporsional, Integral, Derivatif.....	34

BAB 3 PERANCANGAN SISTEM

3.1 Deskripsi sistem	36
3.2 Perancangan perangkat keras	37
3.2.1 Perancangan mikrokontroler	37
3.2.2 Perancangan sirip dan <i>pitch gauge</i>	39
3.2.3 Perancangan driver motor DC	40
3.2.4 Motor DC yang digunakan	41
3.2.5 <i>Gear</i> yang digunakan	42
3.2.6 <i>Rotation sensor</i>	43
3.2.7 Perangkat keras pendukung	44
3.3 Rangkaian perangkat keras keseluruhan	44
3.4 Perancangan kendali PID	46
3.5 Perancangan perangkat lunak	50

BAB 4 PENGUJIAN DAN ANALISIS HASIL

4.1 Penggunaan alat ukur pada pengujian	53
4.2 Pengujian driver motor DC L298	54
4.3 Pengujian <i>rotation sensor</i> sebagai <i>feedback</i>	56
4.4 Pengujian sistem dengan input <i>duty cycle</i>	59
4.5 Pengujian sistem <i>close loop</i>	60
4.5.1 Penalaan pengendali PID dengan metode Ziegler Nichols	61
4.5.2 Pengujian pengendali PID pada sirip	62

BAB 5 PENUTUP

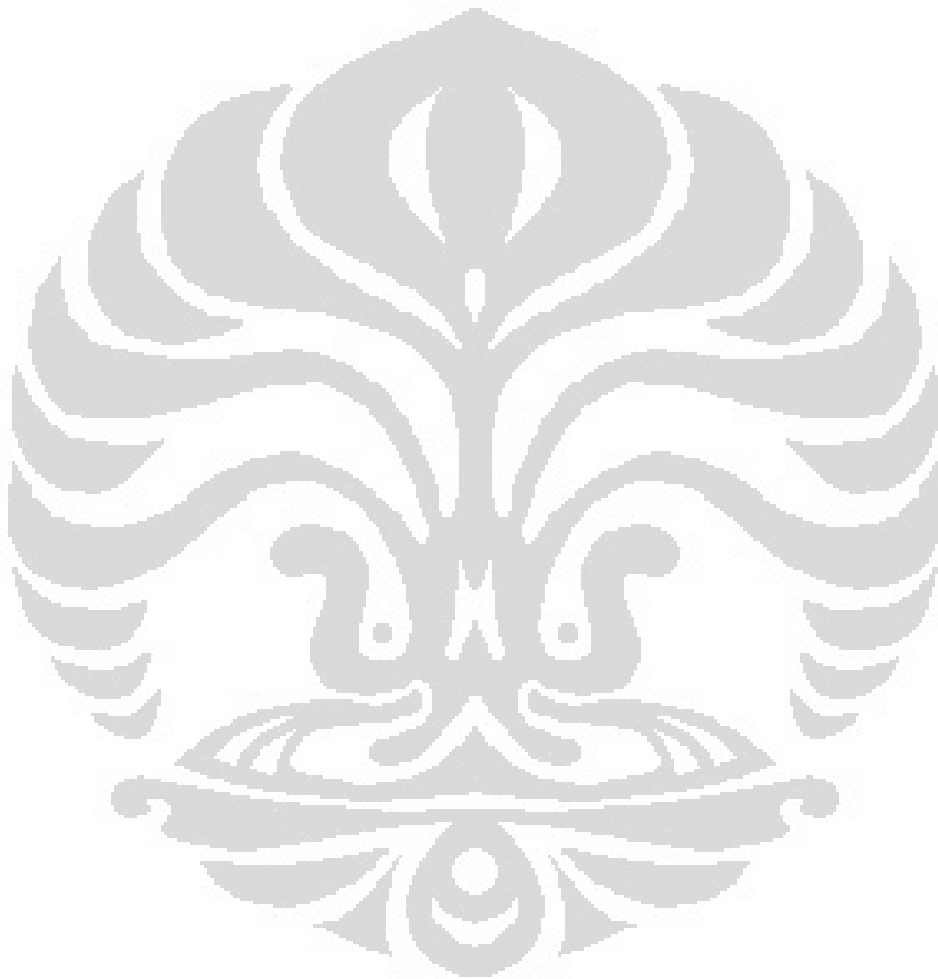
5.1 Kesimpulan	64
5.2 Saran	65

DAFTAR REFERENSI	66
-------------------------------	----

LAMPIRAN	67
-----------------------	----

DAFTAR TABEL

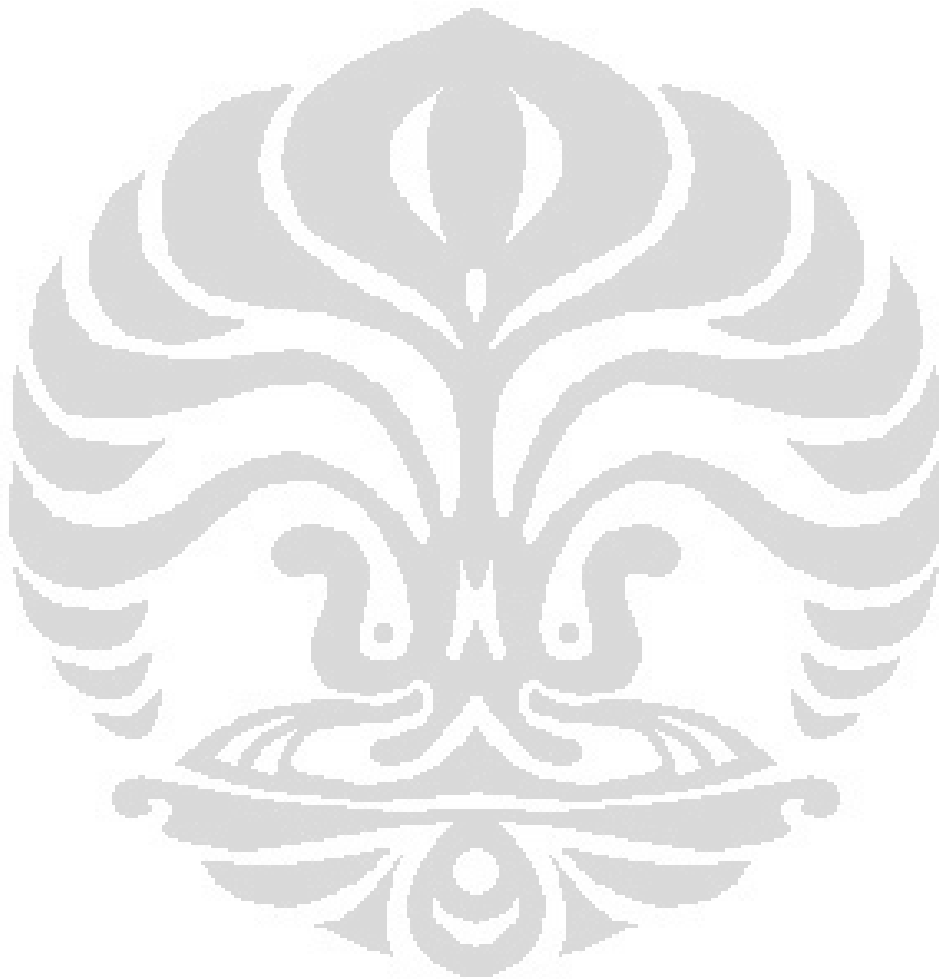
Tabel 2.1	Rodagigi menurut letak poros	17
Tabel 2.2	Respon pengendali PID terhadap perubahan konstanta.....	28
Tabel 3.1	Data <i>brushed</i> DC motor produksi Maxon.....	41
Tabel 3.2	Data <i>planetary gear</i> produksi Maxon.....	43
Tabel 4.1	Hasil pengujian rangkaian driver motor DC L298.....	55
Tabel 4.2	Data ADC pada setiap perubahan sudut	58



DAFTAR GAMBAR

Gambar 2.1	Konfigurasi <i>grid</i> sirip pada roket.....	7
Gambar 2.2	Bagian-bagian motor DC secara garis besar.....	9
Gambar 2.3	Medan magnet yang membawa arus mengelilingi konduktor....	10
Gambar 2.4	Prinsip kerja motor DC.....	11
Gambar 2.5	Blok diagram L298	13
Gambar 2.6	Aplikasi L298 pengendali arah putaran motor DC.....	13
Gambar 2.7	Konfigurasi paralel untuk arus yang lebih besar	14
Gambar 2.8	Penggunaan potensiometer sebagai sensor putaran	15
Gambar 2.9	Potensiometer sensor putaran	16
Gambar 2.10	Rodagigi lurus.....	18
Gambar 2.11	Rodagigi miring	18
Gambar 2.12	Rodagigi kerucut.....	19
Gambar 2.13	Rodagigi cacing	20
Gambar 2.14	Rodagigi kerucut.....	21
Gambar 2.15	<i>Planetary gear</i> buatan MAXON.....	21
Gambar 2.16	Bagian <i>planetary gear</i>	22
Gambar 2.17	Prinsip putaran <i>planetary gear</i>	22
Gambar 2.18	Blok diagram arsitektur mikrokontroler ATmega16.....	23
Gambar 2.19	Rangkaian komunikasi serial RS232 dengan Rx, Tx.....	26
Gambar 2.20	Blok diagram dari kontroler PID	28
Gambar 2.21	Kurva tanggapan transien.....	29
Gambar 2.22	Blok diagram kontrol proporsional	30
Gambar 2.23	Blok diagram pengendali integral.....	32
Gambar 2.24	Blok diagram pengendali derivatif.....	33
Gambar 2.25	Blok diagram kontrol PID	34
Gambar 3.1	Blok diagram sistem.....	36
Gambar 3.2	Modul minimum sistem AVR ATmega16	38
Gambar 3.3	Rangkaian minimum sistem AVR ATmega16.....	38
Gambar 3.4	Bentuk sirip pada prototipe.....	39
Gambar 3.5	<i>Pitch gauge</i> dengan besaran sudut -10° sampai $+10^{\circ}$	40
Gambar 3.6	Rangkaian driver motor DC menggunakan IC L298N	40
Gambar 3.7	Perangkat keras driver motor DC	41
Gambar 3.8	Maxon <i>brushed</i> DC motor.....	42
Gambar 3.9	<i>Gear</i> jenis <i>planetary gear</i>	42
Gambar 3.10	<i>Rotation sensor</i>	43
Gambar 3.11	Skematik rangkaian <i>rotation sensor</i>	44
Gambar 3.12	Rangkaian perangkat keras keseluruhan	45
Gambar 3.13	Blok diagram PID	46
Gambar 3.14	Diagram alir kontrol PID digital	49
Gambar 3.15	Diagram alir perangkat lunak	51
Gambar 4.1	Multimeter digital DT-830B.....	53
Gambar 4.2	Tampilan aplikasi <i>hyperterminal</i>	54
Gambar 4.3	PC <i>Oscilloscope</i> DSO-2090 USB.....	54
Gambar 4.4	Blok diagram pengujian motor DC.....	55

Gambar 4.5	Blok diagram pengujian <i>rotation sensor</i>	57
Gambar 4.6	Grafik pergerakan sirip dengan input <i>duty cycle</i>	59
Gambar 4.7	Grafik pergerakan sirip tanpa pengendali PID	61
Gambar 4.8	Grafik pergerakan sirip dengan pengendali PID.....	63



BAB 1

PENDAHULUAN

Pada bab ini berisi latar belakang masalah, perumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, metodologi penyelesaian masalah, serta sistematika penulisan yang digunakan dalam pembuatan tugas akhir ini.

1.1 Latar Belakang

Penguasaan teknologi roket di dunia kedirgantaraan dan antariksa saat ini sangatlah penting karena roket dibutuhkan untuk memenuhi berbagai macam keperluan suatu bangsa atau negara. Indonesia sebagai negara yang mempunyai batas wilayah yang sangat luas haruslah menguasai teknologi roket tersebut guna menunjang kemandirian bangsa.

Penggunaan aktuasi elektromekanis semakin berkembang pada industri kedirgantaraan saat ini. Banyak di negara maju maupun berkembang telah menggunakan sistem aktuasi elektromekanis pada sistem penerbangannya terutama pada teknologi roket. Pemahaman dan pengaplikasian yang baik dari karakteristik aktuator sangatlah penting karena menunjang kesuksesan aplikasi pada sistem kedirgantaraan di beberapa negara. Sebelum akhirnya aktuator elektromekanis diterapkan secara luas pada teknologi kedirgantaraan, sebelumnya telah dilakukan penelitian, pengujian, dan pengembangan yang luas pada aktuasi penerbangan.

Sistem kontrol digital menjadi sangat penting dalam suatu proses kendali karena berbagai keuntungan yang ditawarkan. Akan tetapi pengontrol digital juga memiliki beberapa kelemahan, untuk itu didesain suatu kontrol sistem yang memiliki kemampuan mengidentifikasi sekaligus mengoreksi kesalahan output sistem dengan menggunakan kontrol PID (*Proportional*, *Integral*, dan *Derivative*). Desain sistem pengendali dan analisis teknologi yang diterapkan secara luas sangat berguna dalam pengembangan aplikasi. Beberapa aplikasi dapat diselesaikan dengan perangkat keras dan oleh kemajuan teknologi dapat juga

ditunjang menggunakan perangkat lunak, sehingga sistem kontrol dapat dianalisis dengan mudah dan detail.

Pada skripsi ini akan diperkenalkan perancangan dan pembuatan prototipe sistem aktuator sirip roket kendali menggunakan *brushed DC motor* dan *planetary gear* serta analisis dan implementasi kontrol posisi sudut pada aktuator sirip roket kendali dengan menggunakan metode pengendali PID. Pada prototipe aktuator ini menggunakan sebuah sirip buatan yang nantinya akan digunakan untuk mengatur pergerakan dari sudut sirip sehingga menentukan arah dari tujuan roket. Sirip ini dikendalikan dengan menggunakan ATMEGA16 dengan menggunakan bahasa pemrograman basic AVR. Mikrokontroler ATMEGA16 akan menerima data serial dari pusat kendali berupa besaran sudut gerak yang akan menggerakkan sirip aktuator, yang mana pergerakan dari sirip ini akan mengubah arah dari roket. Sebagai penggerak dari sirip tersebut digunakan motor DC jenis sikat magnet permanen (*Brushed DC Motor*). Motor DC sering digunakan dalam sistem aktuasi permukaan *aerodynamic* untuk menggerakkan sirip roket. Untuk mereduksi kecepatan motor dan menghasilkan torsi yang sesuai digunakan *gear*, sehingga dapat menahan gaya dorongan angin yang terjadi. Motor DC dikendalikan menggunakan sistem kontrol *close loop* PID untuk menghasilkan respon sistem yang baik. Paramater-parameter pada kendali PID akan ditentukan dengan metode *trial and error*, kemudian akan dibandingkan dengan mencari nilai dari parameter PID secara teori. Penentuan nilai parameter yang tepat akan menghasilkan respon sistem yang baik.

Perancangan dan pembuatan model prototipe sistem aktuator sirip roket kendali menggunakan *brushed DC motor* dan *planetary gear* memperhitungkan ketidaklinieran karena keterbatasan pergerakan mekanik dari sirip, torsi motor, kecepatan sudut, gesekan pada roda gigi dan bantalan, *backlash* pada gigi dan tuas mekanik, dan lain-lain.

1.2 Perumusan Masalah

Pada perancangan dan pembuatan prototipe sistem aktuator sirip roket kendali menggunakan *brushed DC motor* dan *planetary gear* menggunakan mikrokontroler ATMEGA16 dengan menggunakan bahasa basic AVR (*basic*

compiler AVR). Untuk penggerak sirip digunakan motor DC jenis sikat magnet permanen yang dihubungkan dengan *gear* jenis *spur internal (planetary gearhead)* dengan mereduksi kecepatan motor dan menghasilkan torsi yang cukup kuat untuk menahan dorongan angin yang diakibatkan oleh gaya dorong roket sendiri. Untuk sudut pergerakan sirip diatur oleh mikrokontroler dengan memperoleh data serial dari pusat kendali roket.

Berdasarkan pada permasalahan yang diuraikan di atas, maka permasalahan yang dibahas dalam skripsi ini meliputi:

1. Bagaimana merancang dan membuat prototipe sistem aktuator sirip roket kendali menggunakan *brushed DC motor* dan *planetary gear*.
2. Bagaimana memilih motor DC dan *gear* yang tepat sehingga dapat digunakan untuk menggerakkan sirip aktuator.
3. Bagaimana membuat program menggunakan bahasa pemrograman *basic compiler* AVR pada mikrokontroler ATMEGA16 untuk mengendalikan pergerakan sirip.
4. Bagaimana menentukan nilai parameter kontrol PID (K_p , K_i , dan K_d) yang sesuai sehingga dapat menghasilkan respon sistem yang diinginkan.
5. Bagaimana menerapkan teori kontrol PID untuk aplikasi digital pada pemrograman ATMEGA16.

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah mempelajari, merancang, dan membuat prototipe sistem aktuator sirip roket kendali menggunakan *brushed DC motor* dan *planetary gear* sebagai peningkatan dibidang *aerodynamic* khususnya peningkatan dibidang roket.

1.4 Manfaat Penelitian

Hasil dari penelitian ini diharapkan dapat menjadi landasan untuk pengembangan penelitian berikutnya seperti menemukan dan membuat aktuator sirip roket kendali beserta dengan sistem kontrol yang cocok untuk mengendalikan aktuator sirip roket kendali.

1.5 Batasan Penelitian

Batasan masalah pada penelitian ini difokuskan pada perancangan dan pembuatan prototipe sistem aktuator sirip roket kendali menggunakan *brushed DC motor* dan *planetary gear*, rangkaian sistem minimum mikrokontroler, dan komunikasi serial. Permasalahan dibatasi hanya pada pembuatan perangkat keras dan perangkat lunak yang memberikan data masukan bagi proses pengontrolan aktuator sirip roket kendali. Dengan batasan masalahnya meliputi:

1. Pembuatan perangkat keras prototipe aktuator sirip roket kendali menggunakan motor DC jenis sikat magnet permanen dengan kombinasi *planetary gear*.
2. Penggunaan mikrokontroler Atmel AVR ATMEGA16 sebagai pengendali aktuator sirip roket, yang nanti akan diprogram sehingga dapat menerima data masukan sebagai proses pengontrolan.
3. Menggunakan rangkaian RS232 untuk komunikasi serial.
4. Penggunaan metode pengendali PID (*Proportional, Integral, dan Derivative*) sebagai pengontrolan sistem kendali untuk menentukan gerakan besarnya sudut dari sirip.

1.6 Metodologi Penyelesaian Masalah

Dalam pengerjaan tugas akhir ini, dilakukan dalam beberapa tahap yaitu sebagai berikut: tahap tinjauan pustaka, tahap perancangan, tahap pembuatan prototipe, tahap pengujian sistem dan analisis, dan tahap pembuatan laporan. Penjelasan lebih detail adalah di bawah ini:

1. Tinjauan Pustaka

Untuk memperoleh teori-teori dasar serta teori penunjang yang dibutuhkan dalam pembuatan prototipe dan penulisan skripsi ini, penulis melakukan pencarian dari berbagai sumber baik dari buku-buku, literatur, laporan riset, maupun makalah-makalah yang diberikan oleh dosen pembimbing dan didapat dari perpustakaan, dan internet.

2. Perancangan sistem

Dari data-data yang didapatkan, selanjutnya dilakukan perancangan sistem dan analisis yang meliputi data karakteristik dari motor DC sikat

magnet permanen, *gear*, *rotation* sensor (potensio), *driver* motor, dan mikrokontroler AVR ATMEGA16.

3. Pembuatan prototipe

Pembuatan prototipe sistem aktuator sirip roket kendali dengan menghubungkan motor DC dan *gear* sedemikian hingga menjadi gerak arah derajat dan untuk bagian elektronik dibuat *driver* motor untuk mengendalikan putaran motor serta digunakan *rotary* sensor untuk mengetahui besar derajat arah gerak dari sirip.

4. Pengujian sistem dan analisis

Pengujian dan analisis dimaksudkan untuk mengetahui sejauh mana sistem yang dibuat pada prototipe ini dapat berfungsi sesuai dengan proses sistem yang diharapkan.

5. Pembuatan laporan

Membuat dokumentasi dari semua tahapan proses di atas berupa laporan yang berisi tentang dasar teori, perancangan, hasil pengujian sistem, dan analisis dari proyek tugas akhir ini.

1.7 Sistematika Penulisan

Sistematika penulisan yang akan diurai dalam buku laporan tugas akhir ini terbagi dalam lima bab yang akan dibahas sebagai berikut:

BAB 1 PENDAHULUAN

Berisi latar belakang masalah, perumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, metodologi penyelesaian masalah, serta sistematika penulisan yang digunakan.

BAB 2 TINJAUAN PUSTAKA

Merupakan bagian yang menguraikan teori yang mendukung penyelesaian masalah dan teori yang terkait dengan prototipe sistem aktuator sirip roket kendali menggunakan *brushed DC motor* dan *planetary gear*.

BAB 3 PERANCANGAN SISTEM

Pada bab ini membahas mengenai perancangan dan pembuatan alat serta ide-ide dan gambaran tentang alat yang akan dibuat. Bagian

ini akan menampilkan komponen yang dipergunakan, blok diagram, gambar tiap modul dan penjelasan singkatnya serta perancangan dan fungsi tiap-tiap bagian.

BAB 4 PENGUJIAN DAN ANALISIS HASIL

Pada bab ini menjelaskan seluruh hasil pengujian dan analisis dalam pembuatan proyek tugas akhir. Membahas tentang hasil dari alat yang dibuat, apakah alat tersebut telah bekerja sesuai dengan yang diinginkan.

BAB 5 PENUTUP

Pada bab ini berisi rangkuman keseluruhan dari inti penelitian yang telah dilakukan, serta mengacu pada hasil yang telah diperoleh. Bab ini berisi kesimpulan dan saran dari hasil pengujian.



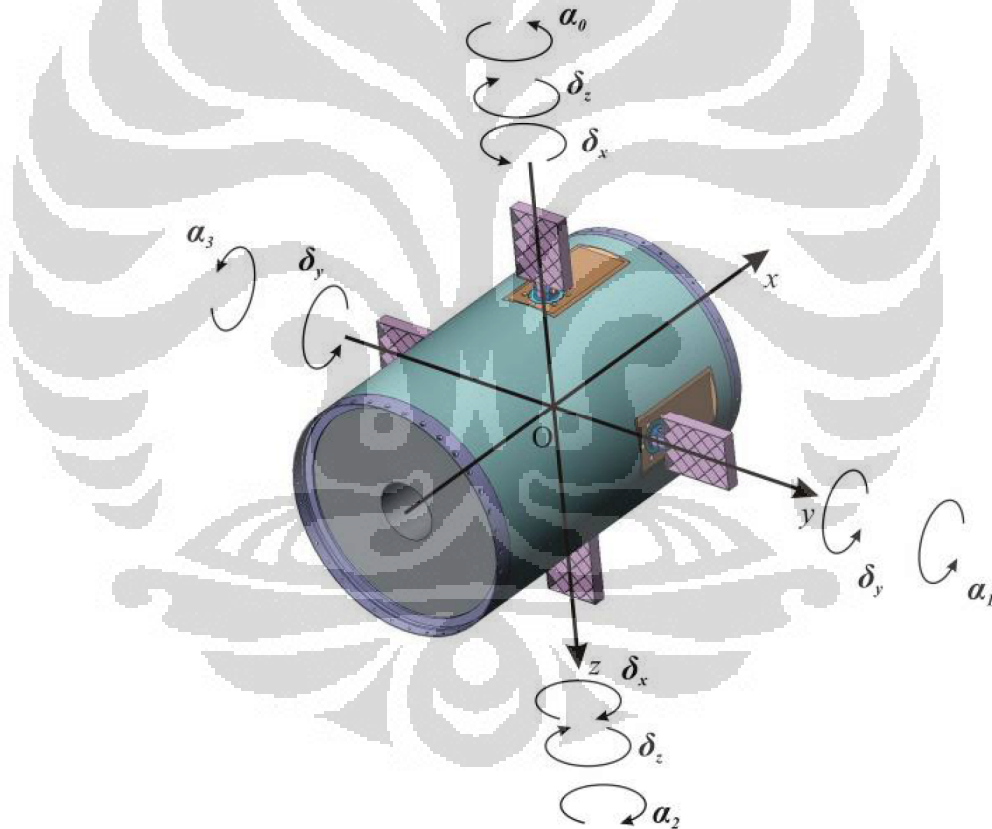
BAB 2

TINJAUAN PUSTAKA

Bab 2 Tinjauan Pustaka akan membahas mengenai teori dan konsep yang terkait. Studi literatur yang dilakukan sebelumnya dan sumber-sumber bacaan yang dijadikan sebagai bahan rujukan dalam pembuatan tugas akhir ini.

