

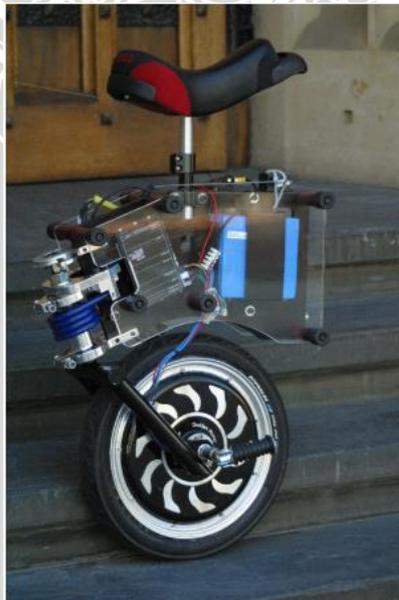
BAB II

TINJAUAN PUSTAKA

2.1 Micycle

Pada penelitian terdahulu *Son of edgar (state space control of electro drive gravity aware ride, 2006)* menjelaskan bahwa segway memiliki komponen listrik yang kompleks, untuk itu masing masing komponen dibuat agar mendapat pasokan daya sesuai kebutuhannya. Papan kelistrikan dipasang oleh baterai 72 volt dan diberikan sekring agar menjamin keamanan pada arus tegangan tinggi, selanjutnya motor dc dirangkai secara paralel (teknik pomits, 2012).

Pada bulan maret 2010 siswa teknik *university of Adelaide* menciptakan alat *balancing unicycle* atau bisa disebut *micycle*. Kendaraan ini menggunakan tunggal roda dengan pendulum terbalik, seperti robot roda dua *self balancing* (yakamoto dan chikamasa, 2009). Dengan cara manusia mengatur pedal bertenaga *unicycle* yang telah di identifikasikan secara eksperimental (sheng dan yamafuji, 1997) ditunjukkan pada gambar 2.1.



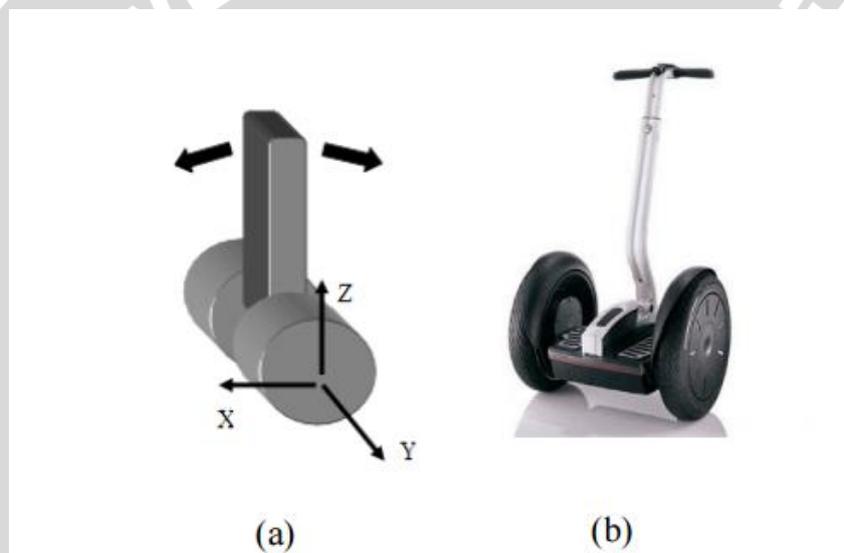
Gambar 2.1 Micycle

Sumber : Jurnal modelling, simulation and control of an electric unicycle

Pada system *micycle* membahas tentang mekanik dan komponen listrik yang digunakan dalam system tersebut. Mekanik *steering mechanism* digunakan untuk

mengarahkan dan membantu pengendara dalam menyeimbangkan arah pada jalannya micycle.

Segway pada dasarnya adalah sebuah robot yang dikenal dengan sebutan “balancing robot”. Balancing robot bekerja dengan cara memutar kedua rodanya agar tubuhnya tetap tegak. Apabila tubuh robot semakin miring ke depan, maka rodanya juga akan semakin cepat berputar ke depan, sehingga posisi tubuh robot menjadi tegak kembali, begitu juga sebaliknya. Namun ada perbedaan antara balancing robot dan segway, yaitu segway dapat berbelok ke kiri dan ke kanan. Agar dapat berbelok ke kiri atau ke kanan, maka kecepatan rodanya harus dibuat berbeda. Perbedaan kecepatan tersebut ditentukan berdasarkan kemiringan stang Kataren *et al.*, (2015). Gambar balancing robot dan segway ditunjukkan pada gambar 2.2.



