

## BAB II

### TINJAUAN PUSTAKA

Bab ini menguraikan teori-teori pendukung skripsi, yang terdiri atas :

#### 2.1 *Unmanned Aerial Vehicle*

Pada dunia aeromodelling, pesawat tanpa awak *Unmanned Aerial Vehicle* (UAV) merupakan miniatur pesawat udara yang bisa diterbangkan baik secara manual maupun secara otomatis yang dapat dikendalikan menggunakan *Remote Control*. Miniatur pesawat udara ini terdiri atas 2 kategori yaitu pesawat udara aerodinamis dan pesawat udara aerostatis. Pesawat udara aerodinamis merupakan pesawat udara yang lebih berat dari udara seperti helikopter. Sedangkan pesawat udara aerostatis merupakan pesawat udara yang lebih ringan dari pada udara seperti balon udara (Espinoza, 2013).

#### 2.2 Quadcopter

Quadcopter adalah jenis pesawat udara aerodinamis bermotor dan bersayap putar (*rotary wing*). Sebagaimana dengan namanya *Quad* yang berarti empat, sehingga pada quadcopter memiliki 4 buah motor *brushless* dan 4 buah baling-baling (*propeller*). Berdasarkan bentuk rangka (*frame*), quadcopter terdiri atas 2 model yaitu model dengan frame berbentuk “plus” (+), dan model dengan frame berbentuk “silang” (x). Pada skripsi ini menggunakan rangka quadcopter yang berbentuk silang, bentuk fisik rangka quadcopter ditunjukkan dalam Gambar 2.1.



Gambar 2.1 Rangka Quadcopter

Sumber: [www.dji.com](http://www.dji.com)

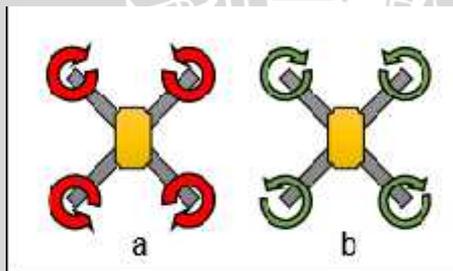
Rangka quadcopter yang digunakan dibuat oleh DJI dengan seri Flame Wheel 450 (F450) yang memiliki spesifikasi:

1. *Frame Weight* : 282g
2. *Diagonal Wheelbase* : 450mm
3. *Takeoff Weight* : 800g ~ 1600g
4. *Recommended Propeller* : 10 x 3,8in ; 8 x 4,5in
5. *Recommended Battery* : 22 x 15mm or 22 x 12mm
6. *Recommended ESC* : 15A OPTO

Pada ujung setiap *frame* terpasang 4 buah motor *brushless* dan *propeller*, motor ini yang akan memutar *propeller*. Dengan mengatur kecepatan putaran dan mengatur arah putaran kedalam, sehingga menghilangkan gaya sentrifugal disekitar quadcopter yang mengakibatkan quadcopter bisa terangkat/terbang dan bermanuver. Ada 4 pengendalian dasar kecepatan putaran *propeller*. Pengendalian *throttle propeller*, pengendalian *pitch propeller*, pengendalian *yaw propeller*, dan pengendalian *roll propeller*.

### 2.2.1 Throttle Control

Pengendalian ini bertujuan untuk melakukan *vertical take-off* dan *landing*, bergerak ke atas dan ke bawah. Dalam Gambar 2.2 dapat dilihat putaran tebal berwarna merah menandakan *propeller* berputar dengan cepat yang mengakibatkan quadcopter akan terangkat (*take-off*), sedangkan putaran tipis berwarna hijau menandakan *propeller* berputar dengan lambat yang mengakibatkan quadcopter akan turun/mendarat (*landing*).



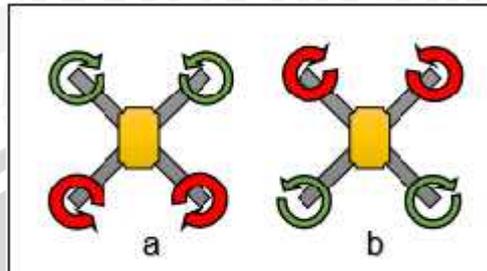
Gambar 2.2 Pengaturan *Throttle Control*. (a) Quadcopter *take off* (b) Quadcopter *landing*

Sumber: Asrul (2014)

### 2.2.2 Pitch Control

Pengendalian ini bertujuan agar quadcopter bergerak maju dan bergerak mundur. Dalam Gambar 2.3.a menampilkan ilustrasi quadcopter bergerak maju, dengan 2 *propeller* pada posisi depan quadcopter yang ditandai dengan warna merah tebal berputar dengan cepat untuk mendorong quadcopter bergerak maju dan 2 *propeller* pada posisi belakang

quadcopter yang ditandai dengan warna hijau berputar dengan lambat. Sedangkan dalam Gambar 2.3.b menampilkan ilustrasi quadcopter bergerak mundur, dengan 2 *propeller* pada posisi belakang quadcopter yang ditandai dengan warna merah tebal berputar dengan cepat untuk mendorong quadcopter bergerak mundur dan 2 *propeller* pada posisi depan quadcopter yang ditandai dengan warna hijau berputar dengan lambat.

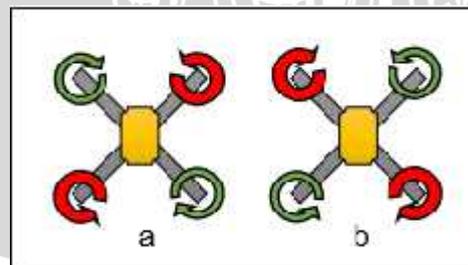


**Gambar 2.3 Pengaturan Pitch Propeller.** (a) Quadcopter bergerak maju (b) Quadcopter bergerak mundur

Sumber: Asrul (2014)

### 2.2.3 Yaw Control

*Yaw control* adalah pengendalian pada *yaw propeller* yang bertujuan agar quadcopter bergerak berputar ke kiri dan bergerak berputar ke kanan. Dalam Gambar 2.4.a menampilkan ilustrasi pergerakan quadcopter bergerak memutar ke kiri dengan 2 *propeller* yang berputar dengan cepat dengan arah putaran ke kiri yang ditandai dengan warna merah tebal dan 2 *propeller* lain yang berputar lambat dengan arah putaran ke kanan yang ditandai dengan warna hijau pada gambar. Sedangkan dalam Gambar 2.4.b menampilkan ilustrasi pergerakan quadcopter memutar ke kanan dengan 2 *propeller* yang berputar dengan cepat dengan arah putaran ke kanan yang ditandai dengan warna merah tebal dan 2 *propeller* lain yang berputar lambat dengan arah putaran ke kiri yang ditandai dengan warna hijau.