2.1 Sistem sirip roket

Sistem kontrol sirip roket di sini adalah kendali dari rudal atau roket menggunakan empat sirip *grid*. Konfigurasi *grid* sirip disajikan dalam Gambar 2.1.



Gambar 2.1 Konfigurasi *grid* sirip pada roket

Dengan menggerakkan *grid* sirip, momen yang dihasilkan akan memutar badan roket tersebut. Perubahan sudut menghasilkan gaya aerodinamika, yang dapat mengarahkan roket ke arah yang diinginkan. Ada 4 sirip yang berada pada roket yang dikendalikan oleh aktuator, dan masing-masing aktuator dikendalikan secara independen akibat penyimpangan pada permukaan yang terjadi,

penyimpangan sirip pada umumnya bergerak kurang dari 10° [1]. Simpangan sirip sebesar 1° mampu membuat perubahan arah roket cukup signifikan, sehingga pergerakan sirip pada roket tidak lebih dari 10° . Pergerakan sirip harus mampu menahan beban angin, gravitasi, dan kecepatan yang dihasilkan oleh pendorong. Kebutuhan torsi untuk menggerakkan sirip dalam kondisi roket sedang terbang pada kecepatan 0,9 Mach (tipe roket kendali RXX 200) adalah 0,236102 Nm. Tekanan dinamis pada sirip berkisar pada 387,947 Pa sampai 78262,18 Pa [14]. Data-data tersebut berguna untuk menentukan jenis motor yang akan dipakai serta sistem transmisinya.

2.2 Motor DC sikat magnet permanen

Motor listrik berguna untuk mengubah energi listrik menjadi energi mekanik. Hal ini dilakukan oleh dua interaksi yaitu medan magnet dan satu stasioner, serta sebuah komponen bagian dalam yang dapat bergerak [2]. Motor DC memiliki kemampuan potensi torsi yang sangat tinggi, mudah untuk dibuat miniatur, dan dapat *men-throttle* melalui penyesuaian tegangan suplainya. Motor DC tidak hanya sederhana, tapi juga motor listrik tertua.

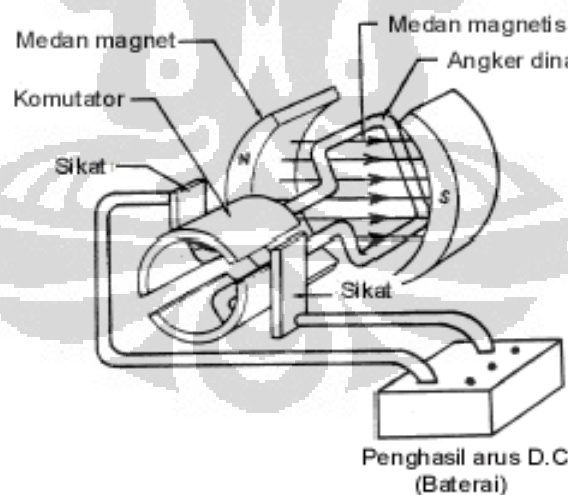
Motor DC adalah motor yang digerakkan oleh energi listrik arus searah. Salah satu jenis motor DC tersebut ialah motor DC sikat magnet permanen (*Permanent magnet brushed DC motor*) yang banyak ditemui penggunaannya baik di industri militer maupun komersil. Terapan motor DC tersebut kebanyakan merupakan sistem yang memerlukan pengaturan baik torsi maupun kecepatan. Kecepatan motor DC dapat diatur dengan menggunakan sebuah pengendali atau pengontrol tertentu yang sesuai dengan karakteristik dari motor DC tersebut agar mencapai suatu kestabilan sistem. Motor DC juga merupakan suatu komponen yang mempunyai kecepatan putaran yang dapat digunakan misalnya untuk memutar *gear*, *impeller* pompa, *fan* atau *blower*, menggerakkan kompresor, mengangkat bahan, dan lain-lain. Kecepatan motor DC itu tidak akan stabil atau bahkan akan merusak suatu alat jika tidak ada pengontrol atau pengendali yang menstabilkan kecepatan tersebut seperti halnya motor DC. Putaran kecepatan, torsi, respon transien, dan sifat-sifat lainnya dari motor DC akan stabil bila

dikontrol oleh sebuah pengontrol baik menggunakan pengontrol PID, pengontrol PWM, *fuzzy logic*, atau metode pengontrol lainnya.

Kestabilan suatu sistem ditentukan oleh tanggapannya terhadap masukan atau gangguan. Berdasarkan definisi kestabilan yang didasarkan atas tanggapan sistem terhadap masukan terbatas bahwa sistem adalah stabil jika setiap masukan terbatas menghasilkan keluaran terbatas, artinya antara masukan acuan dan keluaran yang diinginkan (*set point*) harus terjadi keseimbangan.

2.2.1 Bagian-bagian *brushed* DC motor

Sebuah motor DC magnet permanen biasanya tersusun atas magnet permanen, kumparan jangkar dan sikat (*brush*). Medan magnet yang besarnya konstan dihasilkan oleh magnet permanen, sedangkan komutator dan sikat berfungsi untuk menyalurkan arus listrik dari sumber di luar motor ke dalam kumparan jangkar. Letak sikat di sepanjang sumbu netral dari komutator, yaitu sumbu di mana medan listrik yang dihasilkan bernilai nol. Hal ini dimaksudkan agar pada proses perpindahan dari sikat ke komutator tidak terjadi percikan api. Bagian-bagian motor DC secara garis besarnya dapat dilihat pada Gambar 2.2.

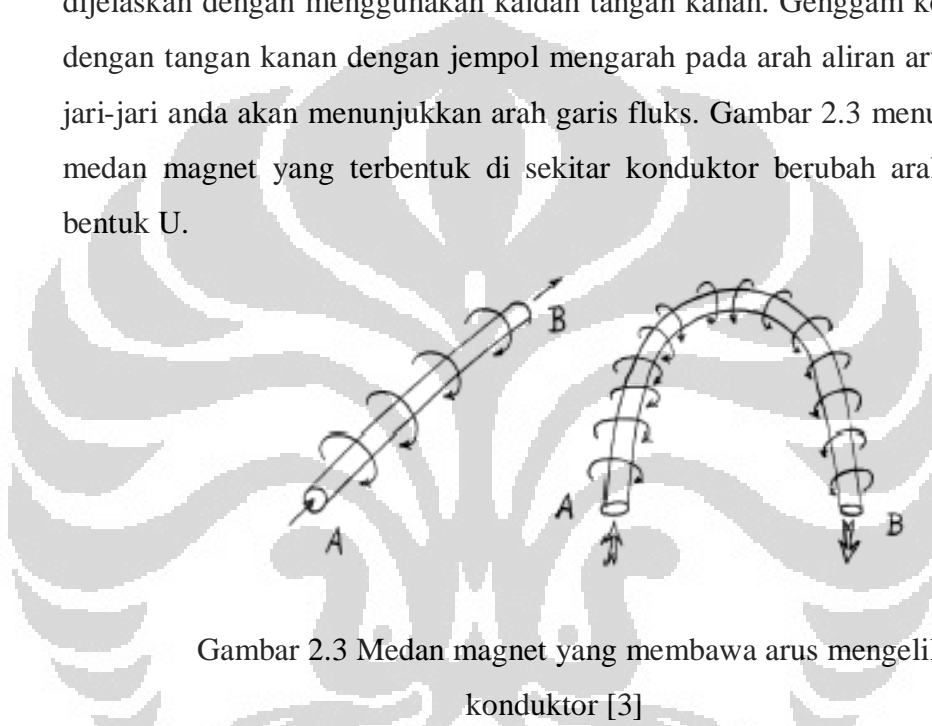


Gambar 2.2 Bagian-bagian motor DC secara garis besar [3]

Catu tegangan DC dari baterai menuju ke lilitan melalui sikat yang menyentuh komutator, dua segmen yang terhubung dengan dua ujung lilitan. Kumparan satu lilitan pada gambar di atas disebut angker dinamo. Angker dinamo adalah sebutan untuk komponen yang berputar di antara medan magnet.

2.2.2 Karakteristik *brushed* DC motor

Gaya gerak magnet, *magnetomotive force* (mmf), selain dihasilkan oleh medan magnet sepanjang sumbu magnet (*magnetic axes*), juga dihasilkan oleh arus yang mengalir di dalam kumparan jangkar. Apabila keduanya bereaksi maka akan timbul torsi yang dapat memutar kumparan jangkar, reaksi ini biasa disebut reaksi jangkar, dan hal ini terjadi secara terus-menerus berkesinambungan. Arah putaran kumparan jangkar dapat dijelaskan dengan menggunakan kaidah tangan kanan. Genggam konduktor dengan tangan kanan dengan jempol mengarah pada arah aliran arus, maka jari-jari anda akan menunjukkan arah garis fluks. Gambar 2.3 menunjukkan medan magnet yang terbentuk di sekitar konduktor berubah arah karena bentuk U.



Gambar 2.3 Medan magnet yang membawa arus mengelilingi konduktor [3]

Perlu diingat bahwa medan magnet hanya terjadi di sekitar sebuah konduktor jika ada arus mengalir pada konduktor tersebut.

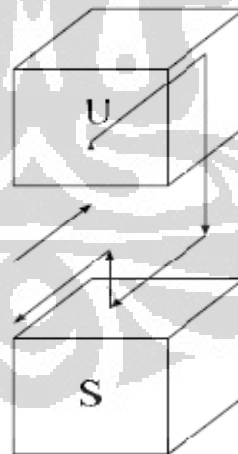
Selama berputar, fluks medan magnet yang dilingkupi oleh kumparan jangkar akan berubah sehingga akan timbul gaya gerak listrik induksi, *electromotive force* (emf), dalam kumparan jangkar yang biasanya disebut *back emf*. Saat kumparan jangkar tersebut mulai berputar, besarnya gaya gerak listrik induksi yang dihasilkan adalah nol sehingga kumparan jangkar akan menarik arus yang besar dari sumber listrik. Hal ini juga menyebabkan mengapa sebuah motor DC mempunyai torsi besar walaupun pada putaran rendah sekalipun.

Mekanisme kerja untuk seluruh jenis motor secara umum :

1. Arus listrik dalam medan magnet akan memberikan gaya.

2. Jika kawat yang membawa arus dibengkokkan menjadi sebuah lingkaran / *loop*, maka kedua sisi *loop*, yaitu pada sudut kanan medan magnet, akan mendapatkan gaya pada arah yang berlawanan.
3. Pasangan gaya menghasilkan tenaga putar / *torque* untuk memutar kumparan.
4. Motor-motor memiliki beberapa *loop* pada dinamanya untuk memberikan tenaga putaran yang lebih seragam dan medan magnetnya dihasilkan oleh susunan elektromagnetik yang disebut kumparan medan.

Pada motor DC, daerah kumparan medan yang dialiri arus listrik akan menghasilkan medan magnet yang melingkupi kumparan jangkar dengan arah tertentu. Konversi dari energi listrik menjadi energi mekanik (motor) maupun sebaliknya berlangsung melalui medan magnet, dengan demikian medan magnet disini selain berfungsi sebagai tempat untuk menyimpan energi, sekaligus sebagai tempat berlangsungnya proses perubahan energi, daerah tersebut dapat dilihat pada gambar 2.4 di bawah ini:



Gambar 2.4 Prinsip kerja motor DC [3]

Agar proses perubahan energi mekanik dapat berlangsung secara sempurna, maka tegangan sumber harus lebih besar dari pada tegangan gerak yang disebabkan reaksi lawan. Dengan memberi arus pada kumparan jangkar yang dilindungi oleh medan maka menimbulkan perputaran pada motor.

Dalam memahami sebuah motor, penting untuk mengerti apa yang dimaksud dengan beban motor. Beban dalam hal ini mengacu kepada keluaran tenaga putar / *torque* sesuai dengan kecepatan yang diperlukan. Beban umumnya dapat dikategorikan ke dalam tiga kelompok:

1. **Beban torque konstan** adalah beban dimana permintaan keluaran energinya bervariasi dengan kecepatan operasinya namun *torquencya* tidak bervariasi. Contoh beban dengan *torque* konstan adalah *corveyors*, *rotary kilns*, dan pompa *displacement* konstan.
2. **Beban dengan variabel torque** adalah beban dengan *torque* yang bervariasi dengan kecepatan operasi. Contoh beban dengan variabel *torque* adalah pompa sentrifugal dan *fan* (*torque* bervariasi sebagai kuadrat kecepatan). Contoh: Peralatan Energi Listrik seperti Motor Listrik.
3. **Beban dengan energi konstan** adalah beban dengan permintaan *torque* yang berubah dan berbanding terbalik dengan kecepatan. Contoh untuk beban dengan daya konstan adalah peralatan-peralatan mesin.

2.2.3 Motor DC Driver

Pengendali posisi sudut dari sirip roket memerlukan sebuah aktuator untuk menggerakkan. Aktuator yang dipakai adalah motor DC. Motor tersebut perlu dikendalikan baik arah putaran dan kecepatan putarannya. Contoh IC driver yang umum digunakan untuk aplikasi pengendalian motor tersebut adalah L298 buatan STMicroelectronics.

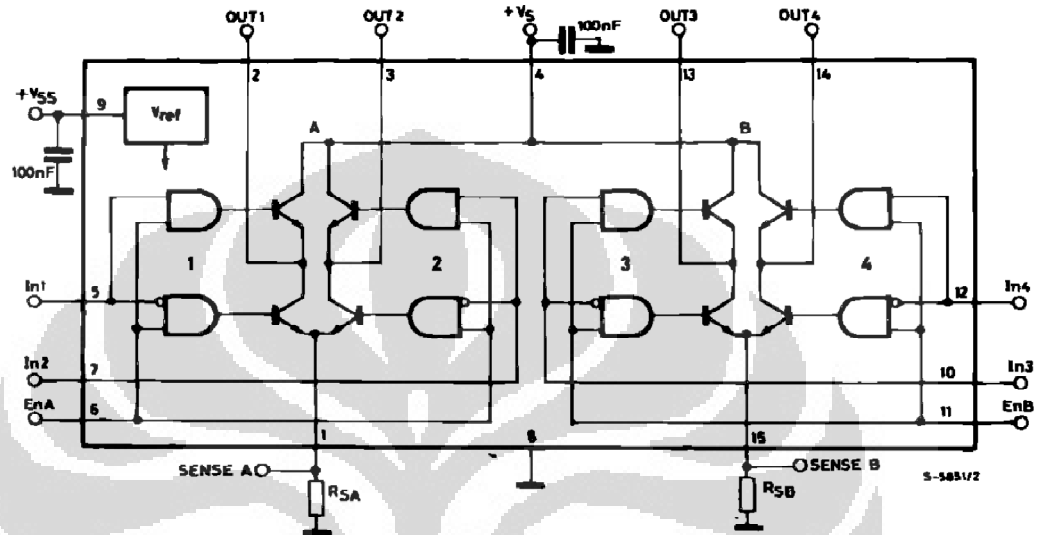
Menurut *datasheet*, L298 adalah *Dual Full bridge driver*. L298 didesain memiliki tegangan penggerak yang tinggi dan arus yang besar dengan menerima input standar *level* logika TTL untuk menggerakkan beban induktif seperti relay, solenoid, DC dan stepping. Berikut adalah spesifikasi singkat dari L298 :

1. Tegangan operasi sampai 46V
2. Total Arus DC sampai 4A
3. Tegangan saturasi yang rendah
4. Proteksi terhadap temperatur berlebih

5. Input tegangan logika “0” sampai 1.5V (*High Noise Immunity*)

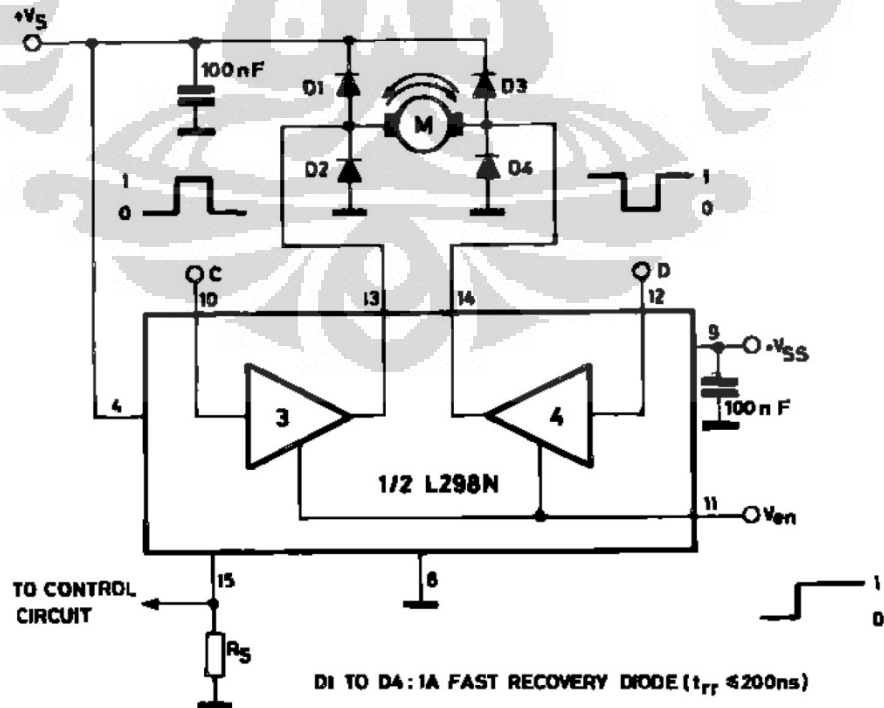
Dua input *enable* disediakan untuk membolehkan atau melarang sinyal input masuk *device* secara independen. Sebuah tambahan input tegangan disediakan agar logika bekerja pada tegangan yang lebih rendah.

Berikut adalah blok diagram dari L298:



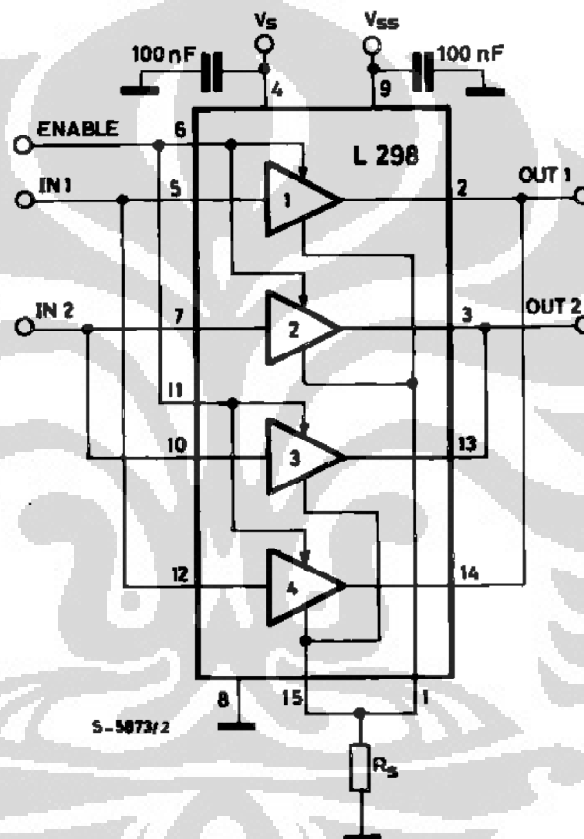
Gambar 2.5 Blok diagram L298 [4]

Berikut adalah diagram skematik aplikasi L298 untuk menggerakkan motor dua arah:



Gambar 2.6 Aplikasi L298 pengendali arah putaran motor DC [4]

Diagram skematik di atas menunjukkan kendali motor DC *bidirectional* yang hanya membutuhkan satu buah *bridge* saja. Eksternal diode D1-D4 dibuat dari empat buah *fast recovery elements*. Fungsi pengereman (*Fast Motor Stop*) membutuhkan bahwa rating absolut maksimal 2 A tidak tercapai. Jika arus repetitif puncak dibutuhkan untuk beban yang melebihi 2 A maka sebuah konfigurasi paralel dibutuhkan. Catatan aplikasi pada *datasheet* L298 mencantumkan konfigurasi tersebut seperti dalam diagram skematik berikut:



Gambar 2.7 Konfigurasi paralel untuk arus yang lebih besar [4]

Konfigurasi di atas dapat menggerakkan beban hingga 3A pada operasi DC dan hingga 3.5A pada arus puncak repetitif.

2.3 Rotation Sensor

Pada skripsi ini sensor putaran menggunakan potensiometer. Pada prinsipnya, potensiometer merupakan suatu resistor variabel yang besar hambatan antara pin keluaran dengan pin *ground* dapat diubah-ubah berdasarkan putaran yang diberikan pada pemutarnya. Jika pada antara pin masukan dan pin *ground*

diberikan beda potensial yang tetap, maka tegangan antara pin keluaran dan *ground* dapat berubah-ubah sesuai dengan sudut putaran yang diberikan pada potensiometer. Tegangan keluaran inilah yang digunakan untuk mengidentifikasi posisi sudut putar pada aktuator yang digunakan. Secara teori, besar perubahan hambatan antara pin keluaran dan pin *ground* sebanding terhadap perubahan sudut putar yang diberikan terhadap potensiometer.