Gambar 2.2 (a)balancing robot(b)segway

Sumber : Sistem Kendali Gerak Segway Berbasis Mikrokontroler vol 2

Berdasarkan cara kerja gerak segway, maka komponen pembentuk sistem ini dapat dibagi menjadi dua bagian, yaitu bagian elektrik dan perangkat lunak. Sedangkan untuk pengaplikasiannya, diperlukan suatu bagian pendukung yaitu bagian mekanik. Bagian mekanik ini sendiri terdiri dari rangka, pijakan, stang, roda dan gear. Kemudian bagian elektrik terdiri dari kontroler yang menggunakan mikrokontroler ATmega 8535, sensor sudut menggunakan accelerometer, sensor kecepatan sudut menggunakan gyroscope, sensor simpangan stang menggunakan potensiometer, driver motor menggunakan driver motor DC jembatan H, motor menggunakan starter motor, serta catu daya menggunakan aki kering. Sedangkan bagian perangkat lunak terdiri dari pengakses sensor sudut,

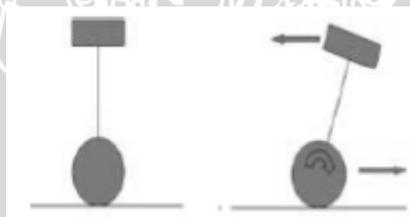
pengakses sensor kecepatan sudut, pengakses sensor simpangan stang, metode kendali PID, penghasil PWM serta low pass filter.

Berdasarkan penelitian son of edgar catu daya yang dipakai pada Segway tersebut menggunakan 72 volt, daya tersebut terlalu besar untuk alat transportasi jarak pendek. Sedangkan pada penelitian alat transportasi micycle, bentuk desain alat tersebut kurang pada interaksi manusia dan komputer. Alat yang akan saya buat menggunakan power daya 24 volt dan desain pada skateboard beroda satu.

Bentuk penyederhanaan Segway yang ditawarkan skateboard roda satu ini adalah hilangnya setang pada bagian depan. Skateboard roda satu ini tidak akan bergerak sebelum ada sepasang kaki yang berdiri di atasnya. Lebih pintar lagi, skateboard roda satu ini akan langsung mengerem saat pengguna menurunkan kakinya.

2.2 Robot Segway

Dasar untuk membuat robot segway beroda dua dapat setimbang adalah mudah dengan cara mengendalikan roda searah dengan jatuhnya bagian atas sebuah robot. Apabila proses tersebut dapat terlaksana maka robot tersebut dapat setimbang Kataren *et al.*, (2015) ditunjukkan pada gambar 2.3.



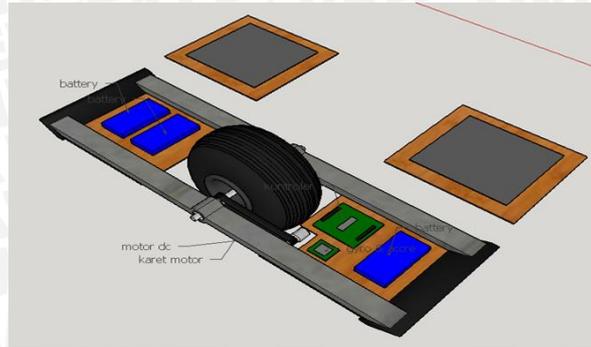
Gambar 2.3 Balancing Segway Robot Beroda Dua Menyeimbangkan Diri

Sumber : Jurnal Elementer Vol.1

Saat balancing robot beroda dua condong kedepan atau miring ke kanan seperti Gambar 2.3, maka tindakan yang perlu dilaksanakan adalah motor bergerak searah dengan arah kemiringan yang terjadi, sehingga robot akan kembali tegak lurus dengan permukaan bidang datar, gaya yang digunakan untuk menyeimbangkan robot didapat dari putaran roda yang dihasilkan dari motor DC.

2.3 Plant

Plant yang digunakan adalah berupa skateboard onewheel yang dapat menyeimbangkan diri. Plant dapat dilihat dalam Gambar 2.4.