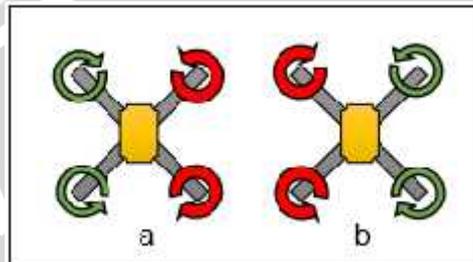
**Gambar 2.4 Pengaturan Yaw Propeller.** (a) Quadcopter bergerak memutar ke kiri (b) Quadcopter bergerak memutar ke kanan

Sumber: Asrul (2014)

### 2.2.4 Roll Control

*Roll control* adalah pengendalian pada *roll propeller* yang bertujuan agar quadcopter bergerak berguling ke kiri dan ke kanan. Dalam Gambar 2.5.a menampilkan ilustrasi

quadcopter bergerak berguling ke kiri dengan 2 *propeller* bagian kiri yang ditandai dengan warna merah tebal berputar dengan cepat namun arah berputarnya saling berlawanan dan 2 *propeller* bagian kanan yang ditandai dengan warna hijau berputar dengan lambat dengan arah yang saling berlawanan juga. Sedangkan dalam Gambar 2.5.b menampilkan ilustrasi bergerak berguling ke kanan dengan 2 *propeller* bagian kanan yang ditandai dengan warna merah tebal berputar dengan cepat namun arah berputarnya saling berlawanan dan 2 *propeller* bagian kiri yang ditandai dengan warna hijau berputar dengan lambat dengan arah yang saling berlawanan juga.



**Gambar 2.5. Pengaturan Roll Propeller.** (a) Quadcopter bergerak berguling ke kiri (b) Quadcopter bergerak berguling ke kanan

Sumber: Asrul (2014)

### 2.3 Mikrokontroler

Mikrokontroler merupakan sebuah *processor* yang digunakan untuk kepentingan kontrol. Meskipun mempunyai bentuk yang jauh lebih kecil dari suatu komputer pribadi dan *computer mainframe*, mikrokontroler dibangun dari elemen-elemen dasar yang sama. Seperti umumnya komputer, mikrokontroler adalah alat yang mengerjakan instruksi-instruksi yang diberikan kepadanya. Artinya, bagian terpenting dan utama dari suatu sistem terkomputerisasi adalah program itu sendiri yang dibuat oleh seorang *programmer*.

Beberapa fitur yang umumnya ada di dalam mikrokontroler adalah sebagai berikut:

- RAM ( *Random Access Memory* )

RAM digunakan oleh mikrokontroler untuk menyimpan *variable*. Memori ini bersifat *volatile* yang berarti akan kehilangan semua datanya jika tidak mendapatkan catu daya.

- ROM ( *Read Only Memory* )

ROM seringkali disebut sebagai kode memori karena berfungsi untuk tempat penyimpanan program yang akan diberikan oleh *user*.

- Register

Merupakan tempat penyimpanan nilai-nilai yang akan digunakan dalam proses yang telah disediakan oleh mikrokontroler.

- *Special Function Register*

Merupakan *register* khusus yang berfungsi untuk mengatur jalannya mikrokontroler. *Register* ini terletak pada RAM.

- *Pin Input dan Output*

*Pin input* adalah bagian yang berfungsi sebagai penerima sinyal dari luar, pin ini dapat dihubungkan ke berbagai media *input* seperti *keypad*, sensor, dan sebagainya. *Pin output* adalah bagian yang berfungsi untuk mengeluarkan sinyal dari hasil proses algoritma mikrokontroler.

- *Interrupt*

*Interrupt* merupakan bagian dari mikrokontroler yang berfungsi sebagai bagian yang dapat melakukan interupsi, sehingga ketika program utama sedang berjalan, program utama tersebut dapat diinterupsi dan menjalankan program interupsi terlebih dahulu.

## 2.4 ATmega328p

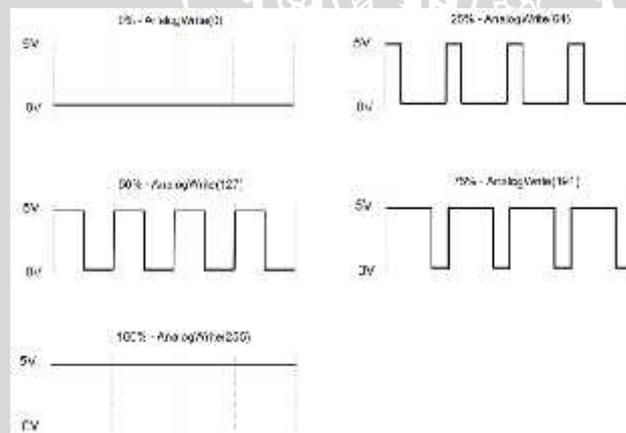
ATmega328p adalah mikrokontroler keluaran dari Atmel yang mempunyai arsitektur *Reduce Instruction Set Computer* (RISC) yang dimana setiap proses eksekusi data lebih cepat dari pada arsitektur *Completed Instruction Set Computer* (CISC). Mikrokontroler ini memiliki beberapa fitur antara lain:

- 130 macam instruksi yang hampir semuanya dieksekusi dalam satu siklus *clock*.
- 32 x 8-bit *register* serbaguna.
- Kecepatan mencapai 16 MIPS dengan *clock* 16 MHz.
- Memiliki EEPROM (*Electrically Erasable Programmable Read Only Memory*) sebesar 1KB sebagai tempat penyimpanan data *semi-permanent* karena EEPROM tetap dapat menyimpan data meskipun catu daya dimatikan.
- Memiliki SRAM (*Static Random Access Memory*) sebesar 2KB.
- Memiliki pin I/O digital sebanyak 14 pin 6 diantaranya PWM (*Pulse Width Modulation*) output.
- *Master / Slave SPI Serial interface*.

Mikrokontroler ATmega328p memiliki arsitektur Harvard, yaitu memisahkan memori untuk kode program dan memori untuk data sehingga dapat memaksimalkan kerja dan *parallelism*. Instruksi-instruksi dalam memori program dieksekusi dalam satu alur tunggal, dimana pada saat satu instruksi dikerjakan instruksi berikutnya sudah diambil dari

memori program. Konsep inilah yang memungkinkan instruksi-instruksi dapat dieksekusi dalam setiap satu siklus *clock*.

ATmega328p memiliki 6 pin I/O PWM. PWM adalah singkatan dari Pulse Width Modulation. Pada Arduino Uno, sinyal PWM adalah sinyal yang beroperasi pada frekuensi 500Hz dan pin yang bisa digunakan untuk PWM adalah pin yang memiliki simbol tilde (~), yaitu pin 3, 5, 6, 9, 10, dan pin 11. PWM pada arduino bekerja pada frekuensi 500Hz, artinya 500 siklus/ketukan dalam satu detik. Untuk setiap siklus, kita dapat diberikan nilai dari 0 hingga 255. Ketika diberikan nilai 0, berarti pada pin tersebut tidak akan pernah bernilai 5 volt (pin selalu bernilai 0 volt). Sedangkan jika diberikan nilai 255, maka sepanjang siklus akan bernilai 5 volt (tidak pernah 0 volt). Jika kita diberikan nilai 127 (dengan menganggap setengah dari 0 hingga 255, atau 50% dari 255), maka setengah siklus akan bernilai 5 volt, dan setengah siklus lagi akan bernilai 0 volt. Sedangkan jika jika memberikan 25% dari 255 ( $1/4 * 255$  atau 64), maka  $1/4$  siklus akan bernilai 5 volt, dan  $3/4$  sisanya akan bernilai 0 volt, dan ini akan terjadi 500 kali dalam 1 detik. Visualisasi siklus PWM ditunjukkan dalam Gambar 2.6.