Hubungan tegangan, arus, dan hambatan listrik ditunjukkan pada persamaan 2.1

$$V = I \cdot R \quad (2.1)$$

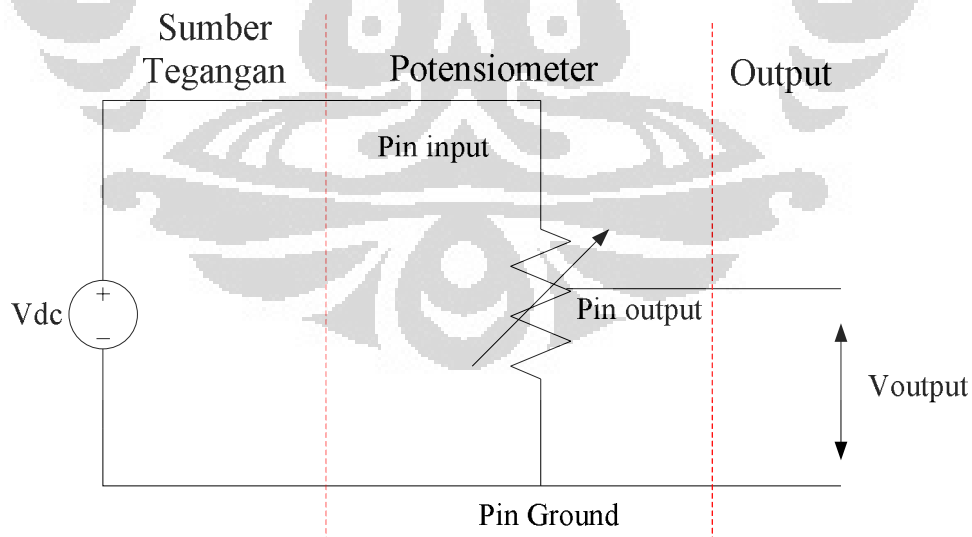
Keterangan:

V = Tegangan

I = Arus

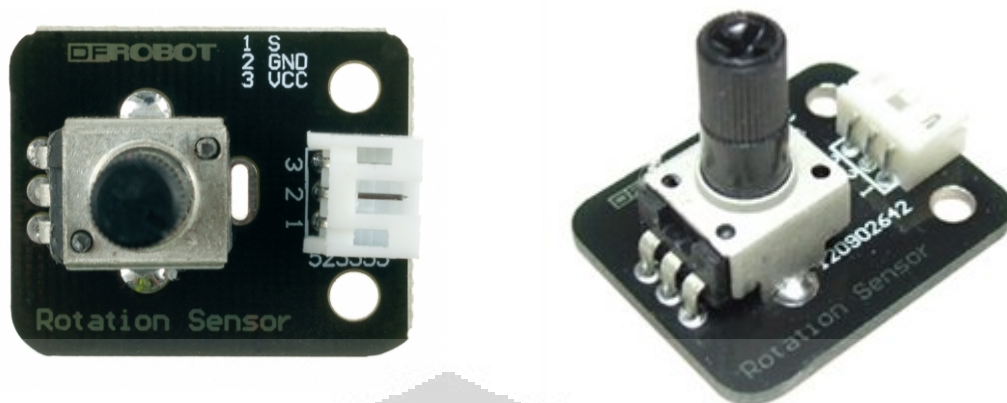
R = Hambatan listrik

Dengan besar arus yang konstan, maka perubahan tegangan akan linier terhadap hambatan listrik, sehingga perubahan sudut putar pada potensiometer akan menyebabkan perubahan tegangan keluaran potensiometer secara linier. Gambar 2.8 menunjukkan ilustrasi penggunaan potensiometer sebagai sensor putaran.



Gambar 2.8 Penggunaan potensiometer sebagai sensor putaran

Pada skripsi ini potensiometer yang digunakan sebagai sensor putaran adalah tipe analog 300 derajat putaran (*single turn*).



Gambar 2.9 Potensiometer sensor putaran [5]

2.4 Gear

Gear atau rodagigi digunakan untuk mentransmisikan daya besar dan putaran yang tepat. Rodagigi memiliki gigi di sekelilingnya, sehingga penerusan daya dilakukan oleh gigi-gigi kedua roda yang saling berkait. Rodagigi sering digunakan karena dapat meneruskan putaran dan daya yang lebih bervariasi dan lebih kompak daripada menggunakan alat transmisi yang lainnya, selain itu rodagigi juga memiliki beberapa kelebihan jika dibandingkan dengan alat transmisi lainnya, yaitu:

1. Sistem transmisinya lebih ringkas, putaran lebih tinggi dan daya yang besar.
2. Sistem yang kompak sehingga konstruksinya sederhana.
3. Kemampuan menerima beban lebih tinggi.
4. Efisiensi pemindahan dayanya tinggi karena faktor terjadinya slip sangat kecil.
5. Kecepatan transmisi rodagigi dapat ditentukan sehingga dapat digunakan dengan pengukuran yang kecil dan daya yang besar.

Rodagigi harus mempunyai perbandingan kecepatan sudut tetap antara dua poros. Disamping itu terdapat pula rodagigi yang perbandingan kecepatan sudutnya dapat bervariasi. Ada pula rodagigi dengan putaran yang terputus-putus.

2.4.1 Klasifikasi rodagigi

Rodagigi diklasifikasikan sebagai berikut:

2.4.1.1 Menurut letak poros

Menurut letak poros maka rodagigi diklasifikasikan seperti tabel berikut:

Tabel 2.1 Rodagigi menurut letak poros [6]

Letak Poros	Rodagigi	Keterangan
Rodagigi dengan poros sejajar	Rodagigi lurus	Klasifikasi atas dasar bentuk alur gigi
	Rodagigi miring	
	Rodagigi miring ganda	
	Rodagigi luar	Arah putaran berlawanan
	Rodagigi dalam dan pinion Batang gigi dan pinion	Arah putaran sama Gerakan lurus dan berputar
Rodagigi dengan poros berpotongan	Rodagigi kerucut lurus	Klasifikasi atas dasar bentuk jalur gigi
	Rodagigi kerucut spiral	
	Rodagigi kerucut zerol	
	Rodagigi kerucut miring	
	Rodagigi kerucut miring ganda	Rodagigi dengan poros berpotongan berbentuk istimewa
Rodagigi permukaan dengan poros berpotongan		
Rodagigi dengan poros silang	Rodagigi miring silang	Kontak gigi
	Batang gigi miring silang	Gerak lurus dan berputar
	Rodagigi cacing silindris	
	Rodagigi cacing selubung ganda	
	Rodagigi cacing samping	
	Rodagigi hiperboloid	
	Rodagigi hipoid	
Rodagigi permukaan silang		

2.4.1.2 Menurut arah putaran

Menurut arah putarannya, rodagigi dapat dibedakan atas:

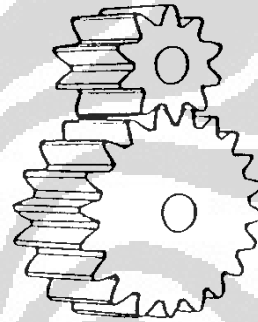
1. Rodagigi luar ; arah putarannya berlawanan.

2. Rodagigi dalam (*Planetary gear*) dan *pinion* ; arah putarannya sama.

2.4.1.3 Menurut bentuk jalur gigi

1. Rodagigi lurus (*spur*)

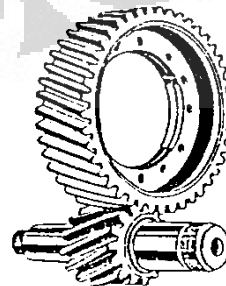
Rodagigi lurus digunakan untuk poros yang sejajar atau paralel. Dibandingkan dengan jenis rodagigi yang lain rodagigi lurus ini paling mudah dalam proses pengerjaannya (*machining*) sehingga harganya lebih murah. Rodagigi lurus ini cocok digunakan pada sistim transmisi yang gaya kelilingnya besar, karena tidak menimbulkan gaya aksial.



Gambar 2.10 Rodagigi lurus [6]

2. Rodagigi miring (*Helical*)

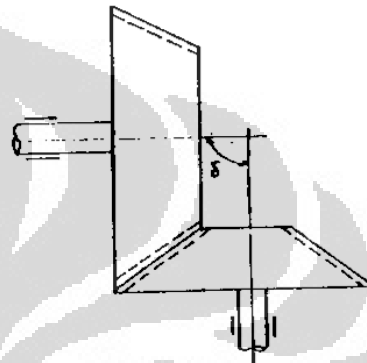
Rodagigi miring kriterianya hampir sama dengan rodagigi lurus, tetapi dalam pengoperasiannya rodagigi miring lebih lembut dan tingkat kebisingannya rendah dengan perkontakan antara gigi lebih dari 1.



Gambar 2.11 Rodagigi miring [6]

3. Rodagigi kerucut (*Bevel*)

Merupakan rodagigi yang paling sering dipakai, akan tetapi rodagigi ini sangat berisik dengan perbandingan kontak yang kecil, macam-macam roda gigi ini meliputi roda gigi kerucut lurus, spiral, miring, dan Zerol. Rodagigi kerucut digunakan untuk mentransmisikan 2 buah poros yang saling berpotongan.

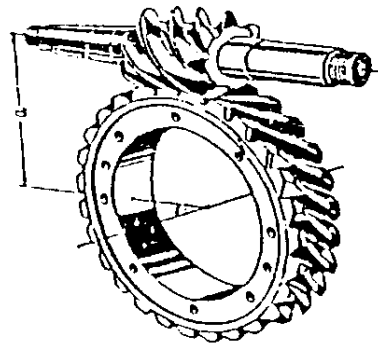


Gambar 2.12 Rodagigi kerucut [6]

4. Rodagigi cacing (*Worm*)

Ciri-ciri rodagigi cacing adalah:

- a. Kedua sumbu saling bersilang dengan jarak sebesar a , biasanya sudut yang dibentuk kedua sumbu sebesar 90° .
- b. Kerjanya halus dan hampir tanpa bunyi.
- c. Umumnya arah transmisi tidak dapat dibalik untuk menaikkan putaran dari roda cacing ke cacing (mengunci sendiri).
- d. Perbandingan reduksi bisa dibuat sampai 1 : 150.
- e. Kapasitas beban yang besar dimungkinkan karena kontak beberapa gigi (biasanya 2 sampai 4).
- f. Rodagigi cacing efisiensinya sangat rendah, terutama jika sudut kisarnya kecil.



Gambar 2.13 Rodagigi cacing [6]

2.4.2 Perbandingan putaran dan perbandingan rodagigi

Jika putaran rodagigi yang berpasangan dinyatakan dengan n_1 (rpm) pada poros penggerak dan n_2 (rpm) pada poros yang digerakkan, diameter lingkaran jarak bagi d_1 (mm) dan d_2 (mm) dan jumlah gigi z_1 dan z_2 , maka perbandingan putaran u adalah:

$$u = \frac{n_1}{n_2} = \frac{d_1}{d_2} = \frac{m \cdot z_1}{m \cdot z_2} = \frac{z_1}{z_2} = \frac{1}{i}$$

$$\frac{z_1}{z_2} = i$$

Harga i adalah perbandingan antara jumlah gigi pada rodagigi dan pinion, dikenal juga sebagai perbandingan transmisi atau perbandingan rodagigi. Perbandingan ini dapat sebesar 4 sampai 5 dalam hal rodagigi lurus standar, dan dapat diperbesar sampai 7 dengan perubahan kepala. Pada rodagigi miring ganda dapat sampai 10.

2.4.3 Penggunaan rodagigi kerucut (*Bevel*) dan *planetary gear*

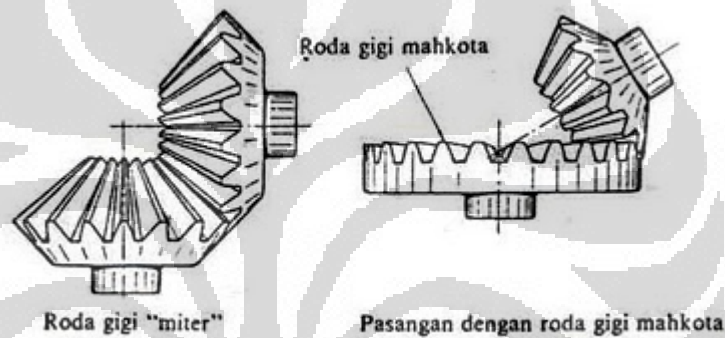
2.4.3.1 Penggunaan *bevel gear* (kerucut)

Pada pembuatan tugas akhir ini penulis menggunakan rodagigi kerucut untuk memindahkan gerakan antara poros yang berpotongan. Dengan sudut perpotongan antara kedua poros sebesar 90° . Rodagigi kerucut lurus adalah jenis rodagigi kerucut yang mudah dan sederhana pembuatannya dan memberikan hasil yang baik dalam pemakaiannya bila dipasangkan secara tepat dan teliti.

Sepasang rodagigi kerucut yang saling berkait dapat diwakili oleh dua bidang kerucut dengan titik puncak yang berhimpit

dan saling menggelinding tanpa slip. Kedua bidang kerucut ini disebut “kerucut jarak bagi”. Besarnya sudut puncak kerucut merupakan ukuran bagi putaran masing-masing porosnya. Rodagigi kerucut yang alur giginya lurus dan menuju ke puncak kerucut dinamakan “rodagigi kerucut lurus”.

Sumbu poros pada rodagigi kerucut biasanya berpotongan dengan sudut 90° . Bentuk khusus dari rodagigi kerucut dapat berupa “rodagigi miter” yang mempunyai sudut kerucut jarak bagi sebesar 45° dan rodagigi mahkota dengan sudut kerucut jarak bagi sebesar 90° . Dimana diperlihatkan pada Gambar 2.14 berikut:



Gambar 2.14 Rodagigi kerucut [7]

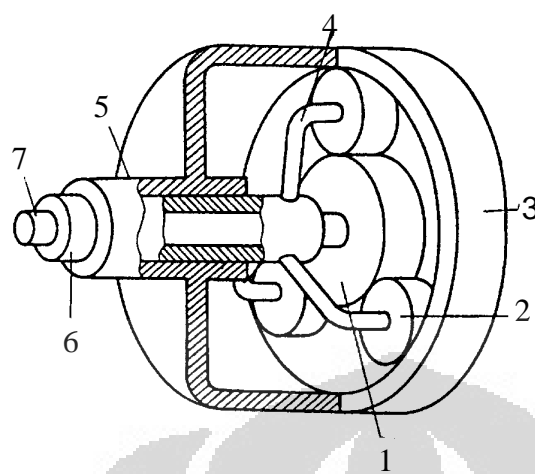
2.4.3.2 Penggunaan *planetary gear*

Planetary gear digunakan untuk mereduksi kecepatan dan meningkatkan torsi yang dihasilkan oleh sebuah motor listrik. Hal ini juga dibutuhkan untuk dapat menyesuaikan RPM motor sesuai dengan kecepatan dari poros sirip roket kendali. Rodagigi ini dapat memberikan torsi yang kuat bahkan pada kecepatan yang sangat rendah.



Gambar 2.15 *Planetary gear* buatan MAXON [8]

Berikut adalah bagian-bagian dari *Planetary gear* :

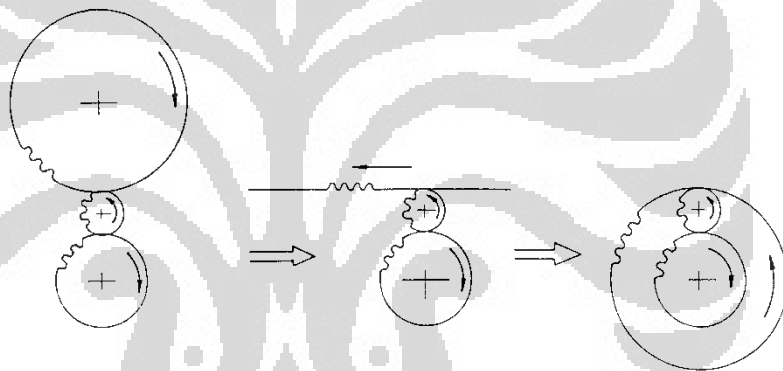


Bagian

1. Roda gigi matahari
2. Roda gigi planet
3. Roda gigi tabung
4. Penyangga roda gigi planet
5. Poros roda gigi tabung
6. Poros penyangga roda gigi planet
7. Poros roda gigi matahari

Gambar 2.16 Bagian *planetary gear* [9]

Putaran gigi set rodagigi planet:



Gambar 2.17 Prinsip putaran *planetary gear* [9]

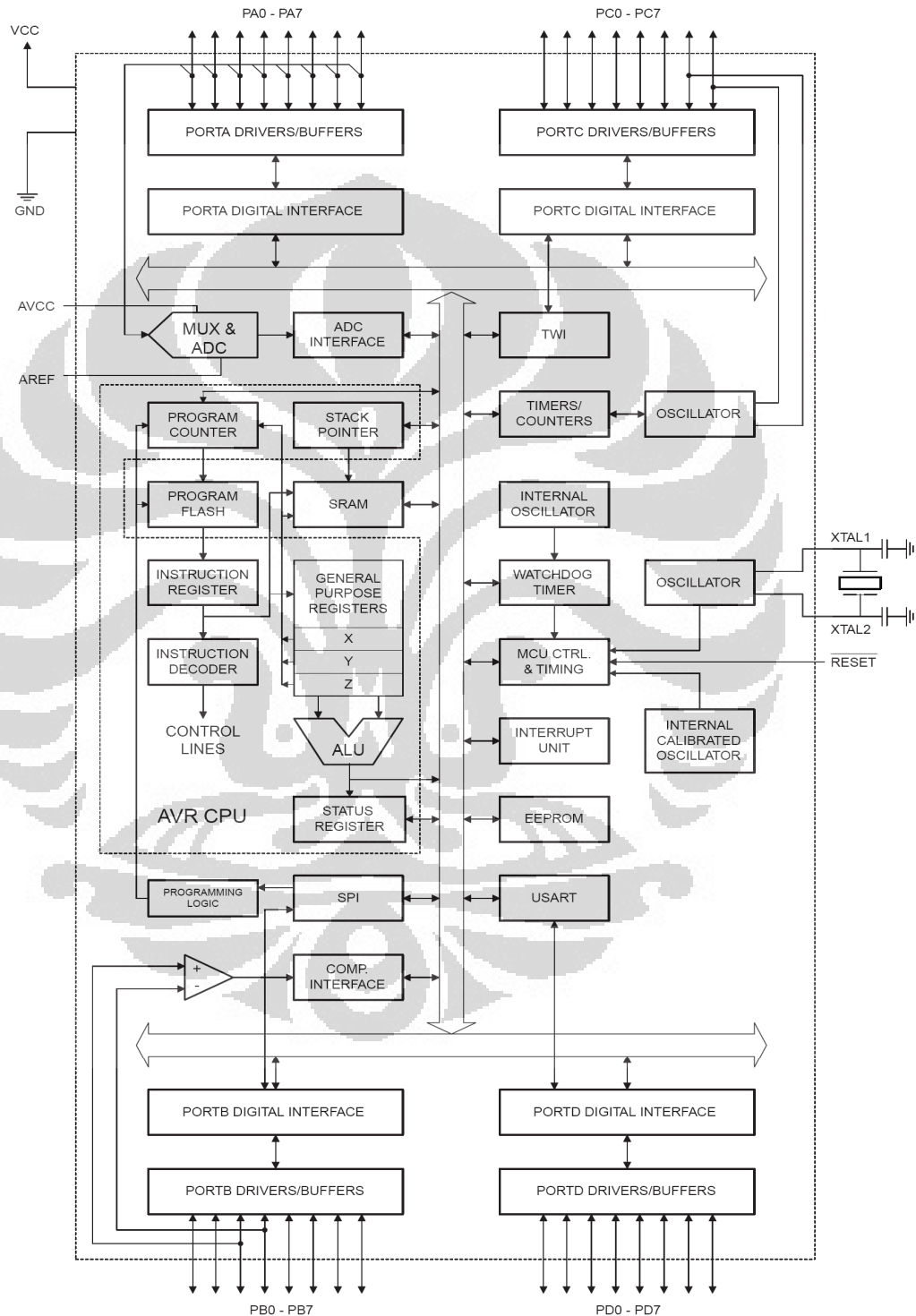
Set gigi biasa dengan gigi antara → gigi besar diluruskan menjadi batang bergigi → batang bergigi ditekuk menjadi roda gigi dalam (tabung).

2.5 Mikrokontroler AVR ATmega16

AVR merupakan seri mikrokontroler *Complementary Metal Oxide Semiconductor* (CMOS) buatan Atmel berbasis arsitektur RISC (*Reduced Instruction Set Computer*). Hampir semua instruksi pada program dieksekusi dalam satu siklus *clock*. AVR mempunyai 32 register *general-purpose*, *timer/counter* fleksibel dengan mode *compare*, interupsi *internal* dan *eksternal*, serial UART, *programmable Watchdog Timer*, *power saving mode*, ADC dan

PWM. AVR pun mempunyai *In-System Programmable (ISP) Flash on-chip* yang memungkinkan memori program untuk diprogram ulang (*read/write*) dengan koneksi secara serial yang disebut *Serial Peripheral Interface (SPI)*.

2.5.1 Arsitektur ATmega16



Gambar 2.18 Blok diagram arsitektur mikrokontroler ATmega16 [10]

AVR memiliki keunggulan dibandingkan dengan mikrokontroler lain, keunggulan mikrokontroler AVR yaitu memiliki kecepatan dalam mengeksekusi program yang lebih cepat, karena sebagian besar instruksi dieksekusi dalam 1 siklus *clock* (lebih cepat dibandingkan mikrokontroler keluarga MCS 51 yang memiliki arsitektur *Complex Instruction Set Compute*).

2.5.2 Fitur ATmega16

ATmega16 mempunyai *throughput* mendekati 1 *Millions Instruction Per Second* (MIPS) per MHz, sehingga membuat konsumsi daya menjadi rendah terhadap kecepatan proses eksekusi perintah.

Beberapa keistimewaan dari AVR ATmega16 antara lain:

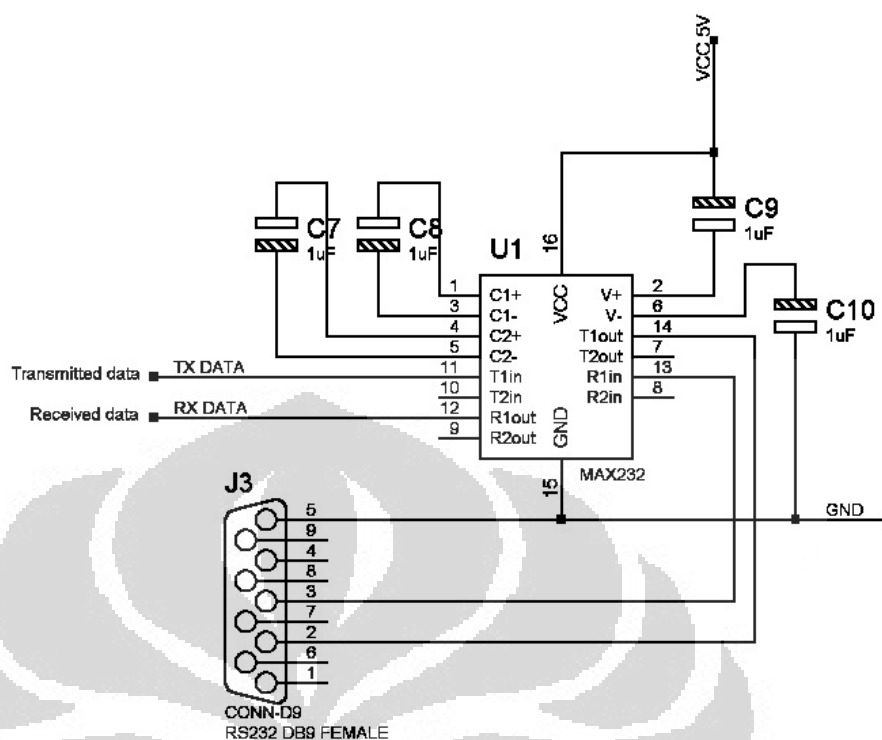
1. Mikrokontroler AVR 8 *bit* yang memiliki kemampuan tinggi dengan konsumsi daya rendah
2. Arsitektur RISC dengan *throughput* mencapai 16 MIPS pada frekuensi 16MHz
3. Memiliki kapasitas *Flash* memori 16 Kbyte, EEPROM 512 Byte dan SRAM 1 Kbyte
4. Saluran I/O sebanyak 32 buah, yaitu *Port A*, *Port B*, *Port C* dan *Port D*
5. CPU yang terdiri dari 32 buah *register*
6. Unit interupsi dan eksternal
7. *Port* USART untuk komunikasi serial
8. Fitur *peripheral*
 - a. Tiga buah *Timer/Counter* dengan kemampuan perbandingan (*compare*)
 - ✓ Dua buah *Timer/Counter* 8 *bit* dengan *Prescaler* terpisah dan *Mode Compare*
 - ✓ Satu buah *Timer/Counter* 16 *bit* dengan *Prescaler* terpisah, *Mode Compare* dan *Mode Capture*
 - b. *Real Time Counter* dengan *Oscillator* tersendiri
 - c. Empat kanal PWM
 - d. 8 kanal ADC

- ✓ *Single-ended Channel* dengan keluaran hasil konversi 8 dan 10 resolusi (register ADCH dan ADCL)
 - ✓ *7 Differential Channel* hanya pada kemasan *Thin Quad Flat Pack* (TQFP)
 - ✓ *2 Differential Channel* dengan *Programmable Gain*
- e. Antarmuka *Serial Peripheral Interface (SPI) Bus*
 - f. *Watchdog Timer* dengan *Oscillator Internal*
 - g. *On-chip Analog Comparator*

9. *Non-volatile program memory*

2.5.3 Komunikasi serial mikrokontroler ATmega16

Mikrokontroler dapat melakukan komunikasi dengan PC dengan menggunakan PORT Tx dan PORT Rx yang dimilikinya. Adapun komunikasi yang terjadi adalah melalui RS232. Serial MAX232 digunakan sebagai interface (antarmuka) dari PC ke perangkat luar (level TTL) atau sebaliknya dari perangkat luar ke PC. Tegangan pada RS232 berbeda dengan level tegangan digital. Rangkaian ini berfungsi untuk komunikasi antara alat dengan komputer. Komponen utama menggunakan IC MAX232 yaitu sebuah IC yang dapat mengubah format digital ke dalam sebuah format atau *level* RS232 dimana pada *level* RS232, tegangan *high* diwakili dengan tegangan +3 sampai +25 V, dan tegangan *low* diwakili dengan tegangan -3 sampai -25 V. Diantara -3 dan +3 merupakan tegangan *invalid* atau tidak sah. Tegangan yang cukup tinggi ini memungkinkan data dapat ditransmisikan cukup jauh.