Gambar 2.4 Plant Skateboard Onewheel

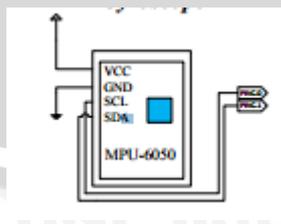
2.4 Sensor Gyroscope dan Accelerometer

Gyroscope merupakan suatu alat elektronik yang berfungsi untuk mengukur kecepatan sudut dengan satuan ($^{\circ}/s$) yang dialami oleh suatu benda *pitch*, *roll* dan *yaw*. Sedangkan sensor *accelerometer* sendiri merupakan suatu piranti elektronik yang digunakan untuk mengukur percepatan yang terjadi pada keadaan tertentu. Sensor *accelerometer* dapat digunakan untuk mendapatkan posisi dari suatu benda dengan melakukan percepatan itu sendiri sebanyak dua kali terhadap waktu (Seifert, dkk, 2007).

Sensor MPU-6050 membutuhkan tegangan kerja 3,3V. Tetapi modul ini telah dilengkapi dengan regulator tegangan 3,3V sehingga bisa langsung dihubungkan ke tegangan 5V. Sensor ini mempunyai dua buah keluaran yaitu SCL dan SDA masing-masing dihubungkan ke PC.0 dan PC.1 ditunjukkan pada gambar 2.4 dan gambar 2.5.



Gambar 2.4. Sensor MPU-6050 3-Axis Accelerometer+ 3-Axis Gyroscope



Gambar 2.5. Konfigurasi Sensor MPU-6050

Sumber : Tugas akhir Self-Balancing Scooter Menggunakan Metode Kendali Proporsional Integral Derivatif

2.5 Arduino Uno

Arduino UNO adalah sebuah board mikrokontroler yang didasarkan pada ATmega328. Arduino UNO mempunyai 14 pin *digital input/output* (6 di antaranya dapat sebagai output PWM), 6 input analog, sebuah osilator Kristal 16 MHz, sebuah koneksi USB, sebuah *power jack*, sebuah ICSP header, dan sebuah tombol reset. Arduino UNO memuat semua yang dibutuhkan untuk menunjang mikrokontroler, mudah menghubungkannya ke sebuah komputer dengan sebuah kabel USB atau mensuplainya dengan sebuah *adaptor* AC ke DC atau menggunakan baterai untuk memulainya. Arduino Uno ditunjukkan dalam Gambar 2.6.

Uno berbeda dengan semua *board* sebelumnya dimana tidak menggunakan *chip driver* FTDI USB-to-serial. Sebagai gantinya, mengutamakan Atmega16U2 (Atmega8U2 untuk versi R2) diprogram sebagai konverter USB-to-serial.



Gambar 2.6 Arduino Uno R3

(Sumber: <http://arduino.cc/en/Main/arduinoBoardUno>)

Nama “Uno” berarti *satu* dalam bahasa Italia, untuk menandai peluncuran Arduino 1.0. Uno dan versi 1.0 akan menjadi versi referensi dari Arduino. Uno adalah yang terbaru dalam serangkaian *board* USB Arduino, dan sebagai model referensi untuk *platform* Arduino, untuk perbandingan dengan versi sebelumnya.

2.5.1 Daya

Arduino Uno dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal (otomatis). Daya eksternal (non-USB) dapat berasal baik dari AC-ke-DC *adaptor* atau baterai. *Adaptor* ini dapat dihubungkan dengan menancapkan *plug jack*

pusat-positif ukuran 2,1 mm ke *jack power* pada *board*. Ujung kepala dari baterai dapat dimasukkan kedalam Gnd dan Vin pin *header* dari konektor *power*. Kisaran kebutuhan daya yang disarankan untuk *board* Uno adalah 7 sampai dengan 12 volt, jika diberi daya kurang dari 7 volt kemungkinan pin 5V Uno dapat beroperasi tetapi tidak stabil kemudian jika diberi daya lebih dari 12V, *regulator* tegangan bisa panas dan dapat merusak *board* Uno.

Pin daya adalah sebagai berikut:

- Vin : Tegangan masukan kepada *board* Arduino ketika itu menggunakan sumber daya eksternal (sebagai pengganti dari 5 volt koneksi USB atau sumber daya lainnya).
- 5V : Catu daya digunakan untuk daya mikrokontroler dan komponen lainnya. Sebuah pasokan 3,3 volt dihasilkan oleh *regulator on-board*.
- 3,3V : Suplai tegangan 3,3 volt dihasilkan oleh *regulator* pada *board*. Arus maksimum yang dapat digunakan adalah 50 mA.
- GND : Pin *ground*.
- IOREF : Pin ini pada *board* Arduino menyediakan tegangan referensi dimana mikrokontroler bekerja.

2.5.2 Memori

ATmega328 memiliki 32 KB (dengan 0,5 KB digunakan untuk *bootloader*), 2 KB dari SRAM dan 1 KB EEPROM.