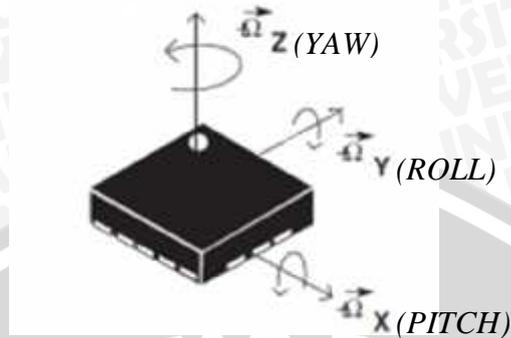
**Gambar 2.6 Siklus PWM**

Sumber: dokumen pribadi

## 2.5 Sensor Gyroscope L3GD20H

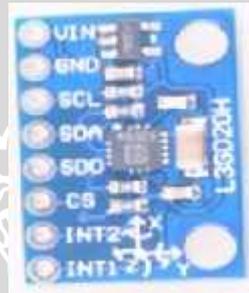
Sensor *gyroscope* adalah suatu unit dalam modul elektronik yang mengumpulkan data kecepatan angular dan akselerasi linear yang kemudian dikirim ke *Central Processing Unit* (CPU) untuk mendapatkan data keberadaan dan pergerakan suatu benda. Sensor *gyroscope* merupakan sensor yang berfungsi untuk mendeteksi perubahan sudut atau mempertahankan posisi terhadap perubahan gravitasi bumi. Sensor *gyroscope* memiliki keluaran berupa perubahan kecepatan sudut dari sumbu x, sumbu y dan sumbu z. Sumbu sudut rotasi dari sensor *gyroscope* ditunjukkan dalam Gambar 2.7.

Tegangan masukan yang dibutuhkan agar sensor L3GD20H aktif adalah tegangan DC 2.5 V sampai 5.5 V. Sedangkan untuk bentuk fisik sensor L3GD20H ditunjukkan dalam Gambar 2.8.



**Gambar 2.7 Sumbu sudut rotasi gyroscope**

Sumber: [www.eeweb.com](http://www.eeweb.com)



**Gambar 2.8 Sensor Gyroscope L3GD20H**

Sumber: dokumen pribadi

Fungsi masing-masing pin pada sensor L3GD20H dapat dilihat dalam Tabel 2.1.

**Tabel 2.1 Fungsi pin sensor L3GD20H**

No	Pin	Fungsi
1	VIN	2.5V sampai 5V power input dari board arduino
2	GND	Ground
3	SCL	I2C serial clock (SCL)
4	SDA	I <sup>2</sup> C serial data (SDA)
5	SDO	SPI serial data output (SDO)
6	CS	I <sup>2</sup> C/SPI mode selection (1: SPI idle mode / I2C communication enabled; 0: SPI communication mode / I <sup>2</sup> C disabled)
7	INT2	fifo interrupt (FIFO threshold/overflow/empty)
8	INT1	Programmable interrupt

Sumber: *datasheet* sensor L3GD20H

## 2.6 Motor *Brushless*

*Brushless* DC motor listrik (BLDC motor, motor BL) juga dikenal sebagai motor elektronik *commutated* (ECM, motor EC) adalah motor sinkron yang didukung oleh sumber listrik DC melalui inverter/*switching* power supply yang terintegrasi, yang menghasilkan sinyal listrik AC untuk menggerakkan motor. Dalam konteks ini, AC, arus bolak-balik, tidak berarti gelombang sinusoidal, melainkan arus bi – directional dengan tidak ada pembatasan pada gelombang. Sensor tambahan dan elektronik mengendalikan amplitudo inverter *output* dan gelombang (karenanya persen DC penggunaan/efisiensi bus) serta frekuensi (yaitu kecepatan rotor). Motor bagian dari motor *brushless* sering berjenis magnet permanen motor sinkron, tetapi juga bisa menjadi motor induksi. Pada skripsi ini menggunakan motor *brushless* DJI 2212/920Kv. Bentuk fisik motor *brushless* ditunjukkan dalam Gambar 2.9.



Gambar 2.9 Motor *brushless* DJI 2212/920Kv

Sumber: dokumen pribadi

Terdapat cara untuk mengetahui jenis motor *brushless* berapa *watt* yang harus digunakan adalah dengan cara menghitung total berat quadcopter agar motor *brushless* yang digunakan sesuai dengan kebutuhan dan dapat mengangkat quadcopter terbang secara vertikal. Dengan menggunakan aturan “*watts per pound*” jenis motor *brushless* yang sesuai spesifikasi quadcopter bisa didapatkan namun karena di Indonesia lebih mengenal dengan satuan gram maka satuan *pound* dapat dikonversi, 1 *pound* = 453,592 gram atau dapat dibulatkan menjadi 450 gram sehingga aturan sebelumnya dapat diubah menjadi “*watts per 450 gram*” ditunjukkan dalam persamaan (2.1).

$$\frac{E}{T} \frac{Q_1}{Q_2} \left( \frac{g}{g} \right) \times 100 = (\text{watt yang dibutuhkan motor}) \quad (2.1)$$

## 2.7 *Electronic Speed Controller* (ESC)

*Electronic Speed Controller* (ESC) adalah alat yang digunakan untuk regulator kecepatan listrik yang mempengaruhi arus motor yang digunakan. Atau bisa juga diartikan sebagai sebuah modul rangkaian elektronik yang berfungsi untuk mengatur putaran motor dengan mengatur suplai arus yang disesuaikan dengan kebutuhan motor *brushless*. ESC yang

akan digunakan adalah ESC Hobbywing Platinum Series 30A OPTO. Bentuk fisik ESC Hobbywing Platinum Series 30A OPTO ditunjukkan dalam Gambar 2.10.



**Gambar 2.10** ESC Hobbywing 30A OPTO

Sumber: dokumen pribadi

## 2.8 Remote Control

*Remote control* merupakan bagian yang berinteraksi langsung dengan pengguna untuk memberikan sinyal perintah-perintah untuk menggerakkan quadcopter dalam arah gerakan arah naik, turun, maju, mundur, kiri dan kanan. Pada kondisi *outdoor* penggunaan frekuensi jika dibandingkan dengan menggunakan sinyal infrared sinar matahari sering memberikan gangguan terhadap sinyal infrared yang sangat mempengaruhi proses kendali, maka gelombang radio merupakan pilihan tepat. Selain itu penggunaan gelombang radio mempunyai keunggulan dimana data yang dipancarkan dapat dikirim pada jarak yang cukup jauh bahkan dapat menembus halangan. Remote Control yang digunakan adalah Flysky FS-T6 *transmitter* dan sepaket dengan *receivernya* yaitu FS 6ch *receiver*. Bentuk fisik *transmitter* Flysky FS-T6 dan *receiver* FS 6ch ditunjukkan dalam Gambar 2.11.