Gambar 2.19 Rangkaian komunikasi serial RS232 dengan Rx, Tx

Data akan diterima oleh mikrokontroler ATmega16 dan akan disimpan dalam memori, Untuk keperluan *monitoring data*, maka mikrokontroler dapat dihubungkan dengan display atau PC sehingga dapat selalu menunjukkan data dari output serial.

2.5.4 Analog to Digital Converter (ADC)

ADC digunakan untuk mengubah sinyal input yang analog menjadi sinyal digital agar dapat diolah oleh mikrokontroler. Adapun fitur dari ADC ATmega16 adalah sebagai berikut:

1. Resolusi 10 bit.
2. Waktu konversi 65-260 us.
3. 0-vcc *range input* ADC.
4. Memiliki 8 *channel* input(PA0-PA7).
5. Tiga mode pemilihan tegangan referensi.

Pada ADC ATmega16 digunakan dua mode ADC, yaitu *single conversion* dan *free running*. Pada mode *single conversion*, pengguna harus mengaktifkan setiap kali ADC akan digunakan. Sedangkan pada mode *free*

running, pengguna cukup sekali saja mengaktifkan, sehingga ADC akan terus mengkonversi tanpa henti.

Mengatur sumber tegangan referensi pada ADC merupakan batas rentang representasi nilai digital hasil konversi. Hasil konversi pada mode *single ended conversion* dirumuskan pada persamaan 2.4.

$$ADC = \frac{V_{IN} \cdot 1024}{V_{REF}} \quad (2.4)$$

Keterangan:

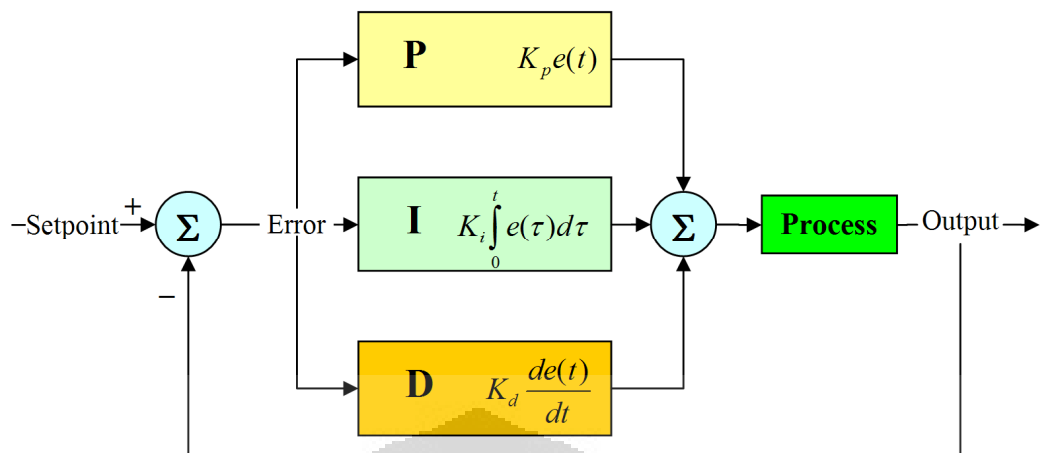
V_{IN} = tegangan masukan analog pada kanal ADC yang aktif

V_{REF} = tegangan referensi yang dipilih

2.6 Pengendali PID

PID (Proportional Integral Derivative) merupakan kontroler untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Komponen kontrol PID ini terdiri dari tiga jenis yaitu Proportional, Integratif dan Derivatif. Ketiganya dapat dipakai bersamaan maupun sendiri-sendiri tergantung dari respon yang kita inginkan terhadap suatu plant. Seiring dengan berkembangnya dunia digital (terutama mikroprosesor dan mikrokontroler) maka algoritma kontrol PID dapat direalisasikan kedalam bentuk persamaan PID digital.

Pada pembahasan skripsi ini menggunakan realisasi praktis dari PID digital. Istilah PID digital pada dasarnya mengacu pada jenis perangkat keras dimana sistem kontrol PID tersebut ditanamkan. Berbeda dengan kontrol PID analog yang realisasi praktisnya dijumpai dalam bentuk perangkat keras rangkaian elektronika, sistem kontrol PID digital implementasinya dapat dijumpai dalam bentuk persamaan matematis yang diprogram pada sistem mikrokontroler.



Gambar 2.20 Blok diagram dari kontroler PID [11]

Berbeda dengan kontrol PID analog yang pengolahannya bersifat kontinyu. Didalam sistem mikrokontroler, pengolahan sinyal kontrol oleh modul digital dilakukan hanya pada waktu-waktu diskret. Dalam hal ini, konversi sinyal dari analog ke digital, pengolahan sinyal error, sampai konversi balik digital ke analog dilakukan pada interval atau waktu cuplik (sampling) T_c tertentu.

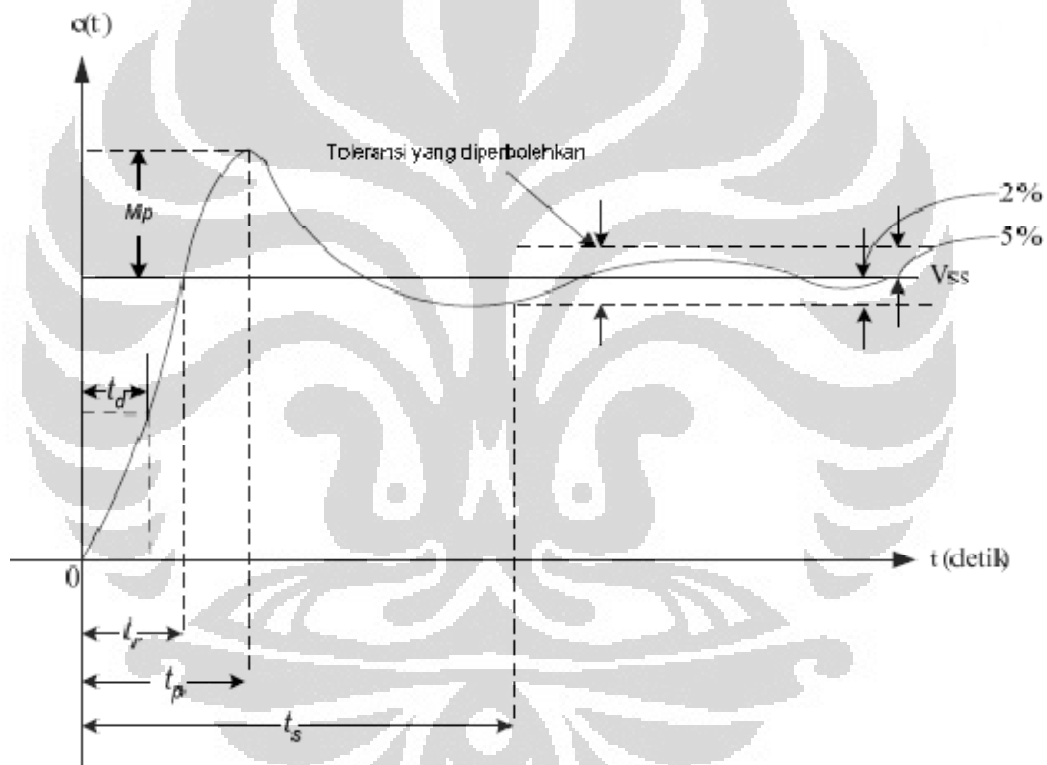
Pengendali PID terdiri dari 3 jenis cara pengaturan yang saling dikombinasikan, yaitu Pengendali P (Proportional), Pengendali I (Integral), dan Pengendali D (Derivatif) masing-masing memiliki parameter tertentu yang harus diatur untuk dapat beroperasi dengan baik, yang disebut dengan konstanta. Kontroler jenis ini mempunyai kelebihan-kelebihan yang dimiliki oleh masing-masing kontroler pembentuknya. Bagian *proportional controller* akan merespon *error* secara proporsional; bagian *integral controller* akan menghilangkan *steady state error*; sedangkan bagian *derivative controller* mampu untuk merespon perubahan *error* secara cepat. Untuk lebih jelasnya dapat dilihat pada tabel 2.2 berikut.

Tabel 2.2 Respon pengendali PID terhadap perubahan konstanta

Respon	<i>Rise time</i>	<i>Overshoot</i>	<i>Settling time</i>	Ess
Kp	Menurun	Meningkat	Perubahan kecil	Menurun
Ki	Menurun	Meningkat	Meningkat	Mendekati nol
Kd	Perubahan kecil	Menurun	Menurun	Perubahan kecil

Parameter-parameter tersebut tidak bersifat independent, sehingga pada saat salah satu nilai konstantanya diubah, maka mungkin sistem tidak akan bereaksi seperti yang diinginkan. Tabel diatas hanya dipergunakan sebagai pedoman jika akan melakukan perubahan konstanta. Untuk merancang pengendali PID biasanya dipergunakan metode *trial and error*. Sehingga perancang harus mencoba kombinasi pengatur beserta konstantanya untuk mendapatkan hasil terbaik yang paling sederhana.

Tanggapan transien suatu sistem kontrol selalu menunjukkan osilasi teredam sebelum mencapai keadaan mantapnya. Tanggapan transient suatu sistem kontrol, terhadap *input step* dapat dilihat pada Gambar 2.21 berikut.



Gambar 2.21 Kurva tanggapan transien

Keterangan:

- ✓ t_d = Waktu tunda, yaitu waktu yang diperlukan tanggapan untuk mencapai setengah nilai akhir untuk waktu yang pertama.
- ✓ t_r = Waktu naik, yaitu waktu yang diperlukan tanggapan untuk naik dari 10% menjadi 90%.
- ✓ t_p = Waktu puncak, yaitu waktu yang diperlukan tanggapan untuk mencapai puncak pertama overshoot.

- ✓ M_p = Maximum (persen) overshoot (<20%).
- ✓ t_s = Setling time, Waktu yang diperlukan untuk mencapai keadaan mantap (2% atau 5%).

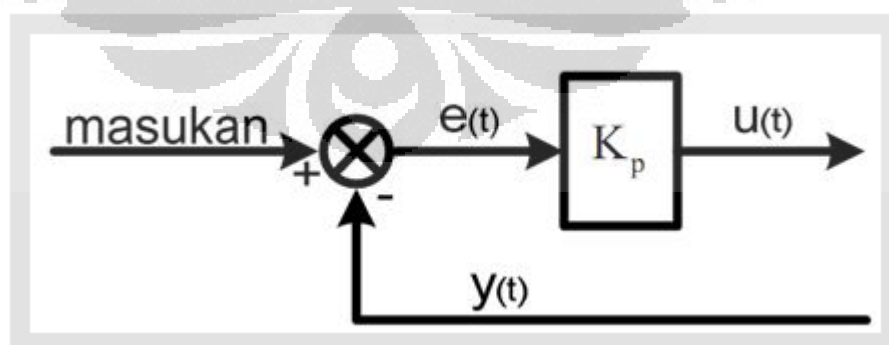
2.6.1 Pengendali Proporsional

Pengendali proposional memiliki keluaran yang sebanding/proposional dengan besarnya sinyal kesalahan (selisih antara besaran yang diinginkan dengan harga aktualnya). Secara lebih sederhana dapat dikatakan, bahwa keluaran kontroler proporsional merupakan perkalian antara konstanta proporsional dengan masukannya. Perubahan pada sinyal masukan akan segera menyebabkan sistem secara langsung mengubah keluarannya sebesar konstanta pengalinya.

Kontrol P jika $G(s) = k_p$, dengan k adalah konstanta. Jika $u = G(s) \cdot e$ maka $u = K_p \cdot e$ dengan K_p adalah Konstanta Proporsional. K_p berlaku sebagai *Gain* (penguat) saja tanpa memberikan efek dinamik kepada kinerja kontroler. Penggunaan kontrol P memiliki berbagai keterbatasan karena sifat kontrol yang tidak dinamik ini. Walaupun demikian dalam aplikasi aplikasi dasar yang sederhana kontrol P ini cukup mampu untuk memperbaiki respon transien khususnya *rise time* dan *settling time*. Hubungan antara $u(t)$ dengan sinyal error $e(t)$ terlihat pada persamaan 2.5.

$$u(t) = K_p e(t) \quad (2.5)$$

K_p adalah konstanta proporsional. Blok diagram kontrol proporsional dapat dilihat pada Gambar 2.22.



Gambar 2.22 Blok diagram kontrol proporsional

Ciri-ciri kontroler proporsional harus diperhatikan ketika kontroler tersebut diterapkan pada suatu sistem. Secara eksperimen, pengguna

kontroler proporsional harus memperhatikan ketentuan-ketentuan berikut ini:

1. Jika nilai K_p kecil, kontroler proporsional hanya mampu melakukan koreksi kesalahan yang kecil, sehingga akan menghasilkan respon sistem yang lambat.
2. Jika nilai K_p dinaikkan, respon sistem menunjukkan semakin cepat mencapai keadaan *steady state*.
3. Namun jika nilai K_p diperbesar sehingga mencapai harga yang berlebihan, akan mengakibatkan sistem bekerja tidak stabil, atau respon sistem akan berosilasi.

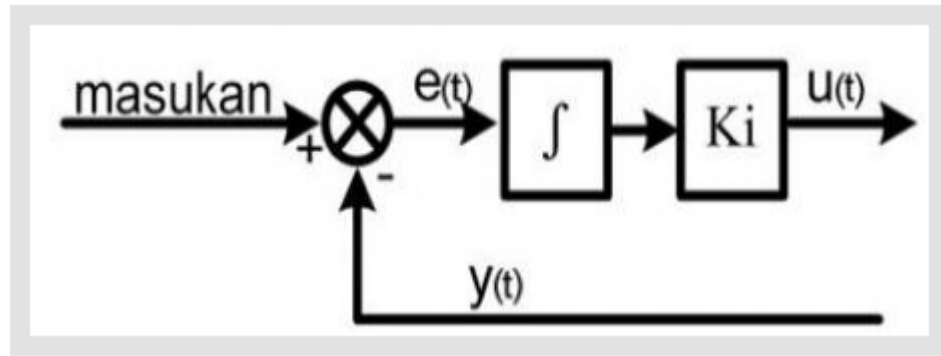
2.6.2 Pengendali Integral

Pengendali integral berfungsi menghasilkan respon sistem yang memiliki kesalahan *steady state* nol. Kalau sebuah *plant* tidak memiliki unsur integrator ($1/s$), pengendali proporsional tidak akan mampu menjamin keluaran sistem dengan kesalahan *steady state* nol. Dengan kontroler integral, respon sistem dapat diperbaiki, yaitu mempunyai kesalahan *steady state*-nya nol.

Jika $G(s)$ adalah kontrol I maka u dapat dinyatakan sebagai $u(t) = [\int e(t) dt] \cdot K_i$ dengan K_i adalah konstanta integral, dan dari persamaan di atas, $G(s)$ dapat dinyatakan sebagai $u = K_i \cdot [de/dt]$ Jika $e(\tau)$ mendekati konstan (bukan nol) maka $u(t)$ akan menjadi sangat besar sehingga diharapkan dapat memperbaiki error. Jika $e(\tau)$ mendekati nol maka efek kontrol I ini semakin kecil. Kontrol I dapat memperbaiki sekaligus menghilangkan respon *steady-state*, namun pemilihan K_i yang tidak tepat dapat menyebabkan respon transien yang tinggi sehingga dapat menyebabkan ketidakstabilan sistem. Pemilihan K_i yang sangat tinggi justru dapat menyebabkan output berosilasi karena menambah orde sistem. Hubungan antara output kontrol integral $u(t)$ dengan sinyal error $e(\tau)$ terlihat pada persamaan 2.6.

$$u(t) = K_i \int_0^t e(t) \quad (2.6)$$

K_i adalah konstanta integral. Blok diagram pengendali integral ditunjukkan pada Gambar 2.23.



Gambar 2.23 Blok diagram pengendali integral

Pengendali integral memiliki karakteristik seperti halnya sebuah integral. Keluaran kontroler sangat dipengaruhi oleh perubahan yang sebanding dengan nilai sinyal kesalahan. Keluaran kontroler ini merupakan jumlahan yang terus menerus dari perubahan masukannya. Kalau sinyal kesalahan tidak mengalami perubahan, keluaran akan menjaga keadaan seperti sebelum terjadinya perubahan masukan.

Ketika digunakan, pengendali integral mempunyai beberapa karakteristik berikut ini:

1. Keluaran kontroler membutuhkan selang waktu tertentu, sehingga kontroler integral cenderung memperlambat respon.
2. Ketika sinyal kesalahan berharga nol, keluaran kontroler akan bertahan pada nilai sebelumnya.
3. Jika sinyal kesalahan tidak berharga nol, keluaran akan menunjukkan kenaikan atau penurunan yang dipengaruhi oleh besarnya sinyal kesalahan dan nilai K_i
4. Konstanta integral K_i yang berharga besar akan mempercepat hilangnya *offset*. Tetapi semakin besar nilai konstanta K_i akan mengakibatkan peningkatan osilasi dari sinyal keluaran kontroler.

2.6.3 Pengendali Derivatif

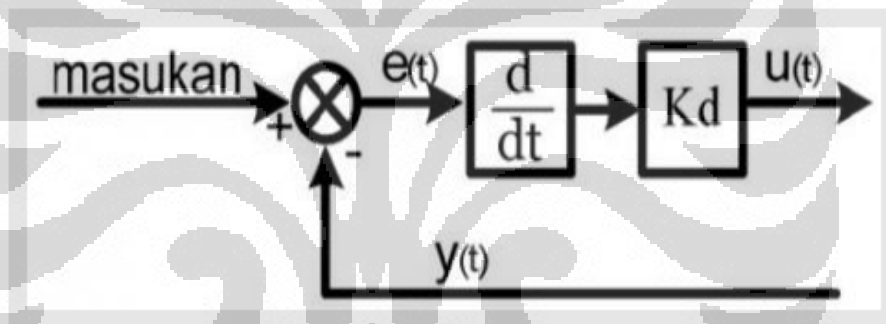
Pengendali derivatif dapat disebut pengendali laju, karena output kontroler sebanding dengan laju perubahan sinyal *error*. Sinyal kontrol u yang dihasilkan oleh kontrol D dapat dinyatakan sebagai $G(s) = s.K_d$. Dari persamaan ini, nampak bahwa sifat dari kontrol D ini dalam konteks "kecepatan" atau rate dari *error*. Dengan sifat ini ia dapat digunakan untuk

memperbaiki respon transien dengan memprediksi *error* yang akan terjadi. Kontrol derivatif hanya berubah saat ada perubahan *error* sehingga saat *error* statis kontrol ini tidak akan bereaksi, hal ini pula yang menyebabkan pengendali derivatif tidak dapat dipakai sendiri. Hubungan antara output kontrol derivatif $u(t)$ dengan sinyal error $e(t)$ dapat dilihat pada persamaan 2.7.

$$u(t) = K_d \frac{de(t)}{dt} \quad (2.7)$$

Pada periode peralihan, kontrol derivatif menyebabkan adanya redaman pada sistem sehingga lebih memperkecil lonjakan. Seperti pada kontrol proporsional, kontrol derivatif juga tidak dapat menghilangkan *offset*.

Blok diagram pengendali derivatif ditunjukkan pada gambar berikut.



Gambar 2.24 Blok diagram pengendali derivatif

Karakteristik pengendali derivatif adalah sebagai berikut:

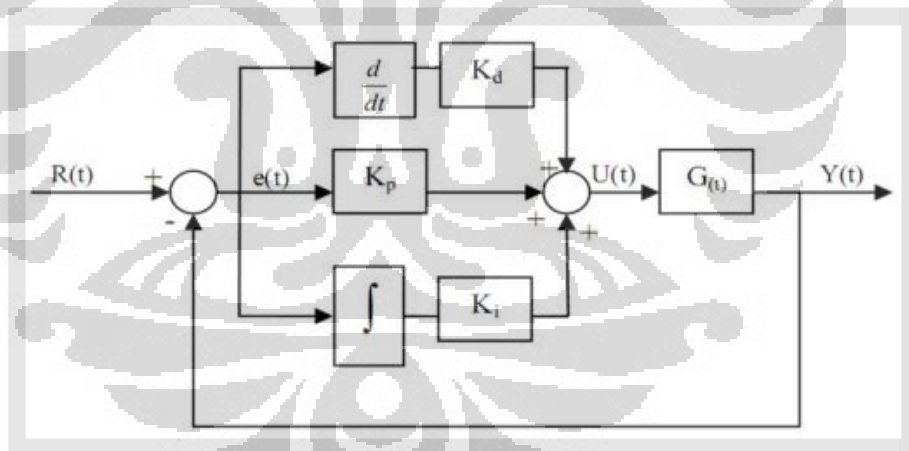
1. Pengendali ini tidak dapat menghasilkan keluaran bila tidak ada perubahan pada masukannya (berupa sinyal kesalahan).
2. Jika sinyal kesalahan berubah terhadap waktu, maka keluaran yang dihasilkan kontroler tergantung pada nilai T_d dan laju perubahan sinyal kesalahan.
3. Pengendali derivatif mempunyai suatu karakter untuk mendahului, sehingga kontroler ini dapat menghasilkan koreksi yang signifikan sebelum pembangkit kesalahan menjadi sangat besar. Jadi kontroler derivatif dapat mengantisipasi pembangkit kesalahan, memberikan aksi yang bersifat korektif, dan cenderung meningkatkan stabilitas sistem.

Berdasarkan karakteristik kontroler tersebut, kontroler derivatif umumnya dipakai untuk mempercepat respon awal suatu sistem, tetapi tidak memperkecil kesalahan pada keadaan tunaknya. Kerja pengendali derivatif hanyalah efektif pada lingkup yang sempit, yaitu pada periode peralihan. Oleh sebab itu kontroler derivatif tidak pernah digunakan tanpa ada kontroler lain sebuah sistem.