2.5.3 Input dan Output

Masing-masing dari 14 pin *digital* di Uno dapat digunakan sebagai *input* atau *output*, dengan menggunakan fungsi *pinMode()*, *digitalWrite()* dan *digitalRead()*, beroperasi dengan daya 5 volt. Setiap pin dapat memberikan atau menerima maksimum 40 mA dan memiliki internal *pull-up resistor* (secara default terputus) dari 20-50 kOhm.

Selain itu, beberapa pin memiliki fungsi khusus:

- *Serial*: 0 (RX) dan 1 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) TTL data serial. Pin ini dihubungkan ke pin yang berkaitan dengan *chip* Serial ATmega8U2 USB-to-TTL.
- *Interrupts Eksternal*: 2 dan 3. Pin ini dapat dikonfigurasi untuk memicu *interrupt* pada nilai yang rendah, dengan batasan tepi naik atau turun, atau perubahan nilai.
- PWM: 3, 5, 6, 9, 10, dan 11. Menyediakan *output* PWM 8-bit dengan fungsi *analogWrite()*.

- *SPI*: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Pin ini mendukung komunikasi SPI menggunakan *SPI library*.
- *LED*: 13. Ada *built-in* LED terhubung ke pin digital 13. Ketika pin bernilai nilai HIGH, LED *on*, ketika pin bernilai LOW, LED *off*.

Uno memiliki 6 masukan analog, berlabel A0 sampai dengan A5, yang masing-masing menyediakan 10 bit dengan resolusi (yaitu 1024 nilai yang berbeda).

Selain itu, beberapa pin memiliki fungsi khusus:

- *TWI*: A4 (pin SDA) dan A5 (pin SCL). Dukungan komunikasi TWI menggunakan *Wire library*.
- *AREF*: Tegangan referensi (0 sampai 5V) untuk *input* analog. Digunakan dengan fungsi *analogReference()*.
- *Reset*: Bawa baris ini LOW untuk me-*reset* mikrokontroler.

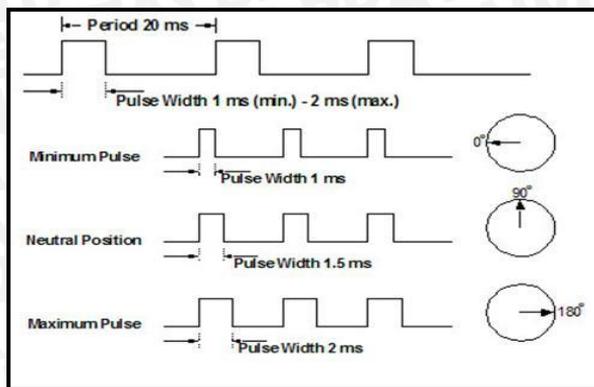
2.5.4 Komunikasi

Arduino Uno memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lainnya. ATmega328 menyediakan UART TTL (5V) untuk komunikasi serial, yang tersedia di pin *digital* 0 (RX) dan 1 (TX). Sebuah Atmega16U2 sebagai saluran komunikasi serial melalui USB dan sebagai *virtual com port* untuk perangkat lunak pada komputer. *Firmware* '16U2 menggunakan *driver* USB standar COM, dan tidak ada *driver* eksternal yang diperlukan. Namun, pada *Windows* diperlukan sebuah *file .inf*. Perangkat lunak Arduino terdapat *monitor* serial yang memungkinkan digunakan memonitor data tekstual sederhana yang akan dikirim ke atau dari board Arduino. LED RX dan TX pada *board* akan berkedip ketika data sedang dikirim melalui *chip* USB-to-serial dengan koneksi USB ke komputer (tetapi tidak untuk komunikasi serial pada pin 0 dan 1).

Sebuah *SoftwareSerial library* memungkinkan untuk berkomunikasi secara serial pada salah satu pin digital pada *board* Uno. ATmega328 juga mendukung I2C (TWI) dan komunikasi SPI. Perangkat lunak Arduino termasuk *Wire library* untuk menyederhanakan penggunaan *bus* I2C. Untuk komunikasi I2C, gunakan *SPI library*.

2.6 PWM (Pulse With Modulation)

PWM (*Pulse Width Modulation*) memiliki fungsi untuk mengatur sudut putaran motor servo. Teknik PWM (*pulse width modulation*) untuk mengatur sudut motor servo dapat dilihat dalam Gambar 2.7.



