

BAB II

TINJAUAN PUSTAKA

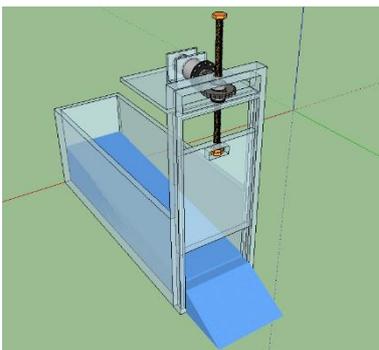
Untuk memudahkan dalam memahami cara kerja rangkaian maupun dasar-dasar perencanaan dari sistem yang dibuat, maka perlu adanya penjelasan dan uraian mengenai teori penunjang yang digunakan dalam penulisan skripsi ini.

Teori penunjang yang akan dibahas dalam bab ini adalah:

- *Plant* berupa miniatur pintu bendung air.
- BTS7960 Driver 43A H-Bridge Drive
- Motor DC 12 V
- Arduino UNO R3
- *Pulse Width Modulation (PWM)*
- Kontroler *Proportional Integral Diferensial (PID)*.
- Metode Ziegler – Nichols.
- Sensor ping

2.1 *Plant*

Plant yang digunakan adalah miniatur pintu bendung yang biasanya digunakan di pembangkit listrik sebagaimana pada Gambar 2.1. Dengan menggunakan sistem kontrol pada pintu bendung penampung, masukan awal sistem yaitu *setpoint* sebesar cm berupa level ketinggian pada miniatur pintu bendung, keluaran sistem berupa level ketinggian. Serta gangguan sistem yang di berikan berupa tekanan debit air pada pintu bendung yang bertambah dikarenakan keadaan alam yang tak menentu seperti curah hujan yang tinggi. Pada plant berikut diharapkan didapatkan error steady state di bawah 2 % serta waktu naik (t_r) di bawah 2 detik agar level ketinggian pada miniatur tetap pada setpoint 6 cm bila terjadi penambahan level ketinggian air pada miniatur



Gambar 2.1 Miniatur Pintu Bendung

2.2 Driver Motor Direct Current (DC) BTS7960

Pada *driver* motor DC ini dapat mengeluarkan arus hingga 43A, dengan memiliki fungsi *Pulse Width Modulation* (PWM). Tegangan sumber DC yang dapat diberikan antara 5.5V-27V DC, sedangkan tegangan *input* level antara 3.3V-5V DC, *driver* motor ini menggunakan rangkaian *full H-bridge* dengan IC BTS7960 dengan perlindungan saat terjadi panas dan arus berlebihan. *Driver* BTS7960 43A dapat dilihat pada Gambar 2.2.



Gambar 2.2 BTS7960 Driver 43A H-Bridge Drive PWM

(Sumber: Infineon Technologies Datasheet)

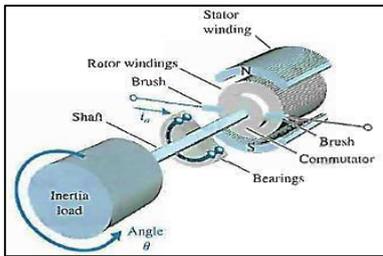
Pin konfigurasi dari penggunaan *driver* 43A *H-Bridge Drive* PWM ini dapat dilihat pada Gambar 2.3.

| | | |
|---|---|--|
| 1 | 2 | 1 = RPWM (Forward Level atau Sinyal PWM Input, aktif HIGH) |
| ■ | ○ | 2 = LPWM (Inversion Level atau Sinyal PWM Input, aktif HIGH) |
| ○ | ○ | 3 = R_EN (Forward Drive Enable input, Aktif HIGH) |
| ○ | ○ | 4 = L_EN (Reverse Drive Enable input, Aktif HIGH) |
| ○ | ○ | 5 = R_IS (Forward Drive, - side current alarm output) |
| ○ | ○ | 6 = L_IS (Reverse Drive, - side current alarm output) |
| 7 | 8 | 7 = VCC (+5v, connect ke arduino) |
| | | 8 = GND (Ground) |

Gambar 2.3 Pin Konfigurasi BTS7960 Driver 43A H-Bridge Drive PWM

2.3 Motor Direct Current (DC) 12V

Motor *direct current* (DC) memerlukan sumber tegangan yang searah pada kumparan medan untuk diubah menjadi energi mekanik. Bagian utama motor DC adalah stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Tegangan sumber DC dari *power supply* atau baterai menuju ke lilitan melalui sikat yang menyentuh komutator (dua segmen yang terhubung dengan dua ujung lilitan). Kumparan dalam satu lilitan disebut angker dinamo. Angker dinamo adalah sebutan untuk komponen yang berputar diantara medan magnet. Bagian bagian motor DC dapat dilihat pada Gambar 2.4.