2.6.4 Pengendali Proporsional, Integral, dan Derivatif

Kombinasi dari aksi kontrol proporsional, aksi kontrol integral, dan aksi kontrol derivatif disebut aksi kontrol proporsional ditambah integral ditambah derivatif. Kombinasi ini mempunyai keuntungan dibanding masing-masing kontroler. Elemen-elemen controller P, I dan D masing-masing secara keseluruhan bertujuan untuk mempercepat reaksi sebuah sistem, menghilangkan *offset* dan menghasilkan perubahan awal yang besar.

Gabungan dari pengendali proporsional, integral, dan derivatif adalah kontrol PID. Diagram blok dari kontrol PID dapat dilihat pada gambar berikut.



Gambar 2.25 Blok diagram kontrol PID [13]

Sehingga persamaan untuk kontrol PID adalah:

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (2.8)$$

$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right) \quad (2.9)$$

Dengan:

$u(t)$ = sinyal output pengendali PID

K_p = konstanta proporsional

T_i = waktu integral

T_d = waktu derivatif

K_i = konstanta integral (K_p/T_i)

K_d = konstanta derivatif ($K_p \times T_d$)

$e(t)$ = sinyal error = referensi - keluaran plant = set point – nilai sensor

Untuk persamaan PID 2.8 merupakan bentuk PID independent dan persamaan PID 2.9 merupakan PID bentuk dependent. Istilah tersebut mengacu kepada ketergantungan setiap suku persamaan terhadap nilai K_p . Untuk persamaan 2.8, jika dilakukan perubahan nilai pada konstanta proporsional (K_p) maka tidak akan mempengaruhi konstanta parameter lainnya. Sedangkan untuk persamaan 2.9, dengan mengubah nilai K_p maka akan berubah nilai dari parameter-parameter lainnya. Pada pembuatan skripsi ini menggunakan persamaan PID bentuk dependent. Jika ingin menggunakan persamaan dependent, maka tinggal memasukkan nilai dari $K_i=K_p/T_i$ dan $K_d=K_p \times T_d$.

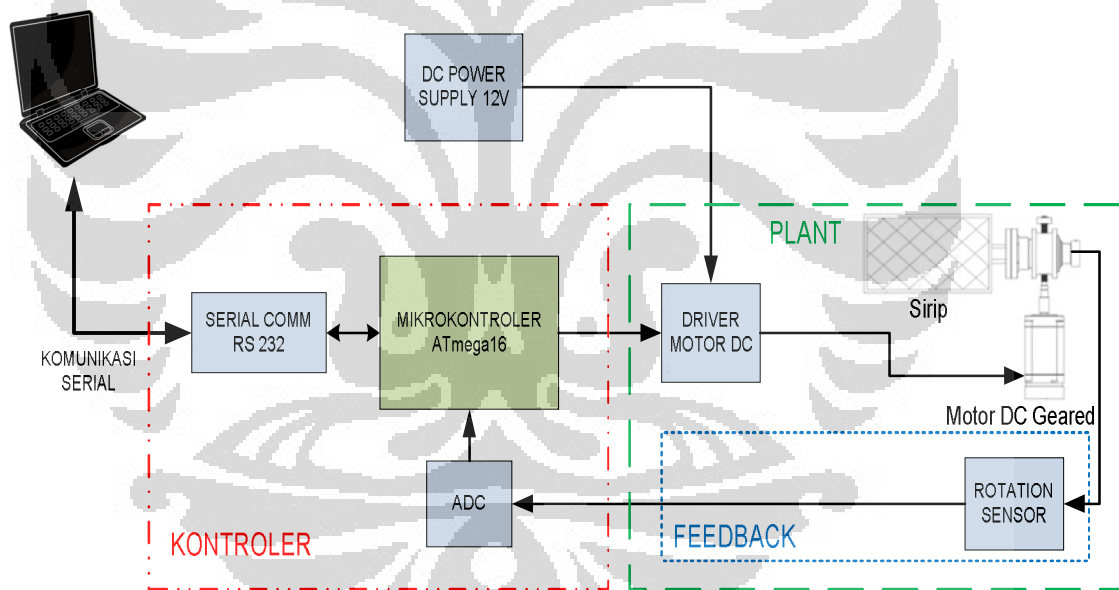
Pada persamaan 2.5 sampai persamaan 2.9 merupakan persamaan dalam kawasan waktu continuous (analog). Sedangkan agar persamaan tersebut dapat direalisasikan dalam bentuk pemrograman, maka persamaan dalam bentuk waktu continuous tersebut harus didiskretisasi terlebih dahulu menjadi kawasan digital.

BAB 3 PERANCANGAN SISTEM

Pada bab ini membahas mengenai perancangan dan pembuatan alat serta ide-ide dan gambaran tentang alat yang akan dibuat. Bagian ini akan menampilkan komponen yang dipergunakan, blok diagram, gambar tiap modul dan penjelasan singkatnya serta perancangan dan fungsi tiap-tiap bagian.

3.1 Deskripsi sistem

Perancangan dan pembuatan prototipe ini melibatkan kepresisian dari pergerakan aktuator sebagai penggerak sirip roket. Respon dari pergerakan motor DC sebagai aktuator utama dikendalikan menggunakan teknik algoritma kendali PID.



Gambar 3.1 Blok diagram sistem

Blok diagram diatas terdiri dari kontroler, *plant*, *feedback*, dan PC sebagai kendali pusat. PC memberikan nilai setpoint berupa besaran derajat ke mikrokontroler ATmega16. Kemudian mikrokontroler akan memproses input data serial tersebut. Dengan algoritma kendali PID yang telah dibuat, akan menghasilkan input untuk plant berupa nilai duty cycle. Aktuator pada plant yaitu motor DC yang telah terhubung dengan *gear* akan merespon input tersebut dengan menghasilkan perubahan posisi. Perubahan posisi tersebut diukur

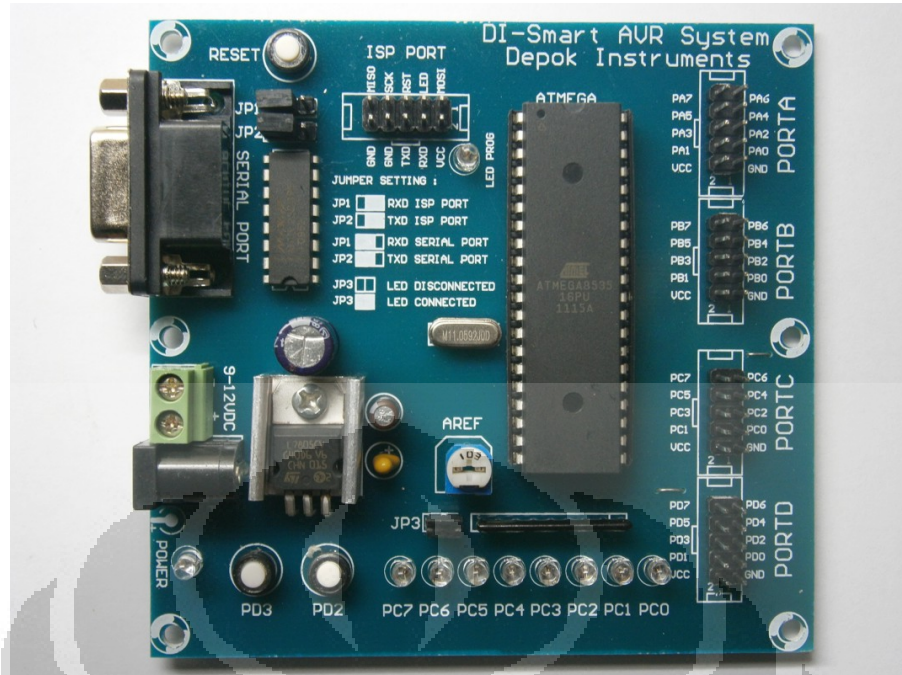
menggunakan *rotation sensor*. *Rotation sensor* kemudian memberikan *feedback* ke mikrokontroler. Mikrokontroler mengolah nilai perubahan posisi tersebut, kemudian membandingkan nilai dari *feedback rotation sensor* dengan nilai dari setpoint. Perbedaan nilai tersebut menghasilkan nilai *error* yang kemudian diolah pada program kontrol untuk menjadi input pada *plant* kembali. Nilai-nilai pengukuran yang didapat direkam dan ditampilkan pada komputer untuk selanjutnya dapat dianalisis.

3.2 Perancangan perangkat keras

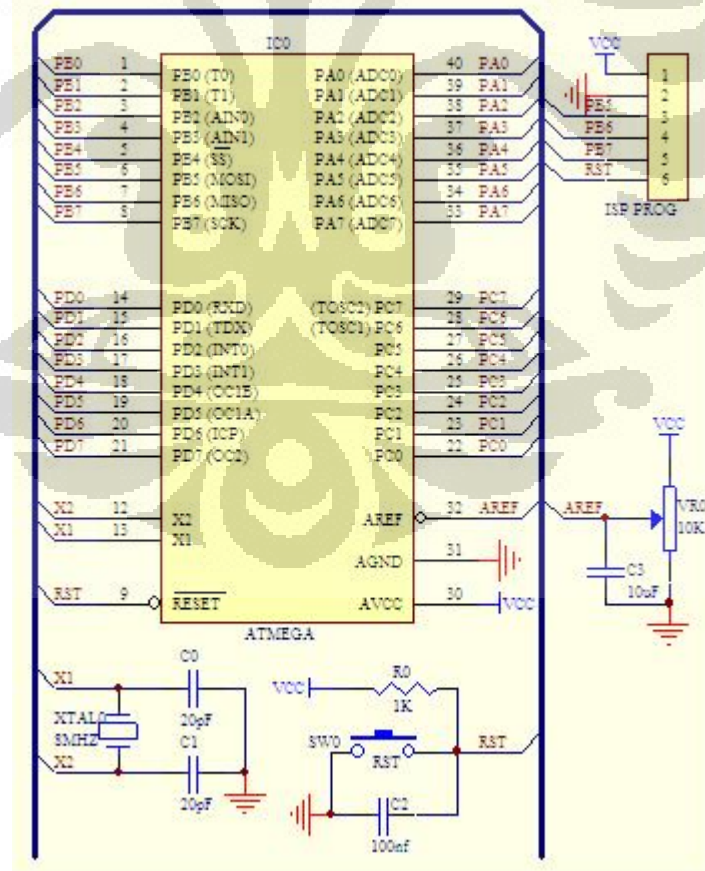
Perancangan blok diagram sistem kemudian diaplikasikan ke perancangan perangkat keras. Masing-masing komponen dirangkai hingga menjadi rangkaian perangkat keras. Perancangan perangkat keras yang digunakan terbagi menjadi tiga bagian utama, yaitu bagian kontroler, *plant* (*Driver* motor DC, motor DC, *gear*), dan *feedback* (*rotation sensor*).

3.2.1 Perancangan mikrokontroler

Pada sistem ini digunakan mikrokontroler jenis AVR ATmega16 dari keluarga ATMEL sebagai pengendali utama. Kontroler ini mampu melakukan komunikasi data serial secara UART RS-232 dengan PC, pemrograman memori melalui ISP (*In-System Programming*), mengolah data, dan untuk menjalankan piranti-piranti lain serta membaca sensor. Bentuk modul minimum sistem AVR ATmega16 dapat dilihat pada Gambar 3.2 dan rangkaian minimum sistem AVR ATmega16 dapat dilihat pada Gambar 3.3.



Gambar 3.2 Modul minimum sistem AVR ATmega16



Gambar 3.3 Rangkaian minimum sistem AVR ATmega16

Dalam sistem ini pin-pin mikrokontroler yang digunakan adalah sebagai berikut:

1. Pin A.0 digunakan sebagai pin masukan untuk *rotation sensor* serta dipergunakan sebagai ADC
2. Pin C.0 dan C.1 digunakan sebagai pin keluaran untuk mengatur arah putaran motor DC
3. Pin C.4 digunakan sebagai pin keluaran untuk mengatur kecepatan putar motor DC
4. Pin D.0 dan D.1 digunakan sebagai pin komunikasi data serial dengan komputer

3.2.2 Perancangan sirip dan *pitch gauge*

Desain bentuk sirip dapat dilihat pada Gambar 3.4. Sirip dibuat dari bahan pertinax (paper phenolic). Sirip dihubungkan dengan *pitch gauge* untuk melihat perubahan sudut.



Gambar 3.4 Bentuk sirip pada prototipe

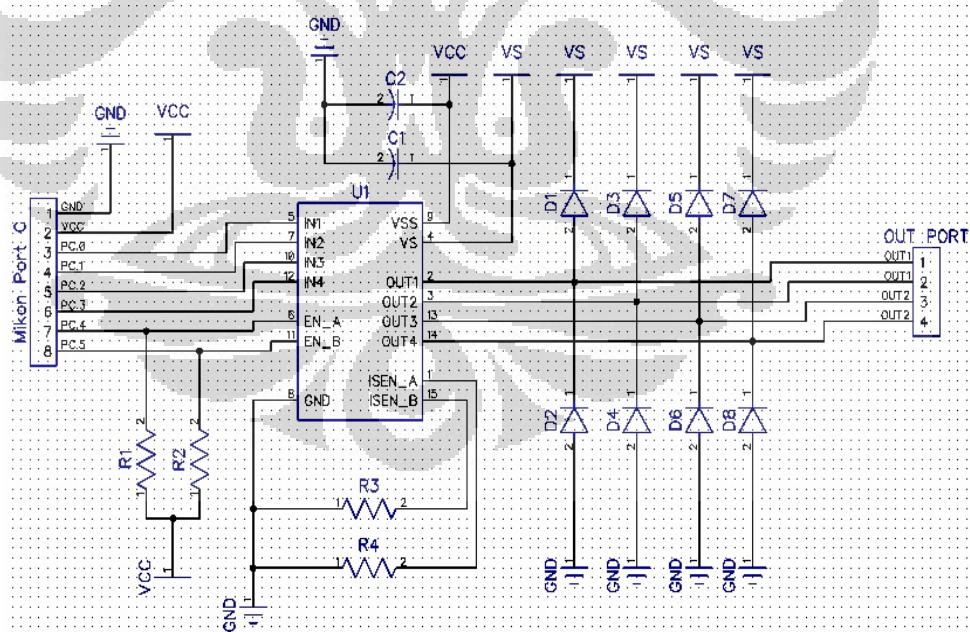
Pitch gauge digunakan sebagai alat ukur sudut pada sirip dengan jarak antara -10° sampai $+10^{\circ}$. Alat ini akan dihubungkan dengan sirip yang apabila sirip bergerak, maka alat ini pun akan bergerak dan menunjukkan besar perubahan sudut yang terjadi. Bentuk *pitch gauge* dapat dilihat pada gambar 3.5.



Gambar 3.5 *Pitch gauge* dengan besaran sudut -10° sampai $+10^{\circ}$

3.2.3 Perancangan driver motor DC

Rangkaian driver motor DC digunakan untuk mengendalikan putaran motor baik dalam hal arah putaran maupun kecepatan putaran. Rangkaian driver ini menggunakan IC L298N. IC L298N memiliki konfigurasi 2 H-Bridge driver motor dengan 4 input dan 4 output untuk mengendalikan arah putaran motor. IC driver motor ini dapat mengendalikan motor DC dari input yang berlogika high atau low. Di bawah ini adalah gambar dari rangkaian driver motor menggunakan IC L298N.



Gambar 3.6 Rangkaian driver motor DC menggunakan IC L298N

Rangkaian driver ini berfungsi untuk menggerakkan motor sesuai dengan input yang diterimanya. Pin 5 yang dihubungkan ke pin C.0 dan

pin 7 yang dihubungkan ke pin C.1 pada rangkaian merupakan pin yang menentukan arah putaran motor, apakah bergerak searah jarum jam atau bergerak berlawanan arah jarum jam. Nilai pin ini ditentukan dari output mikrokontroler. Pin 6 yang dihubungkan ke pin C.4 merupakan pin PWM yang menentukan besarnya kecepatan motor. Pin Vs merupakan sumber tegangan untuk motor yaitu sebesar 12V. Dioda schottky digunakan untuk menangkap back emf motor yang dihasilkan oleh kumparan motor saat power on dan off. Tegangan *flyback* ini dapat lebih tinggi daripada tegangan sumber. Jika tidak digunakan diode, maka transistor dapat terbakar. Berikut ini adalah gambar perangkat keras driver motor DC.



Gambar 3.7 Perangkat keras driver motor DC

3.2.4 Motor DC yang digunakan

Motor DC yang digunakan yaitu motor DC produksi Maxon dengan nomor seri manufaktur 242472 jenis *Brushed*. Motor ini mempunyai tegangan nominal 24 volt, daya yang dihasilkan 15 watt, kecepatan 5930 rpm, dan torsi 3,67 Ncm, diameter 32 mm, panjang badan motor 62,9 mm.

Tabel 3.1 Data *brushed* DC motor produksi maxon

Brushed DC motor produksi Maxon	
Tegangan Nominal	24 Volt
Daya yang dihasilkan	15 Watt
Kecepatan tanpa beban	5930 rpm
Torsi	3,67 Ncm
Diameter	32 mm
Panjang badan motor	62,9 mm



Gambar 3.8 Maxon *brushed* DC motor

3.2.5 Gear yang digunakan

Gear yang digunakan pada sistem adalah jenis *planetary gear* produksi Maxon dengan nomor seri manufaktur 110375 yang mempunyai diameter 32 mm, panjang badan 49,8 mm, panjang shaft 21 mm, diameter shaft 5,5 mm, reduksi kecepatan 246:1. *Gear* ini digunakan karena mempunyai torsi yang besar yaitu 6.75 Nm dan mampu mereduksi kecepatan putaran motor sebesar 246 putaran menjadi 1 putaran dengan torsi yang besar. Dipilihnya *gear* jenis ini karena arah putaran keluaran yang dihasilkan sama dengan arah putaran motor. Berikut adalah gambar *gear* yang digunakan.



Gambar 3.9 Gear jenis *planetary gear*

Tabel 3.2 Data *planetary gear* produksi maxon

<i>Planetary gear</i> produksi Maxon	
Diameter	32 mm
Panjang badan	49,8 mm
Panjang shaft	21 mm
Diameter shaft	5,5 mm
Reduksi kecepatan	246 : 1
Torsi	6,75 Nm

3.2.6 *Rotation sensor*

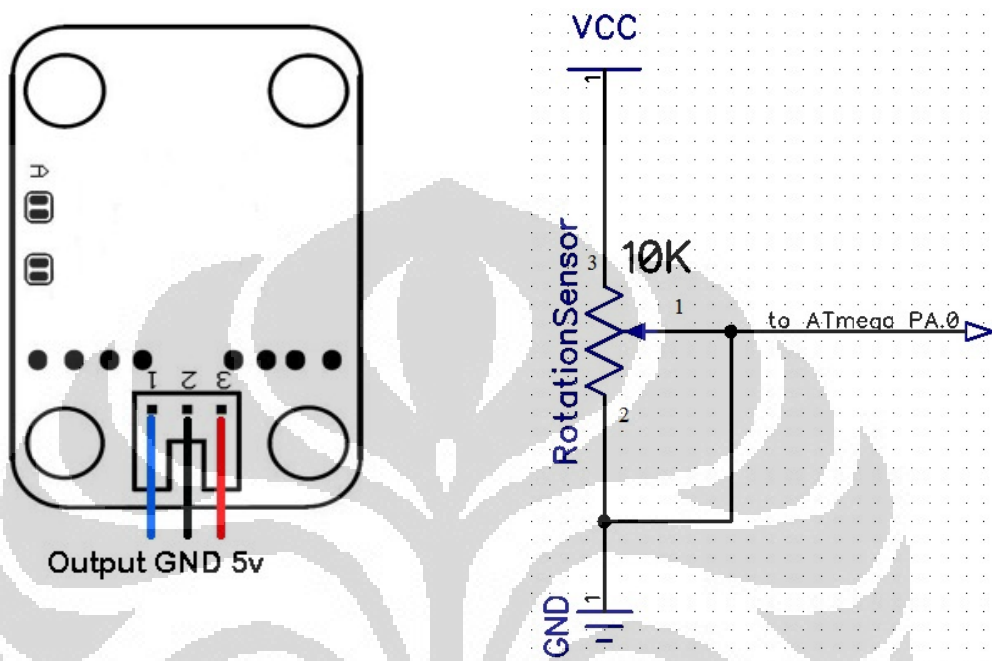
Pada perancangan sensor posisi pergerakan sirip digunakan potensiometer jenis trimmers (single turn) sebesar 10K, dengan tegangan referensi yang diberikan sebesar 5 Volt. Digunakannya potensiometer jenis ini karena cukup memiliki linearitas output terhadap besaran yang diukurnya. *Rotation sensor* yang digunakan diproduksi oleh DFRobot. Bentuk *hardware* dari sensor ini dapat dilihat pada Gambar 3.10.

Gambar 3.10 *Rotation sensor*

Sensor ini mempunyai 3 terminal, terminal 1 digunakan sebagai terminal sinyal (output), terminal 2 digunakan untuk terminal *ground*, dan terminal 3 digunakan sebagai terminal sumber tegangan referensi 5 volt. Output dari terminal 1 yaitu kondisi dimana terminal tersebut terhubung dengan mikrokontroler. Sensor ini tergolong sebagai sensor mekanik yaitu menggunakan sentuhan (putaran) dalam menghasilkan output. Gambar

3.11 menunjukkan skematik rangkaian *rotation sensor*. Tegangan keluaran dari sensor dapat dihitung menggunakan rumus berikut:

$$V_{out} = \frac{R_{12}}{R_{32}} \times V_{CC} \quad (3.1)$$



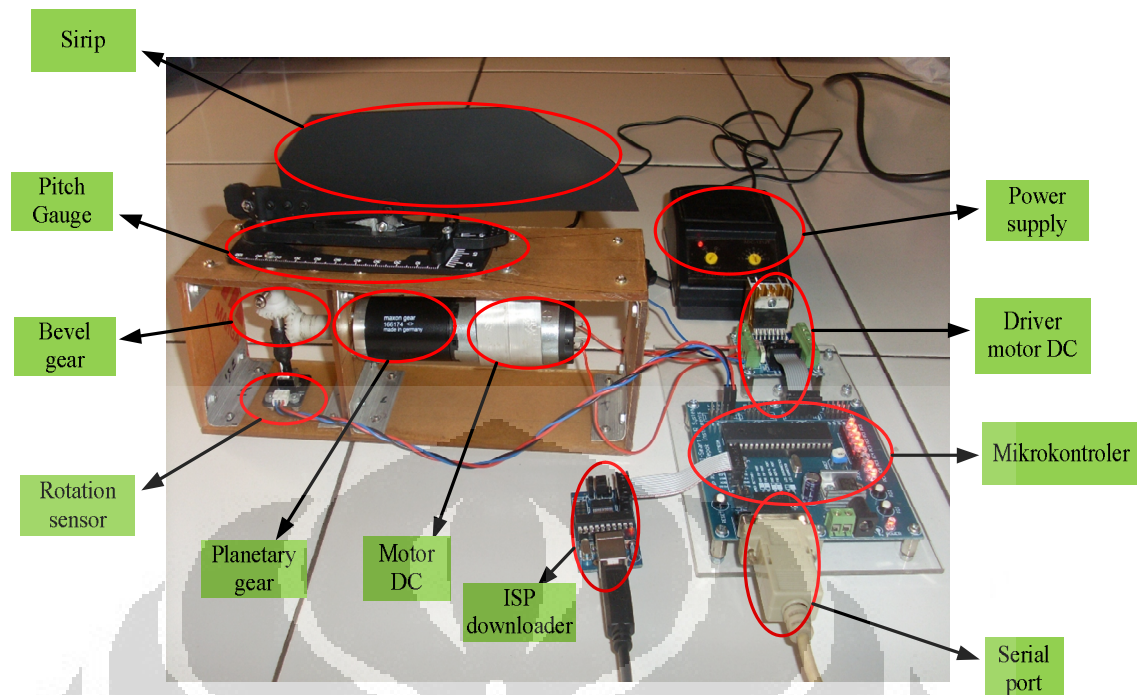
Gambar 3.11 Skematik rangkaian *rotation sensor*

3.2.7 Perangkat keras pendukung

Perangkat keras pendukung terdiri atas PC, peralatan serial, dan *power supply*. PC digunakan untuk memasukkan setpoint besaran sudut dan menampilkan data yang diperoleh melalui komunikasi serial dari mikrokontroler. Peralatan komunikasi serial terdiri atas konektor USB ke serial dan kabel serial ke serial. Sedangkan *power supply* digunakan sebagai sumber tegangan untuk motor DC.