Gambar 2.7 Pengaturan Sudut Motor Servo

Sumber : www.electrooons.com

Berdasarkan Gambar 2.7 dapat diasumsikan bahwa saat diberikan sinyal periodik dengan lebar sebesar 1 ms maka motor servo akan bergerak dengan sudut 0° . Apabila diberikan sinyal sebesar 1.5 ms maka motor servo akan bergerak dengan sudut 90° . Begitu pula jika diberi sinyal 2 ms, maka motor servo akan bergerak dengan sudut 180° . Perhitungan rumus motor servo akan ditunjukkan Persamaan 2-1 sebagai berikut:

$$S = D \left(\frac{1000}{180} \right) + 1000 \mu s \dots \dots \dots (2-1)$$

1)

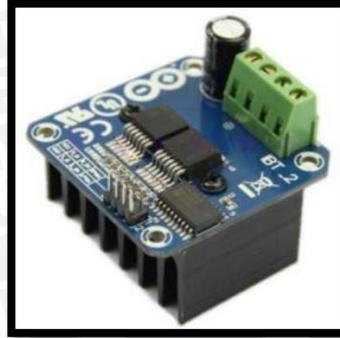
Dimana:

S = Lebar Pulsa dalam μs

D = Sudut putar servo dalam derajat

2.7 Driver BTS 7960

Pada plant jaket penghangat suhu tubu terdapat kawat nikelin sebagai pemanas dalam jaket. Namun karena catu Kawat Nikelin mempunyai range tegangan 0-24 V sehingga dibutuhkan driver untuk memaksimalkan kinerja kawat nikelin saat menghasilkan panas yang diinginkan. Untuk driver plant sendiri menggunakan *high current 30A dc driver (two high current half bridge infineon BTS 7960 chip)*. Driver BTS7960 dapat dilihat pada Gambar 2.8.



Gambar 2.8 Driver BTS7960

(Sumber:datasheet Future Electronics)

2.8 Motor BLDC

Salah satu jenis motor listrik yang paling banyak digunakan akhir-akhir ini adalah Brushless DC (BLDC) Motor dimana motor DC ini tidak menggunakan Brush (sikat) untuk proses komutasi. BLDC motor banyak digunakan pada teknologi otomasi mutakhir, seperti: otomasi industri manufaktur maupun otomasi non-manufaktur (Robot, UAV, ROV, aeromodeling, hingga RC cars). Konstruksi yang sangat simpel menjadi pertimbangan pemakaian BLDC motor pada pengembangan bidang otomasi tersebut. Selain itu, mobil listrik maupun mobil hybrid yang menjadi trend akhir-akhir ini, juga menggunakan BLDC motor sebagai tenaga penggerak karena mempunyai efisiensi yang sangat besar hingga mencapai 95%. BLDC motor juga digunakan pada motor penggerak hardisk yang membutuhkan kecepatan serta ketahanan yang tinggi.



Gambar 2.9 Salah satu contoh BLDC motor

(sumber : www.nwuav.com)

Ø Speed:

Jumlah pole pair = semakin sedikit jumlah pole pair maka semakin cepat putaran rpm motor BLDC.

Dalam jumlah pole pair yang sama maka BLDC yang memiliki resistansi kumparan lebih rendah akan memiliki top speed rpm yang lebih tinggi.

Dalam resistansi kumparan dan pole pair yang sama BLDC yang diberikan Voltase yang lebih tinggi akan memiliki top speed yang lebih tinggi. Hal ini juga sesuai hukum ohm $I=V/R$, apabila R tetap namun V semakin besar, maka I akan semakin besar pula, sehingga bisa dikatakan daya motor BLDC juga akan meningkat.

Ohm's Law

$$I = \frac{V}{R}$$

Electric current = Voltage / Resistance

Ø Torsi

Struktur kumparan sangat berpengaruh terhadap torsi, tipe kumparan segitiga delta dan kumparan tipe star tentunya memiliki torsi yang berbeda.

Jumlah pole pair = semakin banyak jumlah pole pair akan memiliki torsi yang semakin besar.

Lebar magnet = semakin lebar dan semakin kuat daya induksi magnet maka semakin besar pula torsi motor BLDC

Yang terakhir adalah kemampuan kontroller, untuk memenuhi kebutuhan torsi yang besar diperlukan Amper yang besar pula, maka kontroller harus support dan mampu mengalirkan arus yang besar.

2.9 Kontroler

Kontroler merupakan salah satu komponen sistem pengaturan yang berfungsi mengolah sinyal umpan balik dan sinyal masukan acuan (*set point*) atau sinyal *error* menjadi sinyal kontrol. Sinyal error disini adalah selisih antara sinyal umpan balik yang dapat berupa sinyal keluaran plant sebenarnya atau sinyal keluaran terukur dengan sinyal masukan acuan (*set point*).