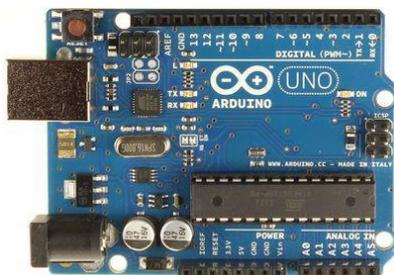
Gambar 2.4 Elemen-Elemen Dasar Motor DC

(Sumber: Dorf and Robert, 2008)

Prinsip kerja motor DC adalah jika sikat arang terhubung dengan satu sumber arus serah diluar dengan tegangan V , maka satu arus I masuk ke terminal kumparan rotor dan menghasilkan fluks. Dengan adanya fluks stator dan arus rotor akan menghasilkan satu gaya yang bekerja pada kumparan yang dikenal dengan gaya *lorentz*.

2.4 Arduino UNO R3

Arduino atau Genuino seperti yang terlihat pada Gambar 2.5 adalah papan mikrokontroler yang menggunakan IC ATmega328P. Arduino mempunyai 14 digital pin *input* atau *output* (yang mana 6 pin bisa digunakan sebagai PWM *output*), 6 analog *input*, 16 MHz *quartz crystal(clock)*, fasilitas *Universal Serial Bus (USB) connection*, sebuah *power jack (DC)*, sebuah *ICSP header* dan tombol *reset*. Arduino UNO R3 berisi segala hal yang dibutuhkan sebagai mikrokontroler, contoh menghubungkan *board* dengan computer dengan kabel USB atau mencatunya dengan *Alternating Current (AC)* ke *Direct Current (DC)* adapter atau baterai untuk menghidupkannya. Spesifikasi Arduino UNO R3 yang lebih detail terlihat pada Tabel 2.1



Gambar 2.5 Arduino UNO R3

Sumber: <http://ndoware.com/apa-itu-arduino-uno.html>

Tabel 2.1
Spesifikasi Arduino UNO R3

| | |
|-----------------------------|--|
| Microcontroller | ATmega328P |
| Operating Voltage | 5V |
| Input Voltage (recommended) | 7-12V |
| Input Voltage (limit) | 6-20V |
| Digital I/O Pins | 14 (of which 6 provide PWM output) |
| PWM Digital I/O Pins | 6 |
| Analog Input Pins | 6 |
| DC Current per I/O Pin | 20 mA |
| DC Current for 3.3V Pin | 50 mA |
| Flash Memory | 32 KB (ATmega328P) of which 0.5 KB used by bootloader |
| SRAM | 2 KB (ATmega328P) |
| EEPROM | 1 KB (ATmega328P) |
| Clock Speed | 16 MHz |
| Length | 68.6 mm |
| Width | 53.4 mm |
| Weight | 25 g |

Sumber: <https://www.arduino.cc/en/Main/arduinoBoardUno>

Sumber tegangan Arduino UNO R3 dapat diaktifkan melalui koneksi USB atau dengan daya eksternal. Eksternal (non-USB) daya dapat berasal baik dari adaptor maupun baterai. Adaptor ini dapat dihubungkan dengan menancapkan *plug jack* pusat positif ukuran 2.1 mm konektor *power*. Ujung kepala dari baterai dapat dimasukkan kedalam Gnd dan Vin pin *header* dari konektor *power*. Arduino dapat beroperasi dengan catu daya eksternal 6 V sampai 20 V. Namun jika menggunakan lebih dari 12 V, regulator tegangan bisa panas dan merusak papan. Kisaran yang disarankan adalah 7 V sampai 12 V.