3.3 Rangkaian perangkat keras keseluruhan

Komponen-komponen perangkat keras yang telah dirancang sebelumnya kemudian diaplikasikan menjadi sistem secara keseluruhan. Masing-masing komponen dirangkai hingga menjadi rangkaian perangkat keras keseluruhan seperti ditunjukkan pada Gambar 3.12.



Gambar 3.12 Rangkaian perangkat keras keseluruhan

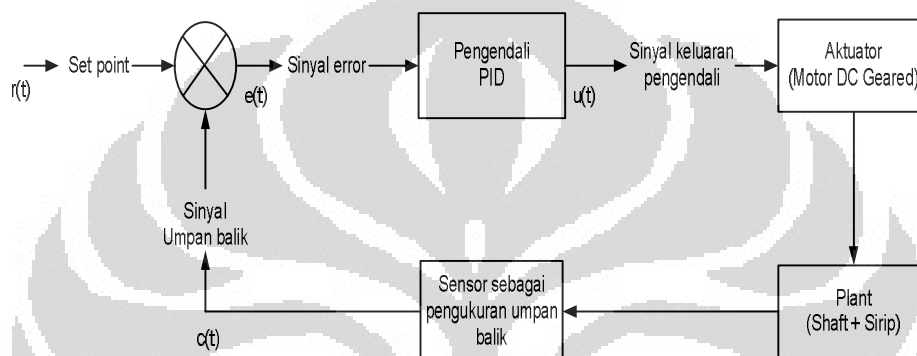
Sistem perangkat keras keseluruhan terdiri dari Maxon motor DC jenis sikat magnet permanen yang diintegrasikan dengan Maxon *planetary gear* yang memiliki reduksi kecepatan putar 246:1. Kemudian *planetary gear* akan menggerakkan *shaft* yang telah terhubung dengan *bevel gear*. Dua buah *bevel gear* yang terhubung secara L digunakan untuk menghubungkan poros *planetary gear* yang terhubung secara L digunakan untuk menghubungkan poros *planetary gear* dengan *shaft*. Kemudian *shaft* terhubung ke sirip yang terdapat dibagian atas dan sirip terhubung ke *pitch gauge* yang berfungsi sebagai penunjuk besarnya perubahan derajat. Pada bagian bawah *shaft* terdapat *rotation sensor* yang terpasang tetap pada akrilik. Sensor berguna untuk membaca pergerakan dari perubahan besar sudut dan arah sirip. Motor DC terhubung ke driver motor yang terkoneksi dengan mikrokontroler. *Power supply* yang berguna sebagai sumber tegangan motor terhubung dengan driver motor. ISP yang dihubungkan dengan PC berguna untuk memprogram mikrokontroler. Sementara port serial terhubung ke PC dengan kabel serial to usb, untuk memasukkan *setpoint* dari PC.

3.4 Perancangan kendali PID

Salah satu bentuk pengendali yang secara luas digunakan pada pengendalian proses sistem adalah pengendali PID. Persamaan pengendali ini untuk output pada dominan waktu adalah sebagai berikut.

$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right) \quad (3.2)$$

Untuk merancang sistem kontrol PID yang ideal, harus diketahui diagram dari sistem aktuator yang akan dikontrol terlebih dahulu.



Gambar 3.13 Blok diagram PID

Fungsi dari pengendali PID adalah mengendalikan sistem dengan memanipulasi sinyal *error*, sehingga respon sistem (*output*) sama dengan yang kita inginkan (*input*).

Gambar 3.13 menunjukkan bahwa pengendali PID menerima sinyal error $e(t)$ untuk kemudian diolah pada mikrokontroler ATmega16. Sinyal error $e(t)$ ini diperoleh dari hasil pengurangan nilai sensor $c(t)$ sebagai sinyal umpan balik dengan nilai set point $r(t)$. Nilai $c(t)$ diperoleh dari hasil pembacaan sensor (*rotation sensor*). Setelah pengendali PID yang terdapat pada mikrokontroler ATmega16 mengolah sinyal error $e(t)$ maka menghasilkan sinyal keluaran $u(t)$ yang lebih stabil. Kemudian selanjutnya sinyal keluaran pengendali PID $u(t)$ akan menggerakkan motor DC sesuai set point yang diberikan. Sinyal keluaran pengendali PID $u(t)$ akan menyebabkan nilai sensor $c(t)$ menjadi sama dengan set point $r(t)$. Agar lebih jelas dapat dilihat pada persamaan 3.3.

$$e(t) = r(t) - c(t) \quad (3.3)$$

Seperti yang telah diketahui bahwa pada persamaan-persamaan PID merupakan persamaan dalam kawasan waktu continuous (analog). Sedangkan

agar persamaan- persamaan tersebut dapat direalisasikan dalam bentuk pemrograman pada mikrokontroler, maka persamaan dalam kawasan waktu continuous tersebut harus didiskretisasi terlebih dahulu (kawasan digital). Berikut adalah diskretisasi menggunakan metode numerik rectangular mundur (*backward rectangular*).

Untuk memudahkan mendapatkan versi diskretnya, persamaan 3.2 di susun ulang seperti berikut:

$$CO(t) = CO_p(t) + CO_I(t) + CO_D(t) \quad (3.4)$$

Dalam hal ini,

$$CO_p(t) = K_p e(t) \quad (3.5)$$

$$CO_I(t) = \frac{K_p}{T_I} \int_0^t e(\tau) d\tau \quad (3.6)$$

$$CO_D(t) = K_p T_D \frac{de(t)}{dt} \quad (3.7)$$

1. Jika persamaan pengendali proporsional didiskretisasi:

Khusus untuk kontrol proporsional, karena bentuk persamaannya statis, versi diskretnya dapat diperoleh langsung tanpa menggunakan pendekatan apapun.

$$CO_p(k) = K_p e(k) \quad (3.8)$$

Persamaan diatas terlihat bahwa output kontrol proporsional untuk waktu cuplik tertentu (waktu cuplik ke – k) pada dasarnya hanya tergantung pada error saat pencuplikan itu saja.

2. Jika persamaan pengendali integral didiskretisasi:

$$CO_I(t) = \frac{K_p}{T_I} \int_0^t e(\tau) d\tau \approx CO_I(k) = \frac{K_p}{T_I} \sum_{i=0}^k e(i) T_c \quad (3.9)$$

atau dapat ditulis:

$$CO_I(k) = \frac{K_p T_c}{T_I} \sum_{i=0}^k e(i) = \frac{K_p T_c}{T_I} [e(0) + e(1) + \dots + e(k-1) + e(k)] \quad (3.10)$$

Persamaan 3.10 nampak bahwa output integrator pada cuplikan ke – k selain tergantung pada error saat itu, juga tergantung pada error terdahulu (waktu cuplikan sebelumnya). Akan tetapi persamaan diatas belum bisa langsung diimplementasikan dalam program, karena sistem kontrol akan membutuhkan memori yang tidak terbatas untuk menyimpan nilai error terdahulu tersebut. Dengan demikian, agar output kontrol integral dapat dihitung secara efisien, implementasinya dalam program harus direalisasikan secara rekursif.

$$CO_I(k-1) = \frac{K_p T_c}{T_I} \sum_{i=0}^{k-1} e(i) = \frac{K_p T_c}{T_I} [e(0) + e(1) + \dots + e(k-1) + e(k)] \quad (3.11)$$

Dengan mensubstitusikan persamaan 3.11 ke dalam persamaan 3.10, sekarang output integrator dapat dirumuskan seperti nampak pada persamaan 3.12.

$$CO_I(k) = CO_I(k-1) + \frac{K_p T_c}{T_I} e(k) \quad (3.12)$$

berdasarkan persamaan 3.12 di atas, terlihat bahwa output integrator pada waktu cuplik ke-k sekarang dapat dihitung berdasarkan error saat itu ditambah dengan sinyal output integrator waktu cuplik sebelumnya.

3. Jika persamaan pengendali derivatif didiskretisasi:

Untuk kontrol diferensial, menggunakan metode *backward difference*, output pada waktu cuplik ke-k dapat dihitung berdasarkan selisih error saat itu dengan error cuplikan sebelumnya.

$$CO_D(k) = K_p T_D \frac{e(k) - e(k-1)}{T_c} \quad (3.13)$$

Sehingga, persamaan PID ideal versi diskret pada sistem digital akan menjadi:

$$CO(k) = CO_p(k) + CO_I(k) + CO_D(k) \quad (3.14)$$

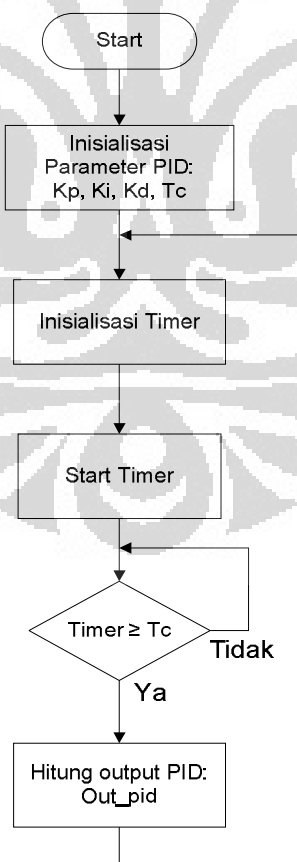
dalam hal ini:

$$CO_p(k) = K_p e(k)$$

$$CO_i(k) = CO_i(k-1) + \frac{K_p T_c}{T_i} e(k)$$

$$CO_d(k) = K_p T_D \frac{e(k) - e(k-1)}{T_c}$$

Dari persamaan 3.14 sudah berupa persamaan digital yang telah didiskretisasi, maka persamaan tersebut sudah dapat diaplikasikan kedalam bahasa pemrograman. Implementasi bahasa pemrograman pada skripsi ini menggunakan bahasa pemrograman *Basic Compiler* (Bascom). Sebelumnya harus ditentukan terlebih dahulu nilai set point dan waktu sampling. Serta melakukan pembacaan nilai dari sensor dengan lamanya periode sesuai waktu sampling. Misalkan set point bernilai 10 dan lamanya waktu sampling (T_c) 100 *microseconds*. Dan untuk mendapatkan nilai sensor dapat dilakukan dengan pembacaan ADC, dan untuk mensampling nilai ADC selama 100 *microseconds* dapat menggunakan fitur *timer* yang sudah ada pada Basic compiler.



Gambar 3.14 Diagram alir kontrol PID digital

Tampak pada diagram alir kontrol PID digital, dalam program utama fungsi `Out_pid` harus selalu dipanggil setiap waktu cuplik yang telah ditentukan sebelumnya. Untuk fungsi inisialisasi parameter PID, pemanggilan dalam program utama dilakukan satu kali, yaitu sebelum masuk pada looping pengontrolan.

Untuk menentukan waktu sampling (T_c) dapat menggunakan teorema *sampling shannon*. Idealnya waktu sampling adalah 0 detik sehingga tidak ada satupun data yang tidak di sampling, namun hal itu tidak akan tercapai karena untuk mencapai jeda sampling 0 detik memerlukan frekuensi sampling yang dapat dihitung dengan persamaan 3.15

$$f_c = \frac{1}{T_c} \quad (3.15)$$

$$f_c = \frac{1}{0} = \infty$$

Sementara f_c harus mempunyai nilai tertentu agar dapat direalisasikan. Teorema *sampling shannon* telah menjabarkan secara matematik pada bidang waktu bahwa:

$$T_c \leq 0.1T_r \quad (3.16)$$

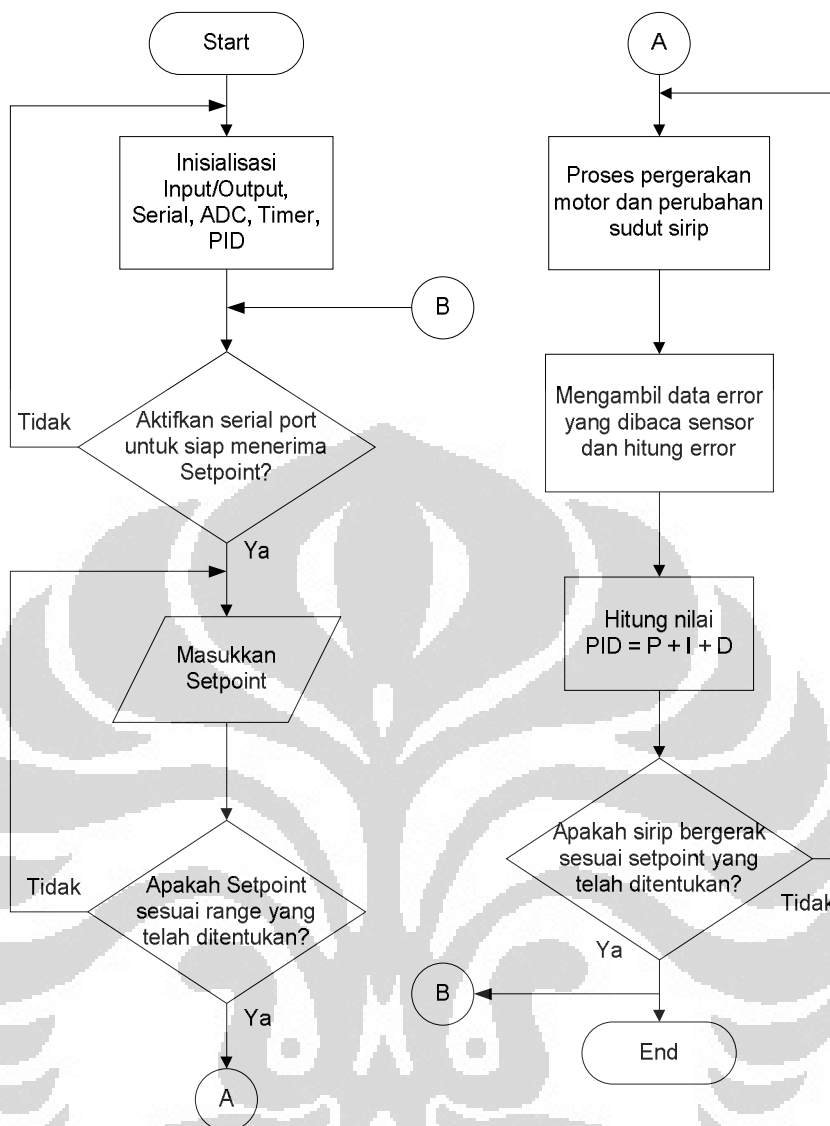
Dimana nilai T_r merupakan waktu naik.

Perpanjangan rentang waktu sampling mengakibatkan ketidakstabilan dan kehilangan data. Penurunan rentang waktu sampling mengakibatkan perubahan antara nilai-nilai sampling akan menurun, dan ini mengakibatkan penurunan resolusi yang pada akhirnya menjadikan data yang diambil kurang akurat.

3.5 Perancangan perangkat lunak

Pada perancangan perangkat lunak, penulisan program menggunakan bahasa Basic dengan menggunakan software Bascom (Basic compiler) dari AVR. Kemudian program yang telah dibuat di-*compile* hingga menjadi file dalam format .Hex. File .Hex inilah yang di-*write* ke mikrokontroler dengan menggunakan program AvrOsp.

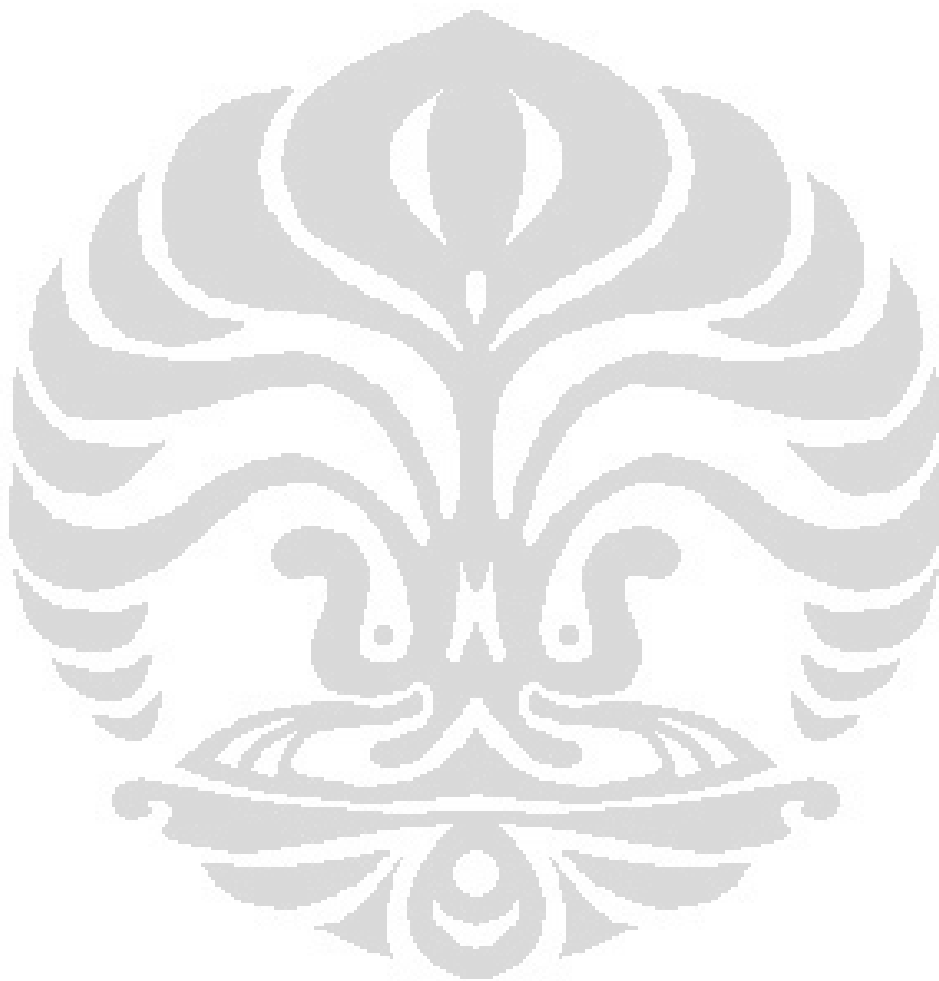
Program yang dijalankan pada mikrokontroler, seperti ditunjukkan pada diagram alir pada Gambar 3.15.



Gambar 3.15 Diagram alir perangkat lunak

Pada awal program dimulai dengan inialisasi program yang terdiri atas inialisasi input/output, inialisasi serial, inialisasi ADC, inialisasi timer, dan inialisasi PID. Untuk memulai program akan meminta input inialisasi untuk mengaktifkan port serial dengan memasukkan kode (@). Jika kode telah sesuai maka program akan meminta untuk memasukkan setpoint besaran sudut. Jika sudut yang dimasukkan sesuai dengan *range* yang telah ditentukan (-10 sampai dengan 10) maka motor akan berputar dan sirip akan bergerak, dari pergerakan tersebut akan menghasilkan error. Kemudian error tersebut akan dibaca oleh sensor. Data error yang diterima akan diolah oleh mikrokontroler yang telah diprogram pengendali PID. Nilai Kp, Ki, dan Kd didapatkan dari penalaan konstanta PID yang telah dijelaskan sebelumnya. Proses selanjutnya

adalah pengaturan kecepatan putar motor dengan mengatur nilai *duty cycle* yang dilakukan oleh mikrokontroler. Sehingga motor akan berputar dan menggerakkan sirip sesuai setpoint yang diinginkan.



BAB 4

PENGUJIAN DAN ANALISIS HASIL

Pada bab ini menjelaskan seluruh hasil pengujian dan analisis terhadap prototipe dan algoritma yang telah dirancang pada bab sebelumnya. Serta membahas tentang hasil dari prototipe yang telah dibuat, apakah alat tersebut telah bekerja sesuai dengan yang diinginkan.

Tujuan pengujian ini adalah untuk mengetahui tanggapan/respon dari pergerakan sirip terhadap pengendalian PID yang diberikan. Pengaturan besar sudut sirip dengan menggunakan motor DC sikat magnet permanen yang dikombinasikan dengan *planetary gearhead* dikendalikan dengan menggunakan teknik PID metode Ziegler Nichols.

4.1 Penggunaan alat ukur pada pengujian

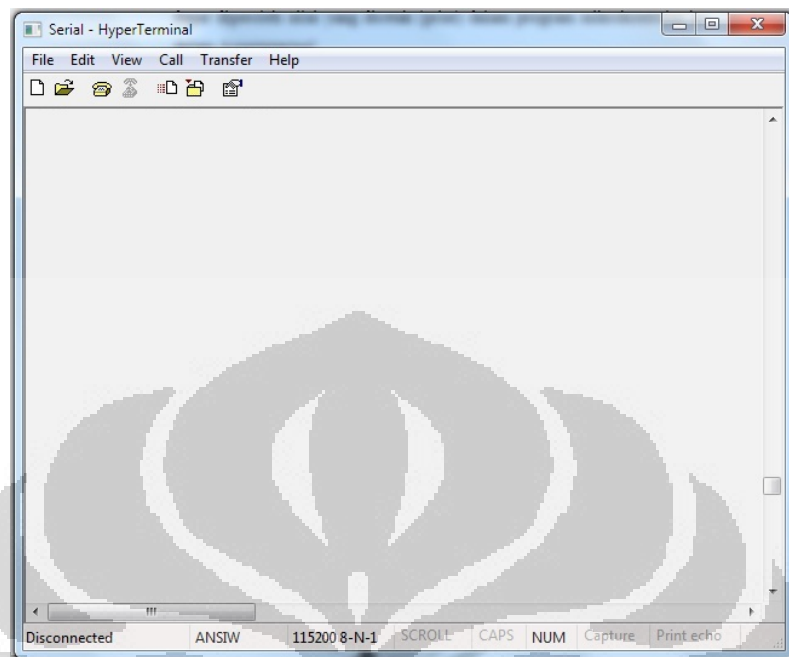
Pada pengujian prototipe yang telah dibuat digunakan alat ukur tegangan yang berupa multimeter digital. Berikut adalah gambar multimeter yang digunakan.



Gambar 4.1 Multimeter digital DT-830B

Agar dapat membaca nilai yang telah dikonversi oleh mikrokontroler dari umpan balik *rotation sensor* digunakan aplikasi *hyperterminal* pada komputer, dengan komunikasi secara serial dari mikrokontroler. Sehingga dapat diperoleh

nilai yang dihasilkan oleh program mikrokontroler ke aplikasi *hyperterminal*. *Hyperterminal* juga digunakan sebagai masukan *setting point* besarnya sudut.