Keberadaan kontroler dalam sebuah sistem kontrol mempunyai kontribusi yang besar terhadap perilaku sistem. Pada prinsipnya hal itu disebabkan oleh tidak dapat diubahnya komponen penyusun sistem tersebut. Artinya, karakteristik plant harus terima sebagaimana adanya, sehingga perubahan perilaku sistem hanya dapat dilakukan melalui penambahan suatu sub sistem, yaitu kontroler.

Salah satu fungsi komponen kontroler adalah mengurangi sinyal kesalahan, yaitu perbedaan antara nilai referensi/ nilai yang diinginkan dan nilai aktual. Hal ini sesuai dengan tujuan sistem kontrol dimana mendapat nilai sinyal keluaran sama dengan nilai yang diinginkan referensi. Semakin kecil kesalahan yang terjadi, semakin baik kinerja sistem kontrol yang diterapkan.

Apabila perbedaan antara nilai referensi dengan nilai keluaran relatif besar, maka kontroler yang baik seharusnya mampu mengamati perbedaan ini untuk segera menghasilkan sinyal keluaran untuk mempengaruhi plant. Dengan demikian sistem secara cepat mengubah keluaran plant sampai diperoleh selisih dengan nilai referensi sekecil mungkin.

Prinsip kerja kontroler adalah membandingkan nilai aktual keluaran plant dengan nilai referensi, kemudian menentukan nilai kesalahan dan akhirnya menghasilkan sinyal kontrol untuk meminimalkan kesalahan (Ogata, 1996: 197-204).

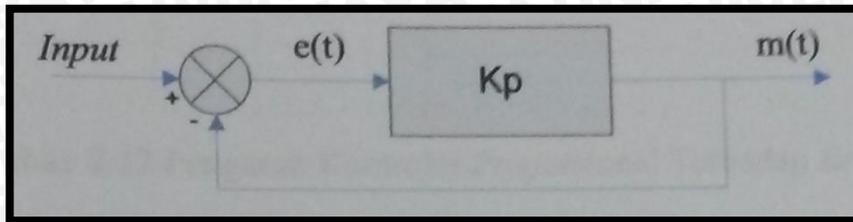
2.10 Kontroler PID (Proportional Integral Derivative)

PID 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 *Proporsional*, *Integral*, *Derivative*. Ketiganya dapat dipakai bersamaan maupun sendiri-sendiri tergantung dari respon yang kita inginkan terhadap suatu plant.

2.10.1 Kontroler Proporsional

Kontroler proporsional memiliki keluaran yang sebanding/ proporsional 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 keluaran sebesar konstanta pengalinya.

Pada Gambar 2.10 menunjukkan blok diagram yang menggambarkan hubungan antara input (besaran referensi yang diinginkan), besaran aktual dengan besaran keluaran kontroler proporsional, dan besaran kesalahan (*error*). Sinyal kesalahan (*error*) merupakan selisih antara besaran setting dengan besaran aktualnya.



Gambar 2.10 Diagram Blok Kontroler Proporsional
(Sumber: Ogata K.,1997)

Pada pengendali proporsional hubungan antara keluaran controller $m(t)$ dan sinyal kesalahan $e(t)$ adalah

$$m(t) = K_p \cdot e(t) \dots \dots \dots (2-1)$$

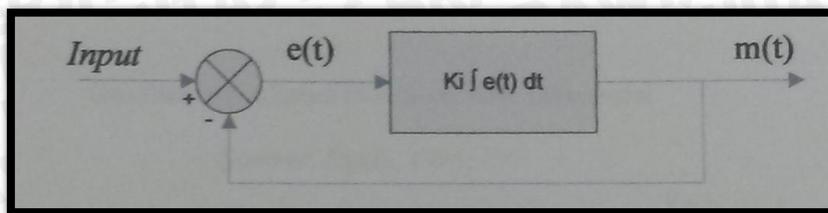
(Sumber : Ogata, 1995; 157)

Dengan K_p adalah penguatan proporsional. Keluaran $m(t)$ hanya tergantung pada K_p dan error, semakin besar error maka semakin besar koreksi yang dilakukan. Penambahan K_p akan menaikkan penguatan sistem sehingga dapat digunakan untuk memperbesar kecepatan respon dan mengurangi kesalahan keadaan mantap.