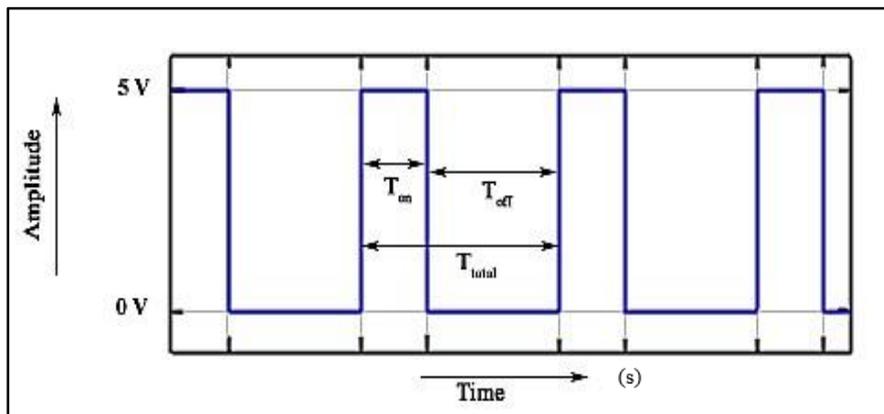
Arduino UNO R3 memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lainnya. Arduino UNO R3 menyediakan 4 UART TTL (5V) untuk komunikasi serial. Sebuah Arduino UNO R3 sebagai saluran komunikasi serial melalui USB dan sebagai port virtual com untuk perangkat lunak pada komputer. *Firmware* '8 U2 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 papan Arduino. LED RX dan TX di papan tulis 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 *Software Serial Library* memungkinkan untuk berkomunikasi secara serial pada salah satu pin digital pada *board* Arduino UNO R3. Arduino UNO R3 juga mendukung I2C (TWI) dan komunikasi *Serial Peripheral Interface* (SPI). Perangkat lunak Arduino termasuk perpustakaan kawat untuk menyederhanakan penggunaan bus I2C

2.5 Pulse Width Modulation

Pengaturan tegangan sumber biasanya menggunakan metode *Pulse Width Modulation* (PWM). Sinyal *Pulse Width Modulation* (PWM) adalah metode yang dapat digunakan untuk mengontrol kecepatan motor DC. Dimana kecepatan motor DC tergantung pada besarnya *duty cycle* yang diberikan pada motor DC tersebut.

Pada sinyal PWM, frekuensi sinyal konstan sedangkan *duty cycle* bervariasi dari 0%-100%. *Duty cycle* adalah besarnya sinyal kontrol yang diberikan pada motor. Persamaan untuk perhitungan *duty cycle* ditunjukkan pada Persamaan 2.1 dengan T_{on} adalah periode logika tinggi, dan T adalah periode keseluruhan. Sinyal PWM secara umum dapat dilihat dalam Gambar 2.6.



Gambar 2.6 Sinyal PWM secara umum

Sumber: (www.8051projects.net)

$$\text{Duty Cycle} = \frac{T_{on}}{T_{total}} \times 100\% \dots \dots \dots (2.1)$$

Timer atau *counter* yang digunakan pada PWM ini yaitu *timer* atau *counter* 0 (8 bit) dengan metode fast PWM dan *prescaler factor* (N) yaitu 256.

2.6 Kontroler

Sistem pengontrolan yang dirancang untuk melakukan dan menyelesaikan tugas tertentu. Syarat utama sistem pengontrolan adalah harus stabil. Di samping kestabilan mutlak, maka sistem harus memiliki kestabilan secara relatif, yakni tolak ukur kualitas kestabilan sistem dengan menganalisis sampai sejauh mana batas-batas kestabilan sistem tersebut jika dikena gangguan (Ogata K.,1997). Selain itu analisis juga di lakukan bagaimana kecepatan sistem dalam merespon *input* dan meredamnya terhadap adanya lonjakan (*over shoot*).

Suatu sistem dikatakan stabil, jika di beri gangguan maka sistem tersebut akan kembali ke dalam *steady state* yaitu *output* berada dalam keadaan tetap seperti tidak ada gangguan. Sistem di katakana tidak stabil, jika *output* beresilasi terus menerus ketika

dikenai suatu gangguan. Karena suatu sistem pengendalian biasanya melibatkan penyimpanan energi maka *output* sistem ketika diberi suatu *input*, tidak dapat mengikuti input secara serentak, tapi menunjukkan respon transien berupa suatu osilasi teredam sebelum mencapai *steady state*. Dalam sistem pengontrolan terdapat 2 macam loop.

1. Pengontrolan dengan loop terbuka

Sistem kontrol loop terbuka adalah sistem kontrol yang keluarannya tidak berpengaruh pada aksi pengontrolannya. Jadi pada sistem kontrol loop terbuka, keluaran tidak diukur atau diumpan balik untuk dibandingkan dengan masukan.