**Gambar 2.11** Transmitter Flysky FS-T6 dan receiver FS 6ch

Sumber: dokumen pribadi

## 2.9 Baterai

Baterai adalah alat yang vital untuk quadcopter, karena baterai berfungsi menyalurkan sumber listrik untuk otak, motor, sensor-sensor, dan transmitter pada quadcopter. Baterai yang digunakan adalah baterai Li-Po. Baterai Li-Po berbeda dengan baterai Li-Ion yang sangat populer. Perbedaan baterai Li-Po dan Li-Ion terletak pada penghantar arus listrik (Elektrolit) yang ada pada kedua jenis baterai tersebut. Baterai Li-Po adalah singkatan *Lithium Polymer*, baterai ini bersifat cair, menggunakan elektrolit polimer yang padat, dan mampu menghantarkan daya lebih cepat dan jenis baterai ini adalah hasil pengembangan dari *Lithium Ion*. Baterai Li-Po ini disebut sebagai baterai ramah lingkungan.

Baterai Li-Po memiliki banyak kelebihan dibanding tipe Li-Ion yaitu :

1. Ramah Lingkungan.
2. Fleksibel bisa dibuat berdasarkan kebutuhan.
3. Lebih Aman.
4. Lebih ringan.

Namun baterai Li-Po juga memiliki kekurangan yaitu :

1. Biaya manufaktur mahal.
2. Harga baterai juga mahal karena biaya untuk energi ini juga mahal.
3. Butuh perawatan khusus untuk isi ulang, seperti jangan sampai baterai habis baru di isi ulang.
4. Usia Baterai lebih pendek.

Baterai yang akan digunakan dalam skripsi ini adalah baterai Lipo TIGER 3500 Mah 3S. Bentuk fisik dari baterai Lipo TIGER 3500 Mah 3S ditunjukkan dalam Gambar 2.12.



Gambar 2.12 Baterai Lipo TIGER 3500 Mah 3S

Sumber: dokumen pribadi

## 2.10 Modul *Bluetooth* HC-05

*Bluetooth* Module HC-05 merupakan modul komunikasi nirkabel pada frekuensi 2.4GHz dengan pilihan koneksi bisa sebagai *slave*, ataupun sebagai *master*. Sangat mudah digunakan dengan mikrokontroler untuk membuat aplikasi *wireless*. *Interface* yang digunakan adalah serial RXD, TXD, VCC dan GND. Built in LED sebagai indikator koneksi *bluetooth*. Tegangan input antara 3.6 ~ 6V. Modul ini dapat langsung dihubungkan ke berbagai macam mikrokontroler (khusus Arduino, 8051, 8535, AVR, PIC, ARM, MSP430, etc.). Jarak efektif jangkauan sebesar 10 meter, meskipun dapat mencapai lebih dari 10 meter, namun kualitas koneksi makin berkurang. Bentuk fisik dari modul *Bluetooth* HC-05 ditunjukkan dalam Gambar 2.13 dan fungsi masing-masing pin pada modul *Bluetooth* HC-05 dapat dilihat dalam Tabel 2.2.



Gambar 2.13 Modul *Bluetooth* HC-05

Sumber: dokumen pribadi

Tabel 2.2 Fungsi pin modul *Bluetooth* HC-05

No	Pin	Fungsi
1	VCC	3.6V sampai 6V power input dari board arduino
2	GND	Ground
3	RXD	Menerima data
4	TXD	Mengirim data
5	STATE	Memberikan informasi apakah modul terhubung atau tidak dengan perangkat lain
6	EN	Mengaktifkan mode AT Command Setup pada modul HC-05

Sumber: *datasheet* modul *bluetooth* HC-05

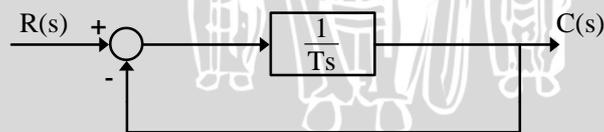
## 2.11 Kestabilan

Sebuah sistem dikatakan tidak stabil jika tanggapannya terhadap suatu masukan menghasilkan osilasi yang keras atau bergetar pada suatu amplitudo/harga tertentu. Sebaliknya suatu sistem disebut stabil jika sistem tersebut akan tetap dalam keadaan diam atau berhenti kecuali jika dirangsang (dieksitasi oleh suatu fungsi masukan dan akan kembali dalam keadaan diam jika eksitasi tersebut dihilangkan). Ketidakstabilan merupakan suatu keadaan yang tidak menguntungkan bagi suatu sistem lingkaran tertutup (*close loop*) sedangkan pada suatu sistem lingkaran terbuka dapat tidak harus stabil. Jelas untuk memperoleh nilai yang memberikan manfaat, praktis sebuah sistem kendali harus stabil. Masukan sistem tidak memberikan pengaruh terhadap kestabilan suatu sistem sehingga jika sistem tersebut stabil terhadap suatu masukan maka sistem akan stabil juga untuk masukan lain. Kestabilan hanya bergantung pada karakteristik sistem itu sendiri.

Tanggapan suatu sistem stabil dapat dikenali dari adanya peralihan yang menurun menuju nol terhadap pertambahan waktu. Ini berarti bahwa untuk mendapatkan sebuah sistem yang stabil, koefisien-koefisien dari suku eksponensial yang terdapat dalam tanggapan peralihan tersebut harus merupakan bilangan-bilangan nyata yang negatif atau bilangan kompleks dimana bagian nyata adalah negatif.

## 2.12 Output Sistem Orde Satu

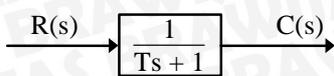
Blok diagram sistem orde satu dapat dilihat dalam Gambar 2.14 yang mempunyai fungsi alih sebagai berikut



Gambar 2.14 Sistem orde satu

$$\frac{C(s)}{R(s)} = \frac{1/Ts}{1 + 1/Ts} \quad (2.2)$$

$$= \frac{1}{Ts + 1}$$



### 2.13 Output Unit Step Pada Sistem Orde Satu

Jika sistem orde satu diberi masukan unit step  $r(t)=1$ , dari Transformasi Laplace fungsi ramp adalah  $R(s) = \frac{1}{s}$ . Keluaran sistem adalah sebagai berikut

$$\begin{aligned} C(s) &= \frac{1}{Ts+1} R(s) \\ C(s) &= \frac{1}{Ts+1} \frac{1}{s} \\ &= \frac{1}{s(Ts+1)} \end{aligned} \quad (2.3)$$

$$\begin{aligned} c(t) &= \mathcal{L}^{-1}[C(s)] \\ c(t) &= 1 - e^{-t/T} \quad (t \geq 0) \end{aligned} \quad (2.4)$$