2.10.2 Kontroler Integral

Kontroler integral berfungsi mengurangi kesalahan keadaan mantap yang dihasilkan pada kontroler proporsional sebelumnya. Kalau sebuah plant tidak memiliki unsur untegrator ($1/s$), kontroler proporsional tidak akan mampu menjamin keluaran sistem dengan kesalahan keadaan mantapnya nol.

Kontroler integral memiliki karakteristik seperti halnya sebuah integral. Keluaran kontroler sangat dipengaruhi oleh perubahan yang sebanding dengan nilai sinyal kasalahan. Keluaran kontroler ini merupakan jumlahan yang terus menerus dari perubahan masukannya. Kalau sinyal kesalahan tidak mengaami perubahan, keluaran akan menjaga keadaan seperti sebelum terjadinya perubahan masukan. Gambar 2.11 menunjukkan blok diagram kontroler integral.



Gambar 2.11 Diagram Blok Kontroler Integral

(Sumber: Ogata K.,1997)

Nilai keluaran kontroler $m(t)$ sebanding dengan integral sinyal kesalahan $e(t)$, sehingga

$$\frac{dm(t)}{dt} = Ki \cdot e(t) \dots \dots \dots (2-2)$$

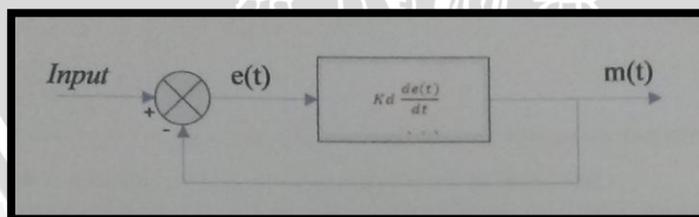
$$m(t) = Ki \int e(t)dt \dots \dots \dots (2-3)$$

(Sumber : Ogata, 1995:157)

Dengan K_i adalah konstanta integral. Jika sinyal kesalahan $e(t)=0$, maka laju perubahan sinyal kendali integral $\frac{dm(t)}{dt} = 0$ atau sinyal keluaran kendali akan tetap berada pada nilai yang dicapai sebelumnya. Aksi kontrol integral digunakan untuk menghilangkan kesalahan posisi dalam keadaan mantap (*error steady state*) tanpa memperhitungkan kecepatan respon.

2.10.3 Kontroler Differensial

Kontroler diferensial memiliki sifat seperti halnya suatu operasi derivatif. Perubahan yang mendadak pada masukan kontroler, akan mengakibatkan perubahan yang sangat besar dan cepat. Gambar 2.12 berikut menunjukkan blok diagram pada kontroler diferensial.



Gambar 2.12 Diagram Blok Kontroler Diferensial

(Sumber: Ogata K.,1997)

Nilai keluaran kontroler $m(t)$ sebanding laju sinyal kesalahan. Hubungan ini dapat ditulis sebagai :

$$m(t) = Kd \frac{de(t)}{dt} \dots\dots\dots(2-4)$$

Sumber: Ogata, 1995: 179

Kontroler diferensial akan memberikan sinyal kendali keluaran $m(t)=0$, untuk sinyal kesalahan $e(t)$ yang konstan sehingga kontroler diferensial tidak mempengaruhi keadaan mantap. Kontroler diferensial digunakan untuk memperbaiki atau mempercepat pespons transien sebuah sistem serta dapat meredam osilasi.

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

Dari ketiga aksi kontrol dasar diatas dapat dibuat kombinasi dari ketiganya, yaitu:

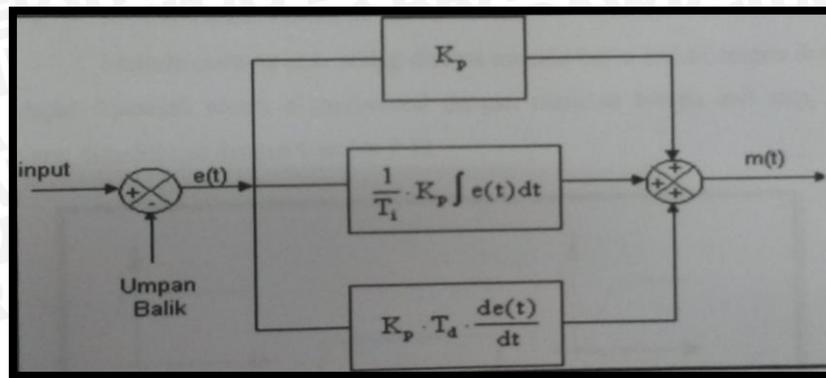
2.10.4 Kontroler Proporsional Integral Diferensial (PID)

Setiap kekurangan dan kelebihan dari masing –masing kontroler P,I dan D dapat saling menutupi dengan menggabungkan ketiga secara paralel menjadi kontroler *proporsional integral diferensial* (PID). Elemen- elemen kontroler P,I dan D masing-masing secara keseluruhan bertujuan untuk mempercepat reaksi sebuah sistem, menghilangkan offset dan menghasilkan perubahan awal yang besar (Ogata K.,1997). Kontroler PID memiliki diagram kendali seperti yang ditunjukkan pada Gambar 2.13.