2. Pengontrolan dengan loop tertutup

Sistem kontrol loop tertutup adalah sistem kontrol yang keluarannya mempunyai pengaruh langsung pada aksi pengontrolannya. Disebut juga sistem yang menggunakan umpan balik untuk memperkecil kesalahan sistem.

2.6.1 Kontroler Proporsional

Kontroler proporsional memiliki keluaran yang sebanding dengan sinyal kesalahan (selisih antara besaran yang diinginkan dengan harga aktualnya). Untuk kontroler dengan aksi kontrol proporsional, hubungan antara keluaran kontroler $m(t)$ dan sinyal kesalahan penggerak $e(t)$ adalah:

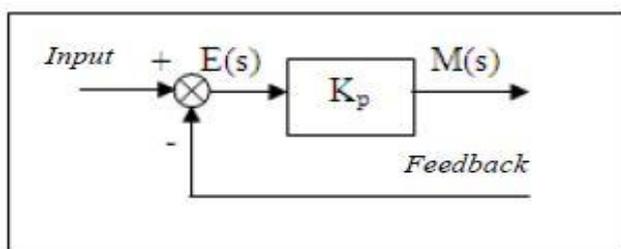
$$m(t) = K_p \cdot e(t) \dots\dots\dots (2.2)$$

atau, dalam besaran transformasi Laplace,

$$\frac{M(s)}{E(s)} = K_p \dots\dots\dots (2.3)$$

Dimana K_p adalah kepekaan proporsional atau penguatan.

Apapun wujud mekanisme yang sebenarnya dan apapun bentuk daya penggeraknya, kontroler proporsional pada dasarnya merupakan penguat dengan penguatan yang dapat diatur (Ogata K.,1997). Diagram blok kontroler proporsional ditunjukkan dalam Gambar 2.7



Gambar 2.7 Diagram Blok Kontroler Proporsional
Sumber: Ogata K., 1997

2.6.2 Kontroler Integral

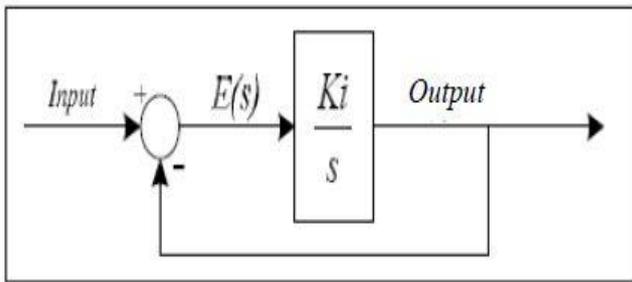
Kontroler integral berfungsi mengurangi kesalahan keadaan mantap pada kontroler proporsional sebelumnya. Pada kontroler dengan aksi integral, harga keluaran kontroler $m(t)$ diubah dengan laju yang sebanding dengan sinyal kesalahan penggerak $e(t)$.

Jadi,

$$\frac{dm(t)}{dt} = K_i \cdot e(t) \dots\dots\dots (2.4)$$

Dimana K_i adalah kepekaan integral

Jika harga $e(t)$ diduakalikan, maka harga $m(t)$ berubah dengan laju perubahan menjadi dua kali semula. Jika kesalahan penggerak nol, maka harga $m(t)$ tetap stasioner. Aksi kontrol integral seringkali disebut *control reset* (Ogata K.,1997). Diagram blok kontroler integral ditunjukkan dalam Gambar 2.8



Gambar 2.8 Diagram Blok Kontroler Integral
Sumber: Ogata K., 1997

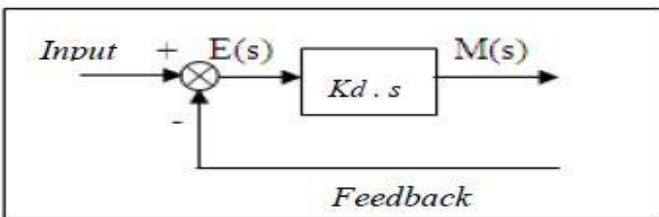
2.6.3 Kontroler Diferensial

Kontroler ini digunakan untuk memperbaiki atau mempercepat respons transien sebuah sistem kontrol dengan cara memperbesar *phase lead* terhadap penguatan kontrol dan mengurangi *phase lag* penguatan tersebut (Ogata K.,1997). Kontroler diferensial tidak dapat mengeluarkan *output* bila tidak ada perubahan *input*, selain itu kontroler differensial tidak dapat digunakan untuk proses yang mengandung *noise*. Hubungan antara keluaran kontroler $m(t)$ dan sinyal kesalahan penggerak $e(t)$ adalah:

$$\frac{M(s)}{E(s)} = K_d \cdot s \dots\dots\dots (2.5)$$

Dimana K_d adalah kepekaan diferensial

Diagram blok kontroler diferensial ditunjukkan dalam Gambar 2.9



Gambar 2.9 Diagram Blok Kontroler Diferensial
Sumber: Teknik Kontrol Automatik, Katsuhiko Ogata, 1997

2.6.4 Kontroler Proporsional Integral Diferensial (PID)

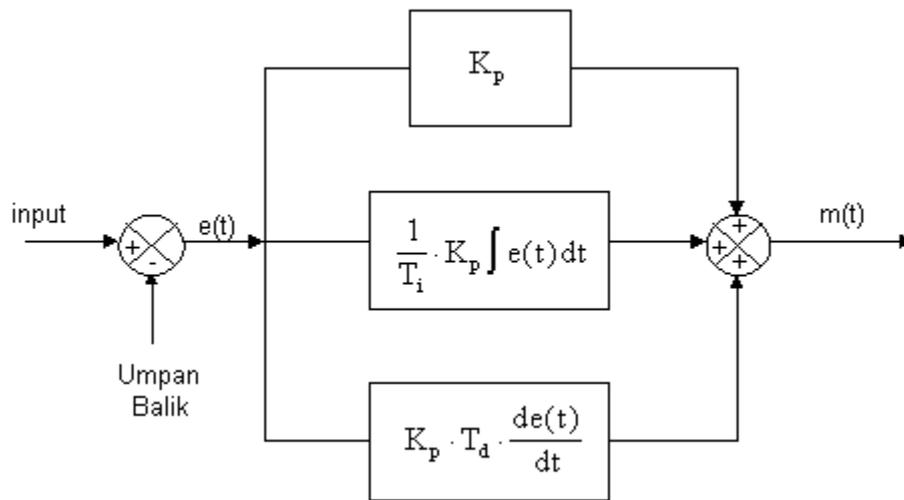
Gabungan aksi kontrol proporsional, integral, dan diferensial mempunyai keunggulan dapat saling menutupi kekurangan dan kelebihan dari masing-masing kontroler. Persamaan kontroler PID ini dapat dinyatakan sebagai berikut:

$$m(t) = Kp e(t) + \frac{Kp}{Ti} \int_0^t e(t) dt + Kp Td \frac{de(t)}{dt} \dots\dots\dots (2.6)$$

Dalam transformasi Laplace dinyatakan sebagai berikut:

$$\frac{M(s)}{E(s)} = Kp \left(1 + \frac{1}{Ti \cdot s} + Td \cdot s \right) \dots\dots\dots (2.7)$$

Diagram blok kontroler diferensial ditunjukkan dalam Gambar 2.10

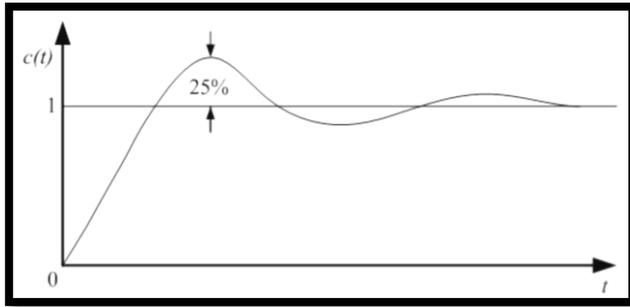


Gambar 2.10 Diagram Blok Kontroler PID
 Sumber: Ogata K., 1997

2.6.5 Metode Kontroler Proporsional Integral Diferensial (PID)

Ziegler dan Nichols mengemukakan aturan-aturan untuk menentukan nilai dari gain proporsional K_p , waktu integral T_i , dan waktu derivatif T_d berdasarkan karakteristik respon transien dari *plant* yang diberikan. Penentuan parameter kontroler PID atau penalaan kontroler PID tersebut dapat dilakukan dengan bereksperimen dengan plan (Ogata, K., 1997).