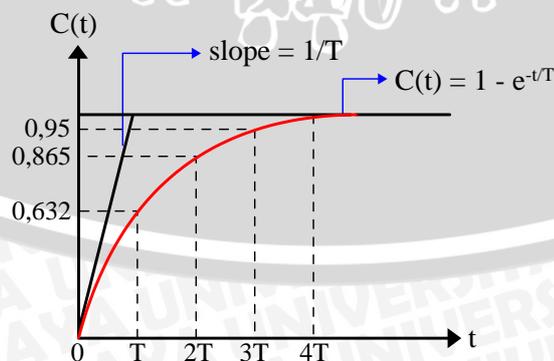
Persamaan keluaran tersebut menyatakan bahwa keluaran  $c(t)$  mula-mula nol kemudian akhirnya menjadi satu (dapat dilihat dalam Gambar 2.15). Salah satu karakteristik penting output eksponensial  $c(t)$  tersebut adalah bahwa pada  $t = T$ , maka

$$c(t) = 1 - e^{-1} \cong 0.632 = \frac{2}{3}$$

$$\begin{aligned} T &= \text{time constant / konstanta waktu sistem} \\ &= \frac{2}{3} \text{ harga akhir} \end{aligned}$$

Konstanta waktu  $T$  yang lebih kecil akan mempercepat output sistem. Karakteristik penting lainnya pada kurva output eksponensial adalah kemiringan garis singgung / gradien pada  $t = 0$  adalah  $1/T$ , karena

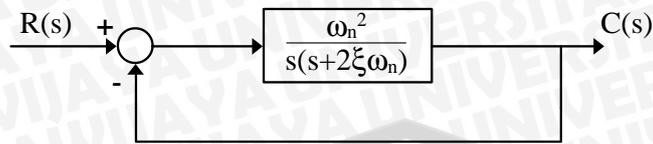
$$\frac{dc}{dt} = \frac{1}{T} e^{-t/T} \Big|_{t=0} = \frac{1}{T}$$



Gambar 2.15 Output unit step sistem orde satu

## 2.14 Output Sistem Orde Dua

Blok diagram sistem orde dua (lihat Gambar 2.16), dengan fungsi alihnya adalah sebagai berikut:



Gambar 2.16 Sistem orde dua

$$\begin{aligned} \frac{C(s)}{R(s)} &= \frac{\zeta_n^2}{s(s+2\zeta_n)} \cdot \frac{1}{1 + \frac{\zeta_n^2}{s(s+2\zeta_n)}} \\ &= \frac{\zeta_n^2}{s(s+2\zeta_n) + \zeta_n^2} \\ &= \frac{\zeta_n^2}{s^2 + 2\zeta_n s + \zeta_n^2} \end{aligned} \quad (2.5)$$

Akar-akar penyebut fungsi alih atau persamaan karakteristik adalah

$$\begin{aligned} s_{1,2} &= \frac{-2\zeta_n \pm \sqrt{(2\zeta_n)^2 - 4\zeta_n^2}}{2} \\ s_{1,2} &= -\zeta_n \pm \zeta_n \sqrt{\zeta_n^2 - 1} \\ s_{1,2} &= -\zeta_n \pm j\zeta_n \sqrt{1 - \zeta_n^2} \\ s_{1,2} &= -\zeta_n \pm j\zeta_d \end{aligned} \quad (2.6)$$

dimana

$\zeta$  = rasio peredaman sistem (*damping ratio*)

$\zeta_n$  = frekuensi natural/alamiah tak teredam

$\zeta_d$  = frekuensi natural/alamiah teredam

Kelakuan dinamik sistem orde dua dapat digambarkan dalam suku dua parameter dan  $\zeta_n$ . Jika ( $0 < \zeta < 1$ ), maka pole loop tertutup merupakan konjugat kompleks dan berada pada bidang  $s$  sebelah kiri. Dalam hal ini, sistem dikatakan dalam peredaman dan tanggapan peralihan berosilasi. Jika ( $\zeta = 1$ ), maka sistem dikatakan teredam kritis. Sistem terlalu teredam berhubungan dengan ( $\zeta > 1$ ). Tanggapan peralihan sistem teredam kritis dan sistem terlalu teredam tidak berosilasi. Jika  $\zeta = 0$ , tanggapan peralihan tidak muncul.

Pada sistem orde dua seperti terlihat dalam Gambar 2.15, berdasarkan output sistem dengan masukan unit step akan terdapat tiga keadaan yang berbeda yaitu keadaan teredam ( $0 < \zeta < 1$ ), teredam kritis ( $\zeta = 1$ ), dan sistem terlalu teredam ( $\zeta > 1$ ).

#### 2.14.1 Keadaan Kurang Teredam / Underdamped ( $0 < \zeta < 1$ )

$$\frac{C(s)}{R(s)} = \frac{\frac{2}{n}}{(s^2 + 2\zeta \frac{2}{n}s + \frac{2}{n})s} \quad (2.7)$$

Jika sistem diberi input berupa unit step atau  $R(s) = \frac{1}{s}$ , maka:

$$C(s) = \frac{\frac{2}{n}}{(s^2 + 2\zeta \frac{2}{n}s + \frac{2}{n})s} \quad (2.8)$$

Dari Tabel Transformasi Laplace didapatkan

$$c(t) = 1 - \frac{1}{\sqrt{1-\zeta^2}} e^{-\zeta t} \sin(\omega_d t + \phi)$$

$$\omega_d = \arctan \frac{\sqrt{1-\zeta^2}}{\zeta}$$

Jika  $\zeta_d = \zeta_n \sqrt{1-\zeta^2}$ ; maka

$$c(t) = 1 - \frac{e^{-\zeta t}}{\sqrt{1-\zeta^2}} \sin\left(\omega_d t + \arctan \frac{\sqrt{1-\zeta^2}}{\zeta}\right) \quad t \geq 0 \quad (2.9)$$

Output sistem tersebut juga bisa diperoleh dengan menggunakan Transformasi Laplace balik jika  $C(s)$  ditulis dalam bentuk berikut:

$$\begin{aligned} C(s) &= \frac{1}{s} \frac{s + 2\zeta \frac{2}{n}}{s^2 + 2\zeta \frac{2}{n}s + \frac{2}{n}} \\ &= \frac{1}{s} \frac{s + \zeta_n}{(s + \zeta_n)^2 + \zeta_d^2} - \frac{\zeta_n}{(s + \zeta_n)^2 + \zeta_d^2} \end{aligned}$$

$$\mathcal{L}^{-1} \left[ \frac{s + \zeta_n}{(s + \zeta_n)^2 + \zeta_d^2} \right] = e^{-\zeta_n t} \cos \zeta_d t$$

$$\mathcal{L}^{-1} \left[ \frac{\zeta_n}{(s + \zeta_n)^2 + \zeta_d^2} \right] = e^{-\zeta_n t} \sin \zeta_d t \quad (2.10)$$

oleh karena itu, transformasi laplace balik dari persamaan

$$C(s) = \frac{\frac{2}{n}}{(s^2 + 2\zeta \frac{2}{n}s + \frac{2}{n})s}$$

diperoleh sebagai

$$\mathcal{L}^{-1}[C(s)] = c(t)$$

$$c(t) = 1 - e^{-\zeta \omega_n t} \left( \cos \omega_d t + \frac{\omega_n}{\omega_d} \sin \omega_d t \right)$$

$$c(t) = 1 - \frac{e^{-\zeta \omega_n t}}{\sqrt{1-\zeta^2}} \sin \left( \omega_d t + \arctan \frac{\omega_n}{\omega_d} \sqrt{1-\zeta^2} \right) \quad t \geq 0 \quad (2.11)$$