Aksi kontrolnya dinyatakan sebagai:

$$m(t) = Kp e(t) + \frac{Kp}{Ti} \int_0^t e(t)dt + Kp Td \frac{de(t)}{dt} \dots\dots\dots(2-5)$$

Jenis kontroler ini digunakan untuk memperbaiki kecepatan respon, mencegah terjadinya kesalaha keadaan mantap serta memprtahankan kesetabilan.



Gambar 2.13 Diagram Blok Kontroler PID

(Sumber: Ogata K.,1997)

Keluaran kontroler PID merupakan penjumlahan dari keluaran kontroler proporsional, integral dan diferensial. Gambar 2.13 diatas menunjukkan hubungan tersebut. Karakteristik kontroler PID sangat dipengaruhi oleh kontribusi besar dari ketiga parameter P,I dan D. Penyetelan konstanta K_p , T_i dan T_d akan mengakibatkan penonjolan sifat dari masing –masing elemen. Satu atau dua dari ketiga konstanta tersebut dapat disetel lebih menonjol dibanding yang lain. Konstanta yang menonjol itulah yang akan memberikan kontribusi pengaruh pada respon sistem secara keseluruhan (Ogata K.,1997).

2.11 Metode Hand Tuning

Kontroler PID dapat di tuning dalam beberapa cara, antara lain *ziegler-nichols tuning*, *loop shaping*, metode analitis, optimasi, *pole placement*, *autotuning* dan *hand tuning* (Smith,1997; Astrom & Hagglund,2004). Dengan menggunakan metode *hand tuning*, nilai parameter P perlu diubah-ubah secara *trial* agar respon yang diperoleh sesuai dengan yang diinginkan.

2.12 Kalman Filter

R.E Kalman mempublikasikan tulisannya pada tahun 1960 berjudul "A New Approach to Linear Filtering and Prediction Problems." R.E Kalman menjelaskan solusinya atas masalah filter linear pada data diskrit. Sekarang Kalman filter itu sendiri sudah sering digunakan di kontrol sistem embedded dan sistem navigasi. Penulis ingin menunjukkan kelebihan Kalman filter untuk memberi prediksi yang baik untuk sudut robot terhadap permukaan bumi dan menjaga robot agar tetap seimbang. Karena Kalman Filter digunakan untuk menghilangkan segala noise pengukuran pada accelerometer dan gyroscope. Kelemahan dari Filter ini sendiri adalah tidak adanya standar dalam metode

mendapatkan persamaan matematis yang akan digunakan di Kalman Filter. Pada paper An Introduction to the Kalman Filter. Dijelaskan bahwa kalman filter dapat digunakan untuk mengilangkan noise dari sensor. Proses atau sistem harus di modelkan dalam sistem linear. Sistem linear dapat direpresentasikan menjadi dua persamaan.

Bentuk umum sistem dinamik stokastik linier diskrit adalah:

$$x_{k+1} = \mathbf{A}_k x_k + \mathbf{B}_k u_k + w_k$$

dengan pengukuran $z_k \in \mathcal{R}$ yang memenuhi

$$z_k = \mathbf{H}_k x_k + v_k$$

$$x_0 \sim N(\bar{x}_0, P_{x_0}); w_k \sim N(0, Q_k); v_k \sim N(0, R_k)$$

Dengan:

x_0 = inisial dari sistem,

x_{k+1} = variabel keadaan pada waktu $k + 1$ dan berdimensi,

x_k = variabel keadaan pada waktu k yang nilai estimasi awalnya x_0

dan kovariansi awal $P_{x_0}, x_k \in \mathcal{R}^n$,

u_k = vektor masukan deterministik, pada waktu k , $u_k \in \mathcal{R}^m$,

w_k = noise pada sistem dengan $\bar{w}_k = 0$ dengan konvariansi Q_k ,

z_k = variabel pengukuran $z_k \in \mathcal{R}^p$,

v_k = noise pada pengukuran dengan mean $\bar{v}_k = 0$ dengan kovariansi R_k ,