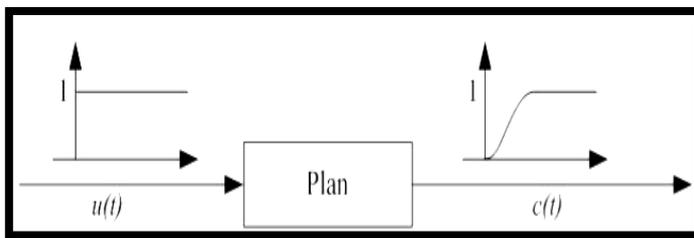
Terdapat dua metode yang disebut dengan aturan penalaan Ziegler-Nichols, pada kedua metode tersebut memiliki tujuan yang sama yaitu untuk mencapai 25% *maximum overshoot* pada respon unit step, ditunjukkan dengan Gambar 2.11



Gambar 2.11 Kurva Respon Unit Step yang Menunjukkan 25% *Maximum overshoot*
 Sumber: Ogata, K., 1997

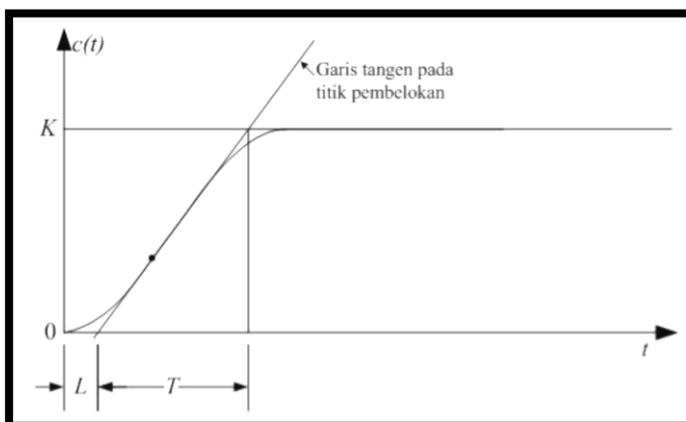
2.6.5.1 Metode Pertama

Metode pertama atau sering disebut metode kurva reaksi, respon dari plan dapat diperoleh secara eksperimental dengan masukan berupa unit step, seperti yang ditunjukkan dengan Gambar 2.12



Gambar 2.12 Respon Plant Terhadap Masukan Berupa Unit Step
 Sumber: Ogata, K. 1997

Jika dalam plan tersebut terdapat integrator atau *dominan complex-conjugate poles*, maka kurva respon unit step berbentuk seperti huruf S, seperti ditunjukkan dengan Gambar 2.13. jika respon tidak memberikan bentuk kurva S, maka metode ini tidak berlaku (Ogata, K., 1997).



Gambar 2.13 Kurva Respon yang Berbentuk S
 Sumber: Ogata, K. 1997

Kurva berbentuk S tersebut dapat dikarakteristikan menjadi dua konstanta yaitu waktu tunda L dan konstanta waktu T .

Waktu tunda dan konstanta waktu ditentukan dengan menggambar sebuah garis tangen pada titik pembelokan dari kurva S, dan menentukan perpotongan antara garis tangen dengan sumbu waktu t dan sumbu $c(t) = K$, seperti yang telah ditunjukkan dalam Persamaan sebagai berikut:

$$\frac{C(s)}{U(s)} = \frac{Ke^{-Ls}}{Ts+1} \dots\dots\dots (2.8)$$

Ziegler dan Nichols menyarankan untuk menentukan nilai-nilai dari K_p , T_i dan T_d berdasarkan pada formula yang ditunjukkan dalam Tabel 2.2 (Ogata, K., 1997)

Tabel 2.2
Aturan Penalaan Ziegler-Nichols Berdasarkan Respon Unit Step Dari Plan

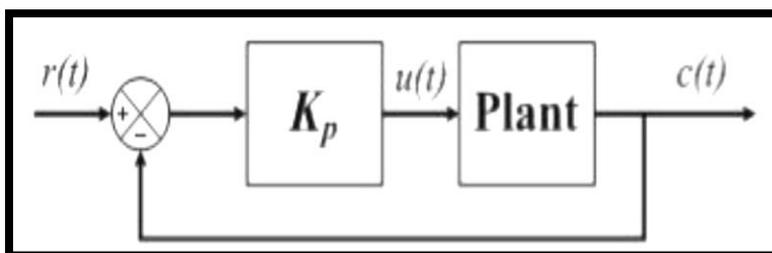
| Tipe Kontroler | K_p | T_i | T_d |
|----------------|------------------|-----------------|---------|
| P | $\frac{T}{L}m$ | ∞ | 0 |
| PI | $0,9\frac{T}{L}$ | $\frac{L}{0,3}$ | 0 |
| PID | $1,2\frac{T}{L}$ | $2L$ | $0,5 L$ |