Sinyal kesalahan / error adalah  $e(t) = r(t) - c(t)$ , dimana

$$r(t) = 1$$

dan

$$c(t) = 1 - e^{-\zeta \omega_n t} \left( \cos \omega_d t + \frac{\omega_n}{\omega_d} \sin \omega_d t \right)$$

sehingga

$$e(t) = e^{-\zeta \omega_n t} \left( \cos \omega_d t + \frac{\omega_n}{\omega_d} \sin \omega_d t \right) \quad t \geq 0$$

$$\text{jika } \zeta = 0 \Rightarrow c(t) = 1 - \cos \omega_n t$$

### 2.14.2 Teredam Kritis / Critically Damped ( $\zeta = 1$ )

Dalam hal ini apabila dua pole  $\frac{C(s)}{R(s)}$  hampir sama, maka sistem dapat didekati dengan bentuk teredam kritis. Jika input berupa unit step dimana  $R(s) = 1/s$  maka:

$$C(s) = \frac{\omega_n^2}{(s + \frac{\omega_n}{2})^2 s} \quad (2.12)$$

$$c(t) = 1 - e^{-\omega_n t} (1 + \omega_n t) \quad t \geq 0$$

### 2.14.3 Terlalu Teredam / Overdamped ( $\zeta > 1$ )

Dalam hal ini pole  $\frac{C(s)}{R(s)}$  adalah bilangan nyata / real negatif yang tidak sama.

Jika input berupa unit step dimana  $R(s) = 1/s$  dan  $C(s)$  dapat ditulis dengan :

$$C(s) = \frac{\omega_n^2}{(s + \zeta \omega_n)^2 s} \quad (2.13)$$

$$C(s) = \frac{1}{\left(s + \frac{n}{2\sqrt{2-1}} + \frac{n}{2\sqrt{2-1}}\right) \left(s + \frac{n}{2\sqrt{2-1}} - \frac{n}{2\sqrt{2-1}}\right)}$$

$$c(t) = 1 + \frac{n}{2\sqrt{2-1}} \left( \frac{e^{-s_1 t}}{s_1} - \frac{e^{-s_2 t}}{s_2} \right) \quad t \geq 0$$

dengan  $s_1 = \left( +\sqrt{2-1} \right)_n$

$$s_2 = \left( -\sqrt{2-1} \right)_n \quad (2.14)$$

Tanggapan  $c(t)$  terdiri dari dua suku eksponensial menurun.

### 2.15 Tanggapan Peralihan

Sistem dengan tenaga tidak dapat memberikan tanggapan seketika dan akan menunjukkan tanggapan peralihan walaupun diberi masukan ataupun gangguan. Karakteristik unjuk kerja sistem kendali yang diinginkan dicirikan oleh suku tanggapan peralihan terhadap masukan unit step karena hal itu mudah dilakukan dan cukup drastis. Jika tanggapan terhadap masukan unit step diketahui, secara matematis dapat dihitung tanggapan untuk masukan yang lain.

Tanggapan peralihan sistem kendali selalu menunjukkan osilasi teredam sebelum mencapai keadaan mantapnya, hal ini juga menunjukkan bahwa sistem tersebut mempunyai rasio peredaman ( $0 < \zeta < 1$ ) yang juga berarti bahwa sistem tersebut merupakan sistem yang kurang teredam / underdamped.

Tanggapan peralihan sistem kendali terhadap masukan unit step umumnya dikelompokkan sebagai berikut (lihat Gambar 2.17):

- 1) Delay Time / Waktu Tunda,  $t_d$

Waktu yang dibutuhkan oleh output untuk mencapai setengah harga akhir pada saat lonjakan pertama

- 2) Rise Time / Waktu Naik,  $t_r$

Waktu yang dibutuhkan oleh output agar bertambah dari 10% menjadi 90% dari nilai akhir

- 3) Peak Time / Waktu Puncak,  $t_p$

Waktu yang dibutuhkan oleh output untuk mencapai puncak pertama lonjakan (maksimum)

- 4) Maximum Overshoot / Lonjakan Maksimum,  $M_p$

Merupakan nilai puncak kurva output diukur dari satu

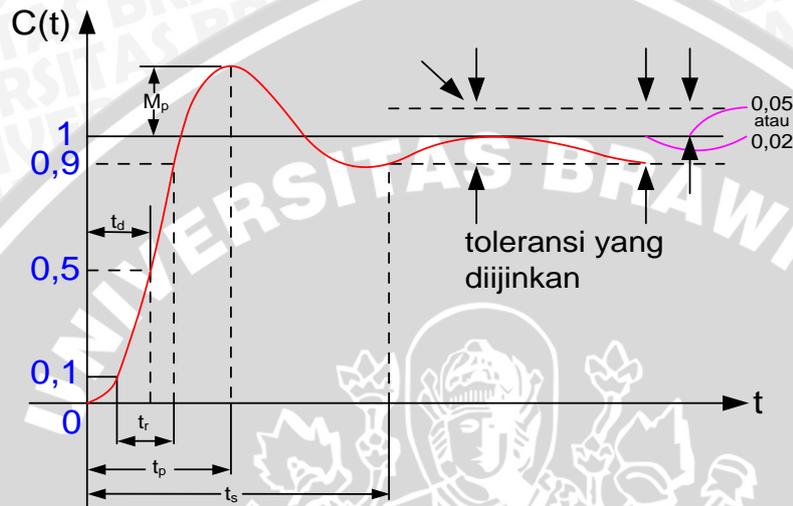
$$M_p = \frac{c(t_p) - c(\infty)}{c(\infty)} \times 100\%$$

dengan  $c(t_p)$  = nilai outputs pada saat lonjakan maksimum.

$c(\infty)$  = nilai outputs pada saat keadaan mantap.

#### 5) Settling Time / Waktu Turun, $t_s$

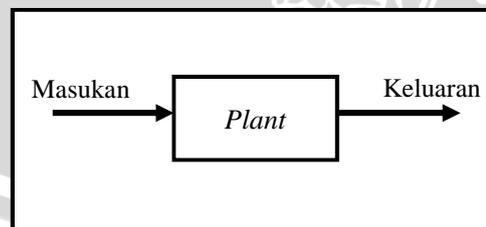
Waktu yang dibutuhkan oleh output untuk mencapai harga tertentu dan tetap dalam range nilai akhir (biasanya 5% atau 2%)



Gambar 2.17 Output Unit Step Sistem Orde Dua

## 2.16 Sistem Kendali

Sistem kendali merupakan hubungan antara komponen yang membentuk sebuah konfigurasi sistem, yang menghasilkan tanggapan sistem yang diharapkan. Jadi harus ada yang dikendalikan, yang merupakan suatu fisis, yang biasa disebut dengan kendalian (*plant*). Blok diagram dari sistem kendali yang sederhana ditunjukkan dalam Gambar 2.18.