Gambar 4.2 Tampilan aplikasi *hyperterminal*

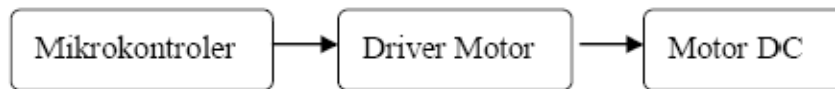
Alat ukur yang digunakan selanjutnya adalah *oscilloscope portable* yang terhubung dengan port USB komputer. *Oscilloscope portable* digunakan untuk mendapatkan grafik sinyal keluaran dari driver motor DC.



Gambar 4.3 PC *Oscilloscope* DSO-2090 USB

4.2 Pengujian driver motor DC L298

Pengujian driver motor ini dilakukan untuk memastikan bahwa motor DC sebagai penggerak utama pada prototipe sirip roket telah berkerja dengan baik. Berikut adalah blok diagram pengujian driver motor DC.



Gambar 4.4 Blok diagram pengujian motor DC

Driver motor DC L298 digunakan sebagai pemicu dari arah perputaran motor DC. Pada L298 memiliki dua buah pin enable, empat buah pin input, dan empat buah pin output yang mana memungkinkan untuk mengontrol dua buah motor DC. Namun pada rangkaian driver kali ini hanya dipakai satu buah motor DC, jadi pin yang dipakai cukup satu buah enable (EN A di PIN 6), dua buah pin input (IN 1 di PIN 5 dan IN 2 di PIN 7), dan dua buah pin output (OUT 1 di PIN 2 dan OUT 2 di PIN 3) yang dihubungkan dengan motor DC.

Pengujian dilakukan dengan cara memberi inputan berupa logika “1” dan “0” pada masing-masing pin inputan secara bergantian kemudian melihat perubahan pada pergerakan motor yang terlebih dahulu sudah disambungkan pada pin-pin output dari driver. Berikut adalah tabel hasil pengujian rangkaian driver motor L298.

Tabel 4.1 Hasil pengujian rangkaian driver motor DC L298

Enable A	Input 1	Input 2	Tegangan input (dari mikrokontroler)	Kondisi motor	Tegangan output (ke motor DC)
0	0	0	0 V	Diam	0 V
0	0	1	0 V	Diam	0 V
0	1	0	0 V	Diam	0 V
0	1	1	0 V	Diam	0 V
1	0	0	0 V	Diam	0 V
1	0	1	4.90V (Input 2)	CCW (ke arah -10)	11.09 V
1	1	0	4.90 V (Input 1)	CW (ke arah 10)	11.10 V
1	1	1	0 V	Diam	0 V

Dari hasil pengujian driver motor DC yang ditunjukkan pada Tabel 4.1 bahwa motor DC akan bekerja ketika pin Enable A diberi logika “1” dan input diberi logika saling berkebalikan. Tegangan output dihasilkan dari *power supply* untuk menggerakkan motor DC. Berikut adalah potongan program pada Bascom untuk menggerakkan motor DC.

```

'Putaran Motor CW ke arah 10 deg
Do
If PinD.2 = 0 Then
Portc.0 = 1
Portc.1 = 0
Portc.4 = 1
End If
'Putaran Motor CCW ke arah -10 deg
If PinD.3 = 0 Then
Portc.0 = 0
Portc.1 = 1
Portc.4 = 1
End If
'Putaran Motor OFF
If PinD.2 = 0, PinD.3 = 0 Then
Portc.0 = 0
Portc.1 = 0
Portc.4 = 0
End If
Loop

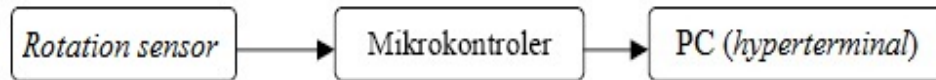
```

Dalam pengujian yang dilakukan, motor DC di program untuk berputar dengan kecepatan penuh, yaitu tidak dikendalikan dengan pengendali PWM ataupun PID. Motor berputar CW ketika pin D.2 diberikan logika “0” dan motor akan berputar CCW ketika pin D.3 diberikan logika “0”. Motor akan berhenti berputar ketika pin D.2 dan pin D.3 diberikan logika “0”. Dengan program ini driver motor dapat berjalan dengan baik pada arah CW, CCW, dan berhenti.

4.3 Pengujian *rotation sensor* sebagai *feedback*

Melakukan pengujian pada sensor posisi dengan menggunakan *rotation sensor* linear yang dihubungkan menggunakan pin ADC mikrokontroler. Pada pengujian *rotation sensor* menggunakan pengolahan data ADC dari pengukuran dengan pembagi tegangan. Sensor ini memiliki putaran dari 0^0 sampai dengan 300^0 , output sensor ini adalah resistor variabel yang bernilai $9,8k\Omega$ untuk putaran

maksimal (dari hasil pengukuran). Untuk mengendalikan motor agar putarannya dapat mencapai sudut yang ditentukan, maka dapat ditentukan nilai perubahan ADC tiap sudutnya sebagai parameter yang nantinya akan digunakan sebagai umpan balik.



Gambar 4.5 Blok diagram pengujian *rotation sensor*

Keluaran dari *rotation sensor* dimasukkan ke port A.6 untuk kemudian dikonversi menjadi nilai digital. Tegangan referensi ADC dapat dipilih antara lain pada pin AREF, pin AVCC atau menggunakan tegangan referensi internal sebesar 2.56 Volt. Pada pengujian ini tegangan referensi ADC dipilih pada pin AVCC. Agar fitur ADC mikrokontroler dapat digunakan maka ADEN (ADC Enable, dalam I/O register ADCSRA) harus diaktifkan (Start ADC). Untuk melihat nilai yang didapat atau dihasilkan oleh mikrokontroler dapat dilihat menggunakan aplikasi *hyperterminal* yang telah disediakan oleh *windows*. Mikrokontroler dihubungkan dengan komputer melalui komunikasi serial. Oleh karena itu, port komunikasi dan *baudrate* yang digunakan pada mikrokontroler disamakan pada *hyperterminal*. Langkah selanjutnya adalah mengirim data untuk ditampilkan ke PC agar dapat didefinisikan ke program selanjutnya. Berikut ini adalah potongan program dalam melakukan pengujian.

```

$regfile = "m16def.dat"           ' specify the used micro
$baud = 115200                    ' use baudrate
Config Adc = Single , Prescaler = Auto , Reference = Avcc
Start Adc
Nilai_adc = 6
Do
Nilai = Getadc(Nilai_adc)
Print Nilai
Loop
End
  
```

Hasil yang didapat pada aplikasi *hyperterminal* dapat dilihat pada tabel 4.2. Dari tabel tersebut didapat nilai yang telah dikonversi menjadi ADC oleh mikrokontroler pada sudut tertentu.

Data hasil *rotation sensor* pada setiap perubahan sudut ditunjukkan pada Tabel 4.2 sebagai berikut:

Tabel 4.2 Data ADC pada setiap perubahan sudut

Sudut	Data ADC	Tegangan input ADC (Volt)	
		Hasil pengukuran	Hasil perhitungan
10°	395	1,93	1,92
9°	404	1,98	1,97
8°	413	2,02	2,01
7°	421	2,05	2,05
6°	430	2,11	2,10
5°	439	2,15	2,14
4°	447	2,18	2,18
3°	455	2,23	2,22
2°	463	2,26	2,26
1°	472	2,31	2,30
0°	480	2,35	2,34
-1°	489	2,38	2,38
-2°	498	2,43	2,43
-3°	507	2,48	2,47
-4°	516	2,51	2,51
-5°	525	2,57	2,56
-6°	534	2,61	2,60
-7°	543	2,65	2,65
-8°	552	2,69	2,69
-9°	561	2,74	2,73
-10°	570	2,79	2,78

Data hasil perhitungan tegangan input ADC diperoleh menggunakan rumus pada persamaan 2.4. V_{ref} yang digunakan adalah 4,99 Volt (hasil pengukuran). Di bawah ini adalah tegangan input pada 0 derajat:

$$ADC = \frac{V_{in}}{V_{ref}} \times 1024; 480 = \frac{V_{in}}{4,99} \times 1024; V_{in} = 2,34$$

Nilai tegangan yang dihasilkan dari sensor posisi diterima oleh mikrokontroler dengan cara keluaran *rotation sensor* dimasukkan ke pin ADC mikrokontroler dan menghasilkan keluaran tegangan rata-rata. Data dari pengujian sensor menghasilkan perbedaan nilai antara pengukuran tegangan dan perhitungan tegangan pada setiap sudutnya. Dari perbedaan ini menghasilkan nilai % error.

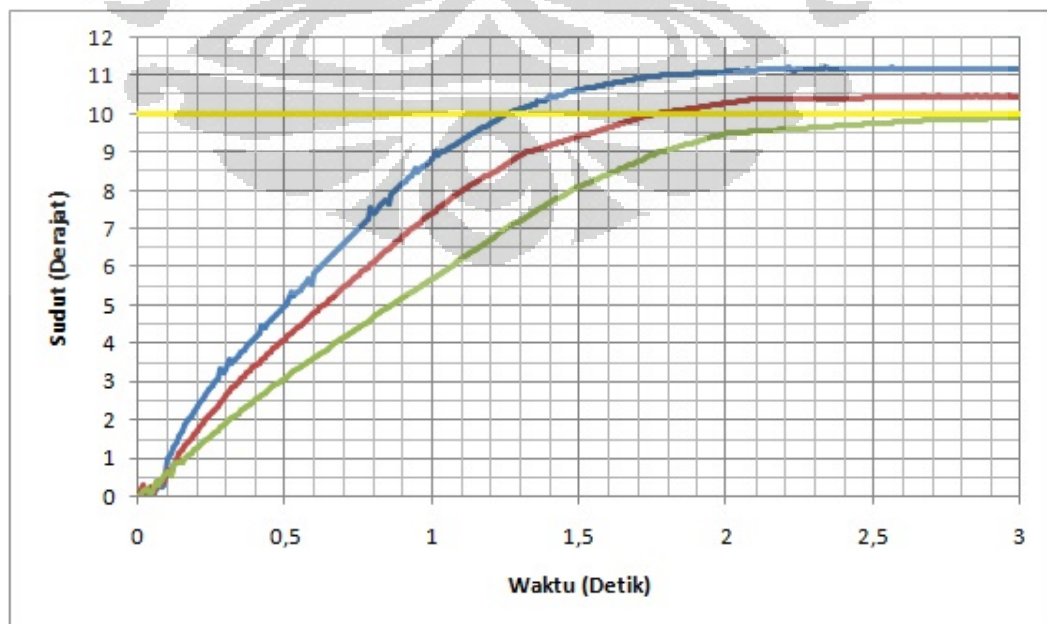
$$\% \text{ error} = \frac{|V_{pengukuran} - V_{perhitungan}|}{V_{perhitungan}} \times 100\%$$

$$\% \text{ error} = \frac{|2,35 - 2,34|}{2,34} \times 100\%$$

$$\% \text{ error} = 0,43\%$$

4.4 Pengujian sistem dengan input *duty cycle*

Pada pengujian sistem ini, dilakukan dengan memberikan input *duty cycle* pada motor. Prosesnya terjadi hanya sekali dengan memberikan nilai setpoint sebesar 10^0 . Pada Gambar 4.6 berikut ini merupakan grafik respon untuk *duty cycle* 50%, 70%, dan 100% terhadap waktu dan sudut yang diberikan.



Gambar 4.6 Grafik pergerakan sirip dengan input *duty cycle*

Keterangan grafik:

Setpoint = Garis kuning

Duty cycle 100% = Garis biru

Duty cycle 70% = Garis merah

Duty cycle 50% = Garis hijau

Dari Gambar 4.6 menunjukkan bahwa, pergerakan dari sirip agak sulit untuk mencapai sudut yang telah ditentukan karena tidak adanya umpan balik (*feedback*) yang akan mengendalikan keluaran. Dengan memberikan *duty cycle* yang semakin besar maka waktu untuk mencapai sudut yang ditentukan semakin cepat namun akan semakin memperbesar error yang terjadi terhadap sudut yang telah ditentukan. Dengan memberikan *duty cycle* yang semakin kecil maka error akan semakin kecil juga namun memperlambat respon untuk mencapai sudut yang telah ditentukan. Error yang melebihi setpoint dikarenakan pergerakan yang cepat tetapi tidak adanya pengendalian keluaran untuk melakukan penyesuaian terhadap setpoint.

Presentase error yang terjadi pada pemberian *duty cycle* 100% adalah

$$\% \text{ error} = \frac{\text{Sudut error} - \text{setpoint}}{\text{setpoint}} \times 100\%$$

$$\% \text{ error} = \frac{11.2 - 10}{10} \times 100\% = 12\%$$

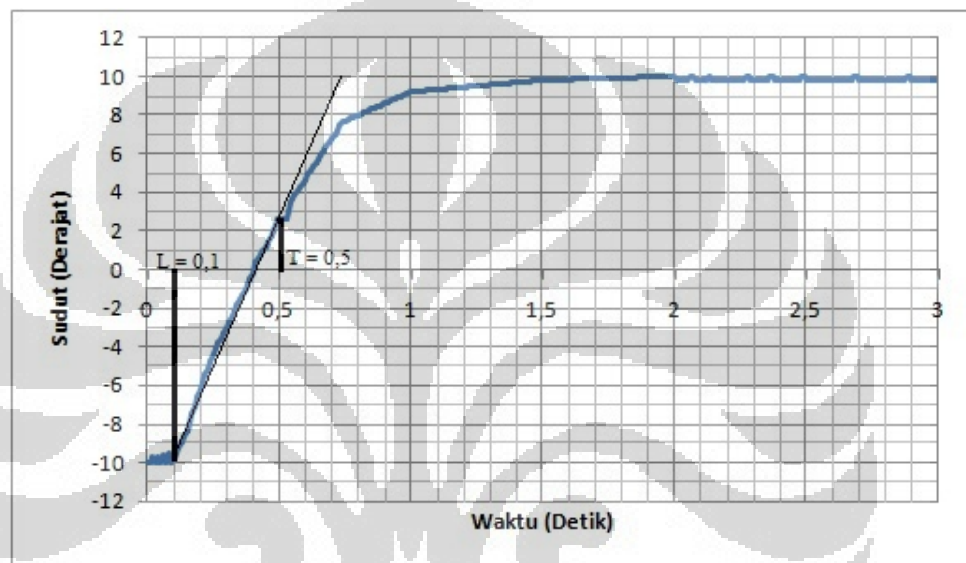
Grafik respon menunjukkan bahwa pergerakan dari sirip masih kurang sempurna dan kurang memuaskan sebab sistem ini membutuhkan waktu penetapan (*settling time*) yang lama jadi respon sistem ini kurang praktis. Jadi yang diinginkan adalah ketika setpoint diberikan, sirip harus menuju ke setpoint secepat mungkin dan sistem mencapai keadaan tunak dengan waktu yang cepat. Oleh karena itu, perlu adanya pengontrol yang sesuai untuk mengontrol sistem tersebut guna mengatur waktu penetapan (*settling time*) dengan overshoot sekecil mungkin.

4.5 Pengujian sistem *close loop*

Untuk mendapatkan kinerja sistem kontrol yang memiliki performansi yang handal dan kualitas pengendalian yang optimal, maka parameter pada kontroler juga harus memiliki nilai yang optimal.

4.5.1 Penalaan pengendali PID dengan metode Ziegler Nichols

Berdasarkan metode tuning PID Ziegler Nichols, cara mengambil garis singgung adalah dengan membuat garis lurus dan menghimpit pada respon yang memiliki nilai linier paling banyak dan harus melalui nilai τ yaitu waktu yang diperlukan oleh respon untuk mencapai 63% dari nilai steady state respon. Dengan memberikan tegangan pada motor maka kecepatannya dapat diketahui dengan *rotation sensor*, dan grafik respon terhadap waktu ditunjukkan pada gambar berikut:



Gambar 4.7 Grafik pergerakan sirip tanpa pengendali PID

Dari grafik pergerakan sudut tersebut dapat diketahui nilai-nilai parameter yang digunakan untuk menghitung nilai parameter PID yang diinginkan, seperti nilai keterlambatan transportasi (L) dan konstanta waktu proses (T). Diketahui nilai keterlambatan waktu proses $L = 0.1$ detik, nilai konstanta waktu $T = 0.5$ detik, dan gain statis proses $K = 0.54^0/\%$.

$$K = \frac{\Delta PV}{\Delta CO}$$

$$K = \frac{(10 - (-10))^0}{(100 - 60)\%}$$

$$K = 0.54^0 / \%$$

Artinya, setiap perubahan sinyal output kontroler (CO) sebesar 1%, variabel proses yang dikontrol akan mengalami perubahan sudut sebesar $0,54^0$.

$$T = 63\% \times \Delta PV + (PV_0)$$

$$T = 12.6^0 + (-10)$$

$$T = 2.6^0 \approx \text{berada pada } 0.5 \text{ detik}$$

Dari nilai parameter yang telah diketahui tersebut, maka dapat dihitung nilai K_p , K_I , dan K_D dari sistem tersebut.

1. Nilai K_p

$$K_p = \frac{1.2T}{KL}$$

$$K_p = \frac{1.2 \times 0.5}{0.54 \times 0.1}$$

$$K_p = 11.1$$

2. Nilai K_I

$$K_I = \frac{K_p}{T_I}$$

$$T_i = 2L = 2 \times 0.1 = 0.2$$

$$K_I = \frac{11.1}{0.2}$$

$$K_I = 55.5$$

3. Nilai K_D

$$K_D = K_p \times T_D$$

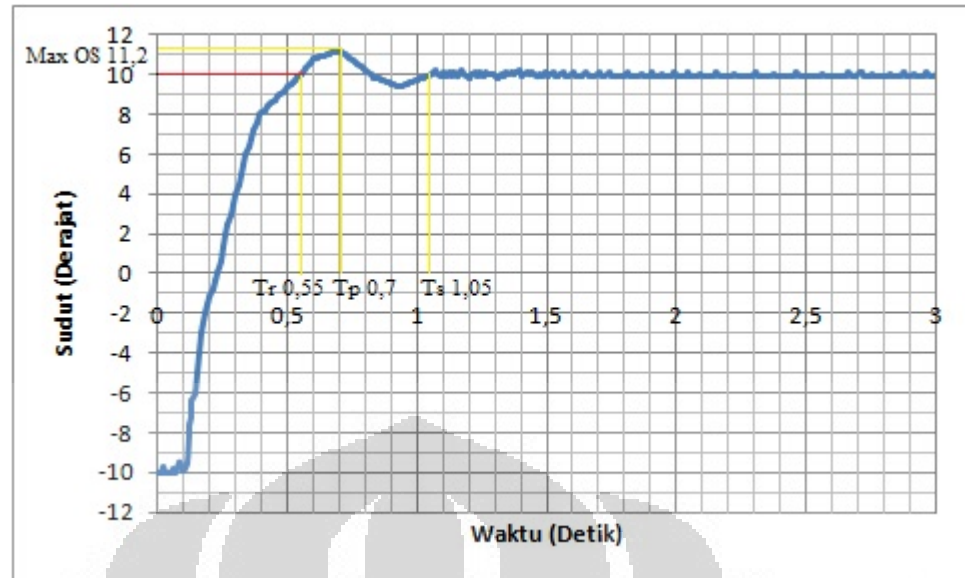
$$K_D = K_p \times (0.5 \times L)$$

$$K_D = 11.1 \times (0.5 \times 0.1)$$

$$K_D = 0.555$$

4.5.2 Pengujian pengendali PID pada sirip

Melakukan pengujian pengendali PID untuk mengontrol posisi sudut sirip. Kontrol ini pada dasarnya adalah kontrol kecepatan putar pada motor DC dan untuk mengendalikan error. Pergerakan sirip yang menggunakan pengendali PID diharapkan akan memberi respon sistem yang lebih baik dari respon sistem pergerakan sirip tanpa menggunakan pengendali PID. Gambar 4.8 dibawah ini adalah grafik pergerakan sirip yang diberikan pengendali PID.



Gambar 4.8 Grafik pergerakan sirip dengan pengendali PID

Respon sistem terbaik diperoleh untuk nilai $K_p = 11$, $K_i = 60.5$, dan $K_d = 0.50$, dimana respon sistem sedikit terjadi lonjakan akan tetapi mampu mencapai keadaan tunak pada $t = 1.05$ detik. Dari Gambar 4.8 di peroleh settling time sekitar 1.05 detik, overshoot tereduksi lebih kecil dari 12%, dan kesalahan keadaan tunak mendekati nol. Jadi, dapat disimpulkan bahwa dengan menggunakan pengendali PID dengan nilai $K_p = 11$, $K_i = 60.5$, dan $K_d = 0.50$ hampir semua permintaan desain yang diinginkan terpenuhi dengan baik atau dapat dikatakan bahwa terjadi nilai kestabilan yang stabil pada sistem sehingga ketika setpoint diberikan sistem dapat mencapai keadaan tunak dengan waktu yang cepat. Nilai-nilai dari parameter K_p , K_i , dan K_d pada program PID ditentukan dengan menggunakan metode *trial and error* yang mengacu pada metode penalaan Ziegler Nichols.

BAB 5

PENUTUP

Pada bab ini berisi rangkuman keseluruhan dari inti penelitian yang telah dilakukan, serta mengacu pada analisis dari hasil yang telah diperoleh. Bab ini berisi kesimpulan dan saran dari hasil pengujian.

4.6 Kesimpulan

Setelah melakukan pengujian dan analisis, maka dapat diambil beberapa kesimpulan mengenai sistem kerja dari prototipe yang telah dibuat, yaitu:

1. Prototipe ini dirancang dan dibuat menggunakan aktuator yang terdiri dari *brushed* DC motor dan *planetary gear* yang mempunyai torsi sebesar 6,75 Nm dan reduksi kecepatan 246 : 1.
2. Dari hasil pengujian diperoleh respon dari motor DC yang lebih cepat dan lebih stabil dengan menggunakan pengendali PID daripada hanya memberikan *duty cycle*.
3. Dari uji coba setting point yang diberikan pada *plant* menghasilkan pergerakan sudut dari sirip sebesar 1° dengan jarak -10° sampai 10° . Perubahan sudut dapat dilihat melalui *pitch gauge*.
4. Dari beberapa pengujian pada sistem dengan variasi parameter K_p , K_i , dan K_d yang berbeda-beda, didapatkan bahwa respon terbaik sistem diperoleh pada saat $K_p=11$, $K_i=60.5$, dan $K_d=0.50$ yang menunjukkan tanggapan sistem dapat mencapai kestabilan dan tidak mengalami lonjakan yang berarti.
5. Dari hasil pengujian didapatkan settling time sebesar 1.05 detik, overshoot tereduksi lebih kecil dari 12%, dan kesalahan keadaan tunak mendekati nol.
6. Untuk menentukan nilai parameter PID (K_p , K_i , K_d) dapat menggunakan metode *trial and error* dengan mengacu pada metode-metode yang berlaku pada kontrol PID, pada tugas akhir ini digunakan metode Ziegler Nichols sebagai penalaan.

7. Pembuatan mekanik yang masih banyak kekurangan, sehingga dapat menyebabkan perbedaan antara pengukuran dengan perhitungan.