Sumber: Ogata, K. 1997

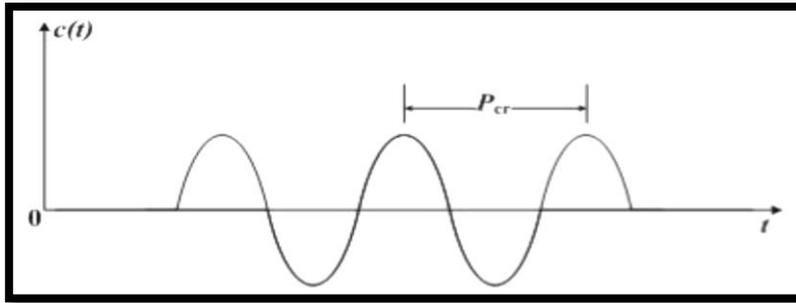
2.6.5.2 Metode Kedua

Dalam metode kedua ziegler-nichols, mula-mula yang dilakukan adalah membuat $T_i = \infty$ dan $T_d = 0$. Kemudian hanya dengan menggunakan tindakan kontrol proporsional, harga ditingkatkan dari nol ke suatu nilai kritis atau *critical gain* yang diibandingkan K_{cr} , disini mula-mula keluaran memiliki osilasi yang berkesinambungan (Jika keluaran tidak memiliki osilasi berkesinambungan untuk nilai K_p manapun yang telah diambil, maka metode ini tidak berlaku). Dari keluaran yang berosilasi secara berkesinambungan, penguatan kritis K_{cr} dan periode P_{cr} (Waktu yang dibutuhkan untuk melakukan satu gelombang) dapat ditentukan.

Diagram blok sistem loop tertutup dengan kontroler proporsional ditunjukkan dengan Gambar 2.14 dan untuk osilasi berkesinambungan dengan periode P_{cr} ditunjukkan dengan Gambar 2.15



Gambar 2.14 Sistem Loop Tertutup dengan Kontroler Proporsional
Sumber: Ogata, K., 1997



Gambar 2.15 Osilasi Berkesinambungan dengan Periode P_{cr}

Sumber: Ogata, K., 1997

Ziegler dan Nichols menyarankan penyetelan nilai parameter K_p, T_i, T_d dan berdasarkan rumus yang diperlihatkan dalam Tabel 2.3 (Ogata, K., 1997)

Tabel 2.3

Aturan Dasar Ziegler-Nichols Critical Gain K_{cr} Critical Period P_{cr}

| <u>Type Kontroler</u> | K_p | T_i | T_d |
|-----------------------|---------------|------------------------|----------------|
| P | $0.5 K_{cr}$ | ∞ | 0 |
| PI | $0.45 K_{cr}$ | $\frac{1}{1.2} P_{cr}$ | 0 |
| PID | $0.60 K_{cr}$ | $0.5 P_{cr}$ | $0.125 P_{cr}$ |

Sumber: Ogata, K., 1997

2.7 Sensor Jarak Ultrasonik (Sensor Ping)

Sensor jarak ultrasonik PING adalah sensor 40 KHz produksi parallax yang banyak digunakan untuk aplikasi atau kontes robot cerdas untuk mendeteksi jarak suatu objek terlihat pada Gambar 2.16.



Gambar 2.16 Sensor Jarak Ultrasonik PING

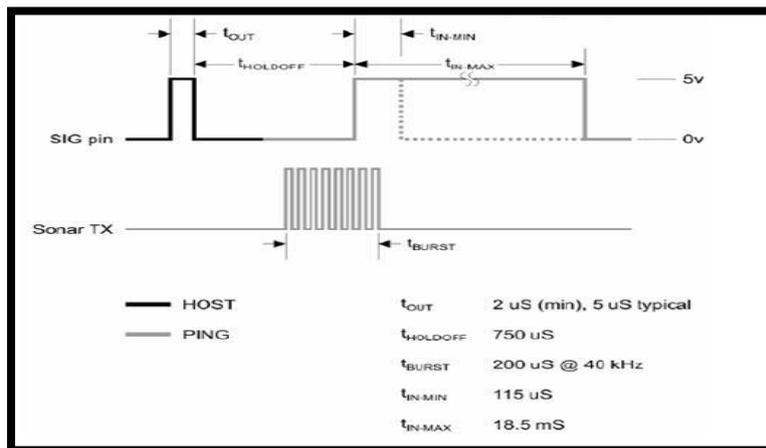
Sumber: DataSheet Sensor Jarak Ultrasonik PING