Gambar 2.18 Input Output Blok diagram

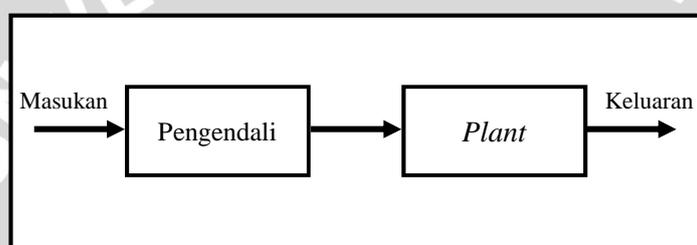
Sumber: Asrul (2014)

Masukan dan keluaran merupakan variabel atau besaran fisis. Keluaran merupakan hal yang dihasilkan oleh kendalian, artinya yang dikendalikan. Sedangkan masukan adalah yang mempengaruhi kendalian, yang mengatur keluaran. Kedua dimensi masukan dan

keluaran tidak harus sama. Sistem kendali adalah proses pengaturan ataupun pengendalian terhadap satu atau beberapa besaran (variabel dan parameter) sehingga berada pada suatu harga atau dalam suatu rangkuman harga (*range*) tertentu. Sistem kendali otomatis merupakan suatu sistem dalam proses kerjanya dikendalikan tanpa adanya campur tangan manusia. Ada dua sistem kendali otomatis, yaitu:

### 2.16.1 Sistem *Open Loop*

Sistem *Open Loop* adalah sistem kendali yang kelurannya tidak berpengaruh terhadap aksi pengendalian. Dengan demikian pada sistem kendali ini, nilai keluaran tidak diumpan-balikkan ke parameter pengendalian. Blok diagram dari sistem kendali *open loop* ditunjukkan dalam Gambar 2.19.



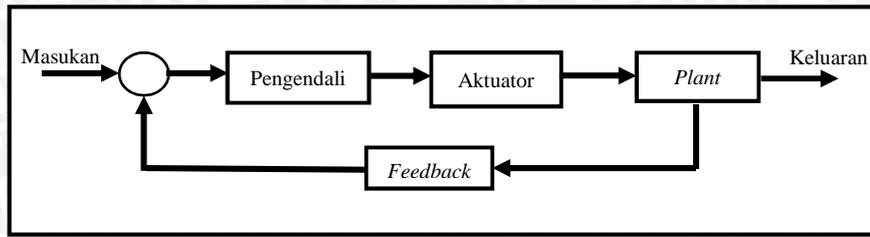
Gambar 2.19 Blok diagram sistem *open loop*

Sumber: Asrul (2014)

### 2.16.2 Sistem *Closed Loop*

Sistem *Closed Loop* adalah sistem kendali yang keluarannya memiliki pengaruh langsung terhadap aksi pengendalian yang dilakukan. Sinyal *error* yang merupakan selisih dari sinyal masukan dan sinyal umpan balik (*feedback*), lalu diumpankan pada komponen pengendalian (*controller*) untuk memperkecil kesalahan sehingga nilai keluaran sistem semakin mendekati harga yang diinginkan/*set point*.

Keuntungan sistem *closed loop* adalah adanya pemanfaatan nilai umpan balik yang dapat membuat respon sistem kurang peka terhadap gangguan eksternal dan perubahan internal pada parameter sistem. Kerugiannya adalah tidak mengambil aksi perbaikan terhadap suatu gangguan sebelum gangguan tersebut mempengaruhi nilai prosesnya. Blok diagram sistem kendali *close loop* ditunjukkan dalam Gambar 2.20.



Gambar 2.20 Blok diagram sistem closed loop

Sumber: Ogata, K. (2010)

### 2.17 Pengendali Proporsional

Pengendali proporsional adalah sebuah pengendali yang memiliki karakteristik mempercepat output. Hubungan antara output pengendali  $u(t)$  dan sinyal *error*  $e(t)$  ditunjukkan dalam persamaan berikut:

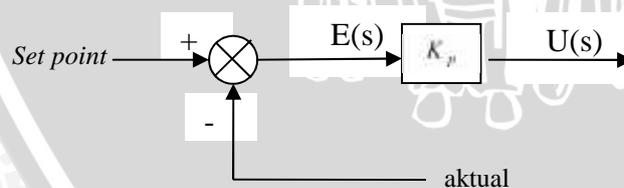
$$u(t) = K_p e(t) \tag{2.15}$$

atau, dalam fungsi alih

$$\frac{U(s)}{E(s)} = K_p$$

dimana  $K_p$  adalah penguatan.

Gambar 2.21 menunjukkan blok diagram yang menggambarkan hubungan antara besaran *set point*, besaran aktual dengan besaran keluaran pengendali proporsional  $u(t)$ . Sinyal keasalahan (*error*) merupakan selisih antara besaran *set point* dengan besaran aktualnya. Selisih ini akan mempengaruhi pengendali, untuk mengeluarkan sinyal positif (mempercepat pencapaian harga *set point*) atau negatif (memperlambat tercapainya harga yang diinginkan).



Gambar 2.21 Blok diagram pengendali proporsional

Sumber: Ogata, K. (2010)

Apapun wujud mekanisme yang sebenarnya dan apapun bentuk daya penggerakannya, pengendali proporsional pada dasarnya merupakan penguat dengan penguatan yang dapat diatur (Ogata K., 2010). Ciri-ciri pengendali proposional harus diperhatikan ketika pengendali tersebut diterapkan pada suatu sistem. Secara eksperimen, pengguna pengendali proporsional harus memperhatikan ketentuan-ketentuan berikut ini :

1. Jika nilai  $K_p$  kecil, pengendali proposional hanya mampu melakukan koreksi kesalahan yang kecil, sehingga akan menghasilkan respon sistem yang lambat.
2. Jika nilai  $K_p$  dinaikan, respon sistem menunjukkan semakin cepat mencapai *set point* dan keadaan stabil.
3. Namun jika nilai  $K_p$  diperbesar sehingga mencapai harga yang berlebihan, akan mengakibatkan sistem bekerja tidak stabil, atau respon sistem akan berosilasi.

### 2.18 Pengendali Integral

Pengendali integral berfungsi menghasilkan respon sistem yang memiliki kesalahan keadaan mantap nol. Jika sebuah *plant* tidak memiliki unsur integrator ( $1/s$ ), pengendali proposional tidak akan mampu menjamin keluaran sistem dengan kesalahan keadaan mantapnya nol. Dengan pengendali integral, respon sistem dapat diperbaiki, yaitu mempunyai kesalahan keadaan mantapnya nol.