4.7 Saran

Hasil dari pengujian prototipe ini ada beberapa hal yang perlu diperbaiki untuk pengembangan selanjutnya, diantaranya:

1. Memperbaiki pembuatan mekanik dari segi kepresisian sirip yang belum kuat jika diberi beban karena bentuk badan dari prototipe yang tidak mendukung sehingga diharapkan perancangan mekanik dapat lebih baik.
2. *Rotation sensor* menghasilkan nilai yang terus berosilasi sebesar ± 1 dari nilai aktualnya walaupun sirip dalam keadaan diam. Hal ini dapat mempengaruhi pemrograman dan juga terhadap respon motor saat bekerja. Pada penelitian selanjutnya diharapkan menggunakan sensor yang lebih presisi seperti encoder atau jenis *rotary sensor* lain yang lebih baik.
3. Bentuk motor DC dan *planetary gear* yang digunakan terlalu panjang sehingga menyebabkan agak sulitnya pembuatan mekanik dari prototipe. Diharapkan penelitian selanjutnya dapat dipergunakan tipe motor DC dan *planetary gear* yang lebih kecil, *compact*, dan dengan performa yang lebih mantap.

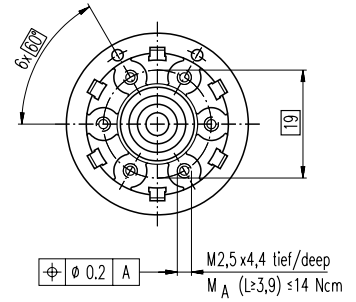
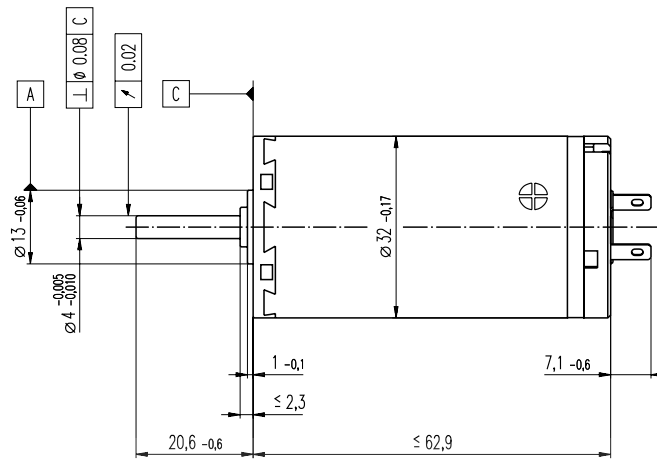
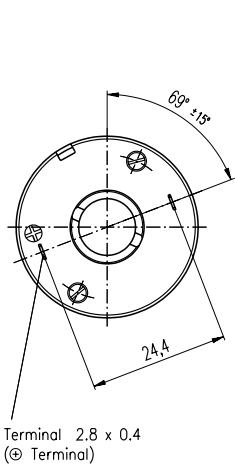
DAFTAR REFERENSI

- [1] Ristanovic, Lazic and Indin. (2006). *Nonlinear PID controller modification of the electromechanical actuator system for aerofin control with a PWM controlled DC motor*. Serbia: University of Belgrade.
- [2] Howe D et al. (2003). *Analysis of a spherical permanent magnet actuator*. Journal of applied physics, vol 81.
- [3] Staff UI. (2010). Makalah motor DC. Diambil tanggal 2 November 2011 dari <http://staff.ui.ac.id/internal/040603019/material/makalahMotorDC.doc>
- [4] SGS-THOMSON. (1995). Datasheet L298 dual full-bridge driver. Microelectronics.
- [5] Dfrobot. (2010). *Rotation sensor*. Diambil tanggal 8 Oktober 2011 dari <http://www.robotshop.com/dfrobot-rotation-sensor-v1.html>
- [6] Chan, Yefri. (2010). Teori dasar roda gigi. Diambil tanggal 15 November 2011 dari <http://yefrichan.files.wordpress.com/2010/05/teori-dasar-rodagigi.doc>
- [7] Ylwols. (2010). *My life & mechanical*. Diambil tanggal 15 November 2011 dari <http://mylifemechanical.blogspot.com>.
- [8] Motor, Maxon. (2010). *Planetary gearhead*. Diambil tanggal 17 November 2011 dari <http://uk.farnell.com/maxon-motors/166174/gearhead-planetary-246-1-ratio/dp/1761297>
- [9] PPPGT VEDC. (2000). Casis dan transmisi. Malang.
- [10] ATMEL. (2010). *ATMEGA16 8 bit microcontroller AVR with 16K Bytes In-System programmable flash*. USA: Atmel Corporation.
- [11] Wikipedia. (2011). Pengendali PID. Diambil tanggal 12 November 2011 dari <http://id.wikipedia.org/wiki/PID>
- [12] Iswanto. (2008). Design dan implementasi sistem embedded mikrokontroler ATmega8535 dengan bahasa Basic. Yogyakarta: Gava Media.
- [13] Setiawan, Iwan. (2008). Kontrol PID untuk proses industri. Jakarta: PT Elex Media Komputindo.
- [14] Lapan. (2012). Raket kendali RXX 200. Diambil tanggal 19 Januari 2012 dari <http://www.lapan.go.id>



LAMPIRAN

A-max 32 Ø32 mm, Graphite Brushes, 15 Watt



- Stock program
- Standard program
- Special program (on request!)

Order Number

236643 236644 236645 236646 236647 236648 236649 236650

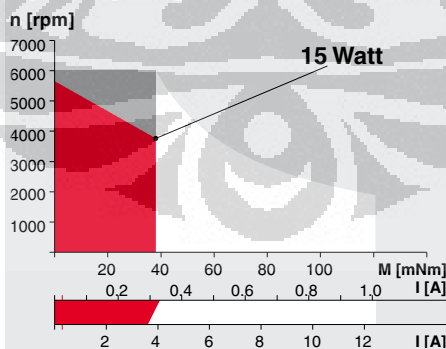
Motor Data

		236643	236644	236645	236646	236647	236648	236649	236650
1 Assigned power rating	W	15	15	15	15	15	15	15	15
2 Nominal voltage	Volt	6.0	9.0	12.0	18.0	24.0	30.0	36.0	48.0
3 No load speed	rpm	5710	4830	4590	5200	5860	5800	5760	3800
4 Stall torque	mNm	100.0	88.2	96.8	119	129	129	126	82.1
5 Speed/torque gradient	rpm/mNm	59.5	57.0	49.1	45.0	46.6	45.9	46.6	47.9
6 No load current	mA	295	163	115	87	74	59	49	24
7 Starting current	mA	10500	5160	4010	3690	3370	2680	2170	704
8 Terminal resistance	Ohm	0.573	1.75	2.99	4.88	7.13	11.2	16.6	68.2
9 Max. permissible speed	rpm	6000	6000	6000	6000	6000	6000	6000	6000
10 Max. continuous current	mA	3610	2070	1580	1240	1020	817	671	331
11 Max. continuous torque	mNm	34.6	35.5	38.2	39.8	39.1	39.4	39.1	38.6
12 Max. power output at nominal voltage	mW	13900	10600	11200	15700	19200	19200	18600	7890
13 Max. efficiency	%	67	67	68	71	72	73	72	67
14 Torque constant	mNm/A	9.59	17.1	24.1	32.2	38.2	48.3	58.3	117
15 Speed constant	rpm/V	996	559	396	297	250	198	164	81.9
16 Mechanical time constant	ms	26	22	21	21	20	20	20	20
17 Rotor inertia	gcm ²	42.4	37.4	41.4	43.8	41.9	42.1	41.3	40.0
18 Terminal inductance	mH	0.07	0.21	0.42	0.74	1.05	1.67	2.44	9.75
19 Thermal resistance housing-ambient	K/W	7.5	7.5	7.5	7.5	7.5	7.5	7.5	7.5
20 Thermal resistance rotor-housing	K/W	2.1	2.1	2.1	2.1	2.1	2.1	2.1	2.1
21 Thermal time constant winding	s	17	15	17	18	17	17	17	16

Specifications

- Axial play 0.12 - 0.22 mm
- Max. **ball bearing** loads
axial (dynamic) 7.6 N
radial (5 mm from flange) 32.0 N
Press-fit force (static) 110 N
- Max. **sleeve bearing** loads
axial (dynamic) 5 N
radial (5 mm from flange) 10.5 N
Press-fit force (static) 110 N
- Radial play **ball bearings** 0.025 mm
- Radial play **sleeve bearings** 0.012 mm
- Ambient temperature range -20/+85°C
- Max. rotor temperature +125°C
- Number of commutator segments 13
- Weight of motor 211 g
- Values listed in the table are nominal. For applicable tolerances (see page 43) and additional details please request out of the maxon selection program on the settled CD-Rom.
- Connector from S 2332... (see page 89)

Operating Range



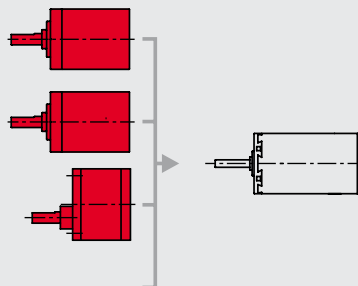
Comments

Details on page 49

- Recommended operating range**
- Continuous operation**
In observation of above listed thermal resistances (lines 19 and 20) the maximum permissible rotor temperature will be reached during continuous operation at 25°C ambient. = Thermal limit.
- Short term operation**
The motor may be briefly overloaded (recurring).
- 110870** Motor with high resistance winding
- 110863** Motor with low resistance winding

maxon Modular System

- Planetary Gearhead**
Ø32 mm
0.75-4.5 Nm
Details page 188
- Planetary Gearhead**
Ø32 mm
1.0-6.0 Nm
Details page 191
- Spur Gearhead**
Ø38 mm
0.1-0.6 Nm
Details page 194



- Options: Sleeve bearings in place of ball bearings and pigtails in place of terminals.

maxon gear – Important considerations

When selecting a gear-motor, please keep the following suggestions in mind: Reduce the motor speed by dividing by the reduction ratio. Increase the motor torque by multiplying times gearhead ratio times gearhead efficiency. The maximum efficiency is stated in the table. It is dependent on the number of stages and is measured at maximum continuous torque and nominal speed. The efficiency is reduced for lower torque requirements. This must be taken into consideration when selecting a gear-motor. To keep noise and gearhead wear low it is advisable not to exceed the max. recommended input speed. Also, the permissible output torque must be observed with a clear distinction between short-term and continuous operation. We will gladly assist you with any gearmotor problem you might have.

Important:

- Primary consideration should be given speed and torque at the output shaft when determining a gear-motor combination.
- An extensive adaption to desired output requirements is achieved by taking advantage of the various windings available with maxon motors as well as the wide selection of gear reduction ratios.
- The recommended maximum gearhead input speed should, if possible, not be exceeded (see gearhead pages, «Technical Data»).
- When ordering a gearhead, state maxon motor or give details of the mounting.
- You will find spur and planetary gearheads on page 104 up to 134 sorted by diameter.

Spur gearheads:

Output torques up to 0.6 Nm.
Reduction ratios from 4:1 to 3000:1.
One or several stages. Each pair of gear wheels represents one stage, whereas the first gear wheel (pinion) is mounted on the motor shaft.
Normally, gearhead output shaft bearings are the sinter sleeve type.



Planetary gearheads:

Output torques up to 180 Nm.
Reduction ratios from 4:1 to 6285:1.
Planetary gearheads are particularly suitable for the transmission of high torque. Gearheads 22 mm in diameter and larger are equipped with ball bearings.



Sequence of steps in determining the components of a gearmotor combination:

1. Power requirement estimate

$$P_L = M \cdot \omega = M_L \cdot \frac{\pi}{30} \cdot n_L$$

2. Gearhead selection

Choose the one gearhead from the gearhead pages that just meets the application's torque requirements.

3. Determination of the theoretical reduction ratio:

$$i_{\text{theor}} = \frac{n_{\text{Gear perm}}}{n_L}$$

4. Determination of motor speed based on the nearest available gearhead reduction ratio.

$$n_{\text{Mot}} = i \cdot n_L$$

5. Gearhead efficiency

$$\eta_{\text{Gear}} = (\eta_{\text{Stage}})^s = \eta_1 \cdot \eta_2 \cdot \eta_3 \cdot \dots$$

6. Calculation of necessary motor torque

$$M_{\text{Mot}} = \frac{M_L [\text{Nm}] \cdot 1000}{i \cdot \eta_{\text{Gear}}} = [\text{mNm}]$$

7. Choice of motor type

The gearhead data pages indicate the motor types compatible with the chosen gearhead. Choose a type that meets power requirements and other prerequisites such as dimensions, tacho option, etc.

8. Motor choice, a cursory assessment

$$n_{\text{Mot}} < n_{\text{limit}}$$

$$M_{\text{Mot}} < M_{\text{permiss}}$$

9. Choice of winding number

a) Determination of the motor's no load speed

$$n_i = n_{\text{Mot}} + \left(\frac{\Delta n}{\Delta M} \cdot M_{\text{Mot}} \right)$$

b) Calculation of theoretical speed constant

$$k_{ni} = \frac{n_i}{U}$$

c) Effective speed constant

The speed constant of the motor selected under item 7 can be obtained from line 15 of the motor data sheet.

d) Choice of winding

$$k_{ni} \approx k_{n\text{Mot}}$$

10. Verification of calculated values

Compare gearmotor speed of the unit assembled using the components selected with the initial requirements. The procedure may need to be repeated with other motor and gearhead data, if necessary.

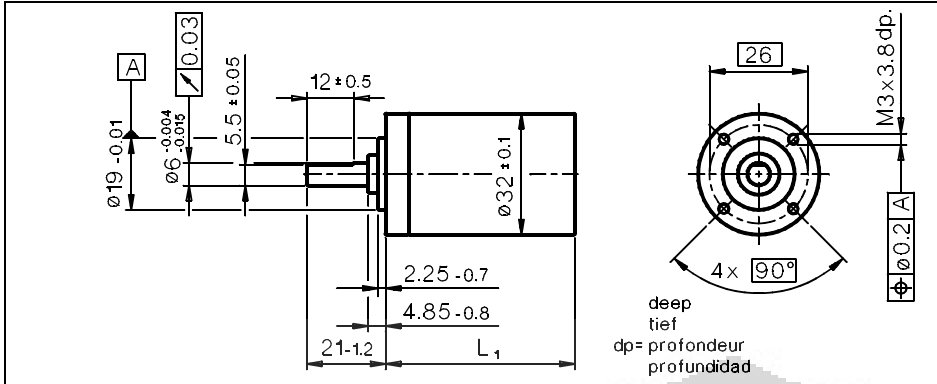
Definition of abbreviations:

i	= Actual gearhead reduction
i_{theor}	= theoretical gearhead reduction
I_{permiss}	= Max. permissible continuous current (Line 10) [mA]
k_M	= Torque constant (Line 14) [mNm/A]
k_{ni}	= Theoretical speed constant [rpm/V]
$k_{n\text{Mot}}$	= Actual motor speed constant (Line 15) [rpm/V]
M_L	= Output torque [Nm]
M_{Mot}	= Motor torque [mNm]
M_{permiss}	= Max. permissible continuous torque (Line 11) [mNm]
	= $k_M \cdot I_{\text{permiss}}/1000$
$n_{\text{Gear perm}}$	= Recommended input speed [rpm]
n_i	= Theoretical no load speed [rpm]
n_L	= Output speed under load [rpm]
n_{limit}	= Max. permissible speed [rpm] (Line 9)
n_{Mot}	= Motor speed [rpm]
P_L	= Output power [W]
s	= Number of gearhead stages
U	= Available voltage [V]
U_N	= Nominal voltage (Line 2) [V]
$\frac{\Delta n}{\Delta M}$	= Speed/torque gradient (Line 5) [rpm/mNm]
η_{gear}	= Gearhead efficiency
η_{stage}	= Efficiency per gearhead stage
Note:	Line references above pertain to motor data pages.

Planetary Gearhead
 Ø32 mm, 0.75-4.5 Nm

CERAMIC VERSION
 on request. New performance levels!

METAL VERSION
 Plastic version see page 125 - 128



Technical Data

Planetary gearhead	straight teeth
Bearing at output	ball bearing
Output shaft	steel not hardened
Radial play, 5 mm from flange	0.28 mm
Axial play	max. 0.47 mm
Max. perm. radial load, 12 mm from flange	140 N
Max. perm. axial load	120 N
Max. permissible force for press fits	120 N
Average backlash no load per stage	< 1.3°
Mass inertia	1.87-1.687 gcm ²
Sense of direction	=
Recommended input speed	< 5000 rpm
Recommended temperature range	-15/+80°C

Order Number	Reduction	Reduction absolute	Order Number	Reduction	Reduction absolute	Order Number	Reduction	Reduction absolute	No. of stages	max. Torque		Weight [g]	L1 [mm]	L2 [mm]	L3 [mm]	L4 [mm]	L5 [mm]	L6 [mm]	L7 [mm]	L8 [mm]	
										Continuous [Nm]	Intermittent [Nm]										
114435	3.7 : 1	3 ⁵ / ₇	110364	4.8 : 1	4 ⁴ / ₅	114436	5.8 : 1	5 ³ / ₄	1	0.75	1.13	80	118	26.50	83.50	100.40	105.30	103.25	90.90	109.00	111.85
110365	14 : 1	13 ³⁹ / ₁₄₉	110366	18 : 1	17 ²⁹ / ₁₃₅	114437	21 : 1	21 ⁵ / ₁₄	2	2.25	3.38	75	162	36.40	93.40	110.30	115.20	113.15	100.80	118.90	121.75
110367	23 : 1	23 ¹ / ₂₅	110368	28 : 1	27 ³ / ₅	110369	33 : 1	33 ¹ / ₁₆	2	2.25	3.38	75	162	36.40	93.40	110.30	115.20	113.15	100.80	118.90	121.75
114438	51 : 1	51 ⁸³ / ₂₄₃	110370	66 : 1	66 ⁸⁴ / ₂₄₅	114439	79 : 1	79 ¹⁶ / ₁₄₉	3	4.50	6.75	70	194	43.10	100.10	117.00	121.90	119.85	107.50	125.60	128.45
110371	86 : 1	85 ¹⁰¹ / ₁₇₅	114440	103 : 1	102 ¹⁹ / ₃₅	110372	111 : 1	110 ⁷⁴ / ₁₂₅	3	4.50	6.75	70	194	43.10	100.10	117.00	121.90	119.85	107.50	125.60	128.45
114441	123 : 1	122 ⁴⁵ / ₅₆	114442	132 : 1	132 ¹² / ₂₅	110373	159 : 1	158 ⁷ / ₁₀	3	4.50	6.75	70	194	43.10	100.10	117.00	121.90	119.85	107.50	125.60	128.45
110374	190 : 1	190 ⁷ / ₆₄							3	4.50	6.75	70	194	43.10	100.10	117.00	121.90	119.85	107.50	125.60	128.45
110375	246 : 1	245 ¹⁶⁴⁹ / ₁₇₁₅	114443	295 : 1	294 ²²⁰ / ₃₄₃	110376	318 : 1	317 ¹⁰⁵¹ / ₁₂₂₅	4	4.50	6.75	60	226	49.80	106.80	123.70	128.60	126.55	114.20	132.30	135.15
110377	411 : 1	410 ⁸⁷⁴ / ₈₇₅	114444	456 : 1	456 ²⁵ / ₁₉₆	114445	492 : 1	492 ¹² / ₁₇₅	4	4.50	6.75	60	226	49.80	106.80	123.70	128.60	126.55	114.20	132.30	135.15
110378	531 : 1	530 ⁵²⁶ / ₆₂₅	114446	589 : 1	589 ¹⁶ / ₃₅	114447	636 : 1	635 ¹¹³ / ₁₂₅	4	4.50	6.75	60	226	49.80	106.80	123.70	128.60	126.55	114.20	132.30	135.15
114448	706 : 1	706 ²⁷ / ₂₂₄	110379	762 : 1	761 ¹⁹ / ₂₅	114449	913 : 1	912 ²¹ / ₄₀	4	4.50	6.75	60	226	49.80	106.80	123.70	128.60	126.55	114.20	132.30	135.15
114450	1093 : 1	1093 ³³ / ₂₅₆							4	4.50	6.75	60	226	49.80	106.80	123.70	128.60	126.55	114.20	132.30	135.15
110380	1181 : 1	1180 ²⁷⁶ / ₈₅₇₅	114451	1414 : 1	1414 ⁴⁷⁸ / ₁₇₁₅	110381	1526 : 1	1525 ³⁹⁹ / ₆₁₂₅	5	4.50	6.75	50	258	56.50	113.50	130.40	135.30	133.25	120.90	139.00	141.85
114452	1694 : 1	1694 ¹²⁹ / ₆₈₆	114453	1828 : 1	1827 ⁸³⁷ / ₁₂₂₅	110382	1972 : 1	1971 ³⁰⁵¹ / ₄₃₇₅	5	4.50	6.75	50	258	56.50	113.50	130.40	135.30	133.25	120.90	139.00	141.85
114454	2189 : 1	2189 ¹⁰¹ / ₂₄₅	114455	2362 : 1	2361 ⁸¹³ / ₆₇₅	110383	2548 : 1	2548 ¹²⁴ / ₃₁₂₅	5	4.50	6.75	50	258	56.50	113.50	130.40	135.30	133.25	120.90	139.00	141.85
114456	2623 : 1	2622 ⁵⁷⁵ / ₇₈₄	114457	2829 : 1	2829 ⁶⁹ / ₁₇₅	114458	3052 : 1	3052 ²¹² / ₆₂₅	5	4.50	6.75	50	258	56.50	113.50	130.40	135.30	133.25	120.90	139.00	141.85
114459	3389 : 1	3389 ⁵⁹ / ₁₄₀	114460	3656 : 1	3656 ⁵⁶ / ₁₂₅	114461	4060 : 1	4060 ¹⁷³ / ₈₉₆	5	4.50	6.75	50	258	56.50	113.50	130.40	135.30	133.25	120.90	139.00	141.85
114462	4380 : 1	4380 ³ / ₂₅	114463	5247 : 1	5247 ⁹ / ₁₆₀	114464	6285 : 1	6285 ³⁰³ / ₁₀₂₄	5	4.50	6.75	50	258	56.50	113.50	130.40	135.30	133.25	120.90	139.00	141.85

Stock program
 Standard program
 Special program (on request!)

Continued from page 121

Continued on page 123

+ Motor Order Number

23 32 - 62 . 2 1 6 - 2 0 0

Basic motor 2332...-12.216-200
 Details see page 63

23 32 - 75 . 2 3 6 - 2 0 0

Basic motor 2332...-51.236-200
 Details see page 63

23 32 - 75 . 2 3 6 - 2 0 0

Basic motor 2332...-51.236-200
 Details see page 63

23 32 - 76 . 2 3 6 - 2 0 0

Basic motor 2332...-51.236-200
 Details see page 63

23 32 - 81 . 2 2 5 - 2 0 0

Basic motor 2332...-52.215-200
 Details see page 64

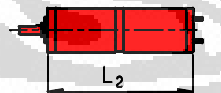
23 32 - 75 . 2 2 5 - 2 0 0

Basic motor 2332...-52.215-200
 Details see page 64

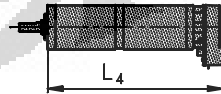
23 32 - 76 . 2 2 5 - 2 0 0

Basic motor 2332...-52.215-200
 Details see page 64

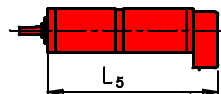
+ Tacho/Encoder



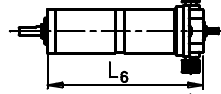
DC-Tacho
 Ø22 mm, 0.52 V
 Details see page 141



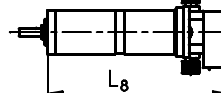
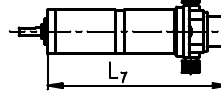
DC-Tacho/Encoder
 Details see page 153 **36 04**



Digital Encoder
 HED_ 5540, 500 CPT, 3 channels
 Details see page 146



DC-Tacho
 Ø22 mm, 0.52 V
 Details see page 141



Digital Encoder
 HED_ 5540, 500 CPT, 3 channels
 Details see page 147