Sensor PING mendeteksi jarak objek dengan cara memancarkan gelombang ultrasonik (40 KHz) selama $t = 200 \text{ us}$ kemudian mendeteksi pantulannya. Sensor PING

memancarkan gelombang ultrasonik sesuai dengan kontrol dari mikrokontroler pengendali (pulsa *trigger* dengan tout min 2 us). Spesifikasi sensor ini:

- Kisaran pengukuran 3cm-3m.
- Input *trigger* –*positive* TTL pulsa, 2uS min., 5uS tipikal.
- Echo hold off* 750uS dari *fall of trigger pulse*.
- Delay before next measurement* 200uS.
- Burst indicator* LED menampilkan aktifitas sensor.

Diagram waktu sensor PING ditunjukkan dengan Gambar 2.17

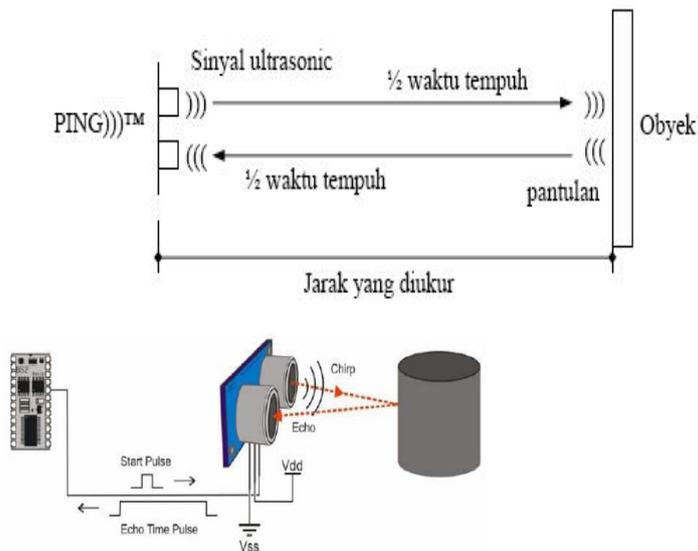


Gambar 2.17 Diagram Waktu Sensor PING

Sumber: DataSheet Sensor Jarak Ultrasonik PING

2.8.1 Prinsip Kerja Sensor Jarak Ultrasonik (Sensor Ping)

Pada dasarnya, Sensor PING terdiri dari sebuah chip pembangkit sinyal 40 KHz, sebuah speaker ultrasonik dan sebuah mikropon ultrasonik. Speaker ultrasonik mengubah sinyal 40 KHz menjadi suara sementara mikropon ultrasonik berfungsi untuk mendeteksi pantulan suaranya. Sensor PING mendeteksi jarak obyek dengan cara memancarkan gelombang ultrasonik (40 kHz) selama t_{BURST} (200 μ s) kemudian mendeteksi pantulannya. Sensor PING memancarkan gelombang ultrasonik sesuai dengan kontrol dari mikrokontroler pengendali (pulsa *trigger* dengan t_{OUT} min. 2 μ s). Seperti yang terlihat pada Gambar 2.18



Gambar 2.18 Prinsip Kerja Sensor PING

Sumber: DataSheet Sensor Jarak Ultrasonik PING

Gelombang ultrasonik ini melalui udara dengan kecepatan 344 meter per detik, mengenai obyek dan memantul kembali ke sensor. PING mengeluarkan pulsa *output high* pada pin SIG setelah memancarkan gelombang ultrasonik dan setelah gelombang pantulan terdeteksi PING akan membuat *output low* pada pin SIG. Lebar pulsa *High* (t_{IN}) akan sesuai dengan lama waktu tempuh gelombang ultrasonik untuk 2x jarak ukur dengan obyek. Maka jarak yang diukur adalah

$$S = (t_{IN} \times V) \div 2$$

Dimana :

S = Jarak antara sensor ultrasonik dengan objek yang dideteksi.

V = Cepat rambat gelombang ultrasonik di udara (344 m/s).

t_{IN} = Selisih waktu pemancaran dan penerimaan pantulan gelombang.