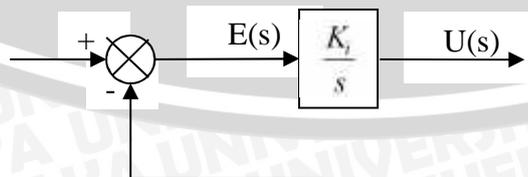
Pengendali integral memiliki karakteristik seperti halnya sebuah integral. Keluaran sangat dipengaruhi oleh perubahan yang sebanding dengan nilai sinyal kesalahan. Keluaran pengendali ini merupakan penjumlahan yang terus menerus dari perubahan masukannya. Kalau sinyal kesalahan tidak mengalami perubahan, keluaran akan menjaga keadaan seperti sebelum terjadinya perubahan masukan. Pengendali integral (I) memiliki kemampuan untuk mengurangi *offset* yang diakibatkan oleh pengendali proporsional. Output pengendali  $u(t)$  diubah dengan laju yang sebanding dengan *error*  $e(t)$ . Persamaan pengendali integral (I) ditunjukkan dalam persamaan (2-16) sedangkan Gambar 2.22 menunjukkan blok diagram antara besaran kesalahan dengan keluaran suatu pengendali integral (Ogata K., 2010).

$$\frac{d}{dt} \int K_I e(t) dt$$

$$u(t) = K_I \int_0^t e(t) dt$$

$$\frac{U(s)}{E(s)} = \frac{K_I}{s}$$

(2.16)



Gambar 2.22 Blok diagram hubungan antara besaran kesalahan dengan pengendali integral

Sumber: Ogata, K. (2010)

Pengendali integral mempunyai beberapa karakteristik berikut ini:

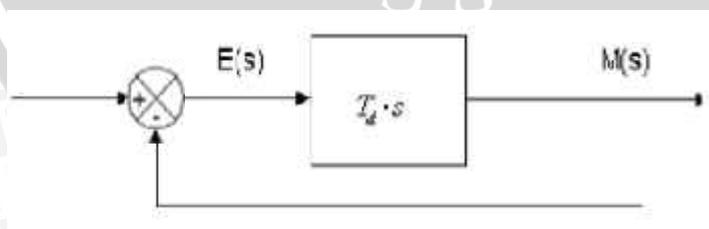
1. Keluaran pengendali membutuhkan selang waktu tertentu, sehingga pengendali integral cenderung memperlambat respon.
2. Ketika sinyal kesalahan berharga nol, keluaran pengendali akan bertahan pada nilai sebelumnya.
3. Jika sinyal kesalahan tidak berharga nol, keluaran akan menunjukkan kenaikan atau penurunan yang dipengaruhi oleh besarnya sinyal kesalahan dan nilai  $K_i$  (Johnson, 1993, 376).
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 pengendali (Guterus, 1994, 7-4).

### 2.19 Pengendali *Differential*

Pengendali *differential* dapat memberikan respon terhadap laju perubahan sinyal kesalahan penggerak dan dapat menghasilkan koreksi sebelum sinyal kesalahan penggerak menjadi terlalu besar. Jadi, pengendali *differential* mendahului sinyal kesalahan penggerak, mengawali aksi koreksi dini, dan cenderung memperbesar kestabilan. Keluaran pengendali *differential* memiliki sifat seperti halnya suatu operasi *differential*. Perubahan yang mendadak pada masukan pengendali, akan mengakibatkan perubahan yang sangat besar dan cepat. Persamaan pengendali *differential* (D) ditunjukkan dalam persamaan (2.17). Sedangkan blok diagram yang menggambarkan hubungan antara sinyal kesalahan dengan keluaran pengendali ditunjukkan dalam Gambar 2.23.

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

$$\frac{U(s)}{E(s)} \approx K_d s \quad (2.17)$$



Gambar 2.23 Blok diagram pengendali *differential*

Sumber: Ogata, K. (2010)

Karakteristik pengendali *differential* 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 pengendali tergantung pada nilai  $T_d$  dan laju perubahan sinyal kesalahan. (Powel, 1994, 184).
3. Pengendali *differential* mempunyai suatu karakter untuk mendahului, sehingga pengendali ini dapat menghasilkan koreksi yang signifikan sebelum pembangkit kesalahan menjadi sangat besar. Jadi pengendali *differential* dapat mengantisipasi pembangkit kesalahan, memberikan aksi yang bersifat korektif, dan cenderung meningkatkan stabilitas sistem (Ogata, 1997, 240).

Berdasarkan karakteristik pengendali tersebut, pengendali *differential* umumnya dipakai untuk mempercepat respon awal suatu sistem, tetapi tidak memperkecil kesalahan pada keadaan stabilnya. Kerja pengendali *differential* hanyalah efektif pada lingkup yang sempit, yaitu pada periode peralihan. Oleh sebab itu pengendali *differential* tidak pernah digunakan tanpa ada pengendali lain sebuah sistem (Sutrisno, 1990).

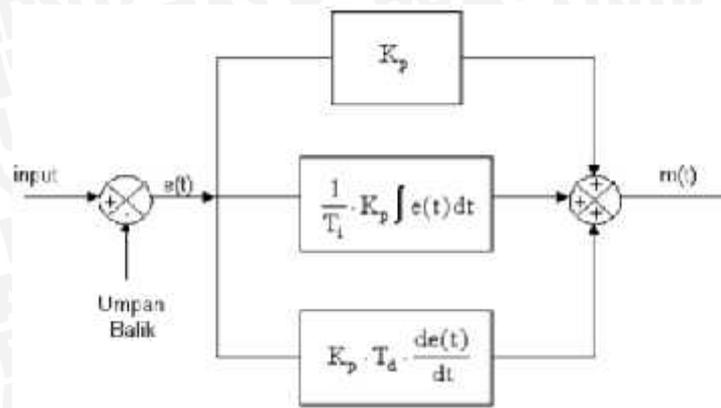
## 2.20 Pengendali Proporsional Integral *Differential* (PID)

Setiap kekurangan dan kelebihan dari masing-masing pengendali P, I, dan D dapat saling menutupi dengan menggabungkan ketiganya secara paralel menjadi pengendali proposional plus integral plus *differential* (pengendali PID). Elemen-elemen pengendali P, I, dan D masing-masing secara keseluruhan bertujuan untuk mempercepat reaksi sebuah sistem, menghilangkan *offset* dan menghasilkan perubahan awal yang besar. Persamaan pengendali PID dalam waktu kontinyu ditunjukkan oleh persamaan (2.18). Sedangkan blok diagram untuk pengendali PID analog ditunjukkan dalam gambar 2.24.

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d e(t)}{dt} \quad (2.18)$$

Keterangan

- $u(t)$  = Keluaran pengendali PID  
 $K_p$  = Konstanta proporsional  
 $K_i$  = Konstanta integral  
 $K_d$  = Konstanta *differential*



Gambar 2.24 Blok diagram pengendali PID analog

Sumber: Ogata, K. (2010)

Pengendali PID merupakan penjumlahan dari keluaran pengendali proporsional, keluaran pengendali integral, dan keluaran pengendali *differential*. Karakteristik pengendali PID sangat dipengaruhi oleh kontribusi besar dari ketiga parameter P, I dan D. Pengaturan konstanta  $K_p$ ,  $T_i$ , dan  $T_d$  akan mengakibatkan penonjolan sifat dari masing-masing elemen. Satu atau dua dari ketiga konstanta tersebut dapat diatur lebih menonjol dibanding yang lain. Konstanta yang menonjol itulah yang akan memberikan kontribusi pengaruh pada respon sistem secara keseluruhan.

