

BAB II

DASAR TEORI

2.1 KAJIAN PUSTAKA

Pada penelitian ini, sistem perancangan untuk akuarium yang dapat mengontrol pH air bukan yang pertama dan juga menggunakan berbagai metode. Pada subbab ini menjelaskan penelitian sebelumnya yang berkaitan dengan penelitian yang akan digunakan sebagai referensi.

Penelitian berjudul "Rancang Bangun Sistem Pengendalian Ph Air Pada Tambak Ikan Bandeng Menggunakan Kontroller PID Berbasis Labview" dilakukan oleh Agung Setya W dan rekannya pada tahun 2020. Studi ini dilakukan karena banyak pengusaha mengalami kerugian karena gagal panen, salah satunya disebabkan oleh perubahan kualitas air. Metode PID (Proporsional Integral Derivatif) berbasis Labview digunakan untuk mengendalikan pengkodisian pH air secara otomatis. Penelitian ini menggunakan sensor pH SKU:SEN0161, driver motor L298N, pompa DC 12V, LCD 16x2, dan Arduino Atmega 2560. Selain itu, cairan asam dan basa digunakan. Pada pengujian ini, nilai sensor rata-rata sebesar 7,77, nilai pH meter sebesar 7,8, dan nilai *error* sebesar 0,03. Alat yang digunakan dapat memperbaiki hasil respon dinamis dari nilai pH sehingga dapat sesuai dengan setpoint yang ditentukan. Pada penelitian ini didapatkan nilai K_p sebesar 18,5, nilai K_i sebesar 1,78 dan nilai K_d sebesar 99,34. Respon sistem yang didapatkan pada penelitian yaitu nilai *error steady state* sebesar 0,1%, nilai t_d sebesar 9,889 detik, nilai t_s sebesar 42,804 detik, nilai t_r sebesar 31,35 detik dan nilai *maximum overshoot* sebesar 1,18% [13].

Penelitian berjudul "Pengembangan Sistem Otomatisasi Berbasis IoT (*Internet of Things*) untuk Pemantauan Suhu dan pH pada Aquascape Menggunakan Nodemcu ESP8266 dalam Aplikasi Telegram" dilakukan oleh Ramdani D dan rekan-rekannya pada tahun 2020. Studi ini membahas perancangan alat otomatisasi untuk memantau suhu dan pH pada Aquascape. Alat ini menggunakan sensor DS18B20 yang mendeteksi suhu dari 25°C hingga 28°C dan modul deteksi pH yang mendeteksi pH dari 6,0 hingga 8,0. Selain itu, penjadwalan

pencahayaannya dan kipas pendingin secara real-time dapat direncanakan melalui NTP *Client Server*. Mikrokontroler Nodemcu ESP8266, yang sudah dilengkapi dengan fitur Wi-Fi, terhubung ke sensor-sensor untuk mengatasi gangguan fotosintesis dan memastikan keberlanjutan ekosistem Aquascape. Ketika sensor mendeteksi suhu di bawah 25°C, lampu Aquascape akan secara otomatis menyala. Jika suhu lebih tinggi dari 28°C, kipas pendingin akan aktif. Untuk sensor pH, Nodemcu Esp8266 akan mengirimkan pemberitahuan langsung ke bot Telegram jika pH terdeteksi di bawah 6 atau di atas 8. Pengujian sensor DS18B20 dengan termometer digital menunjukkan kesalahan rata-rata 2,15%, sedangkan pengujian dengan pH meter menunjukkan kesalahan rata-rata 0,216%. Untuk perencanaan penerangan Aquascape selama 8 jam per hari, server NTP dapat memberikan informasi secara berkala [14].

Penelitian yang berjudul "Perancangan Sistem Otomatisasi pada Aquascape Berbasis Mikrokontroler Arduino Nano" dilakukan oleh Triawan Y dan rekannya pada tahun 2020. Penelitian tersebut membahas bagaimana membuat sistem otomatisasi untuk Aquascape yang memiliki fitur seperti memberikan makanan kepada ikan, mengatur suhu, ketinggian air, dan pergantian air secara berkala. Arduino Nano berfungsi sebagai mikrokontroler dalam sistem ini, RTC berfungsi sebagai modul timer perencanaan, sensor DS18B20 mendeteksi suhu, dan sensor HC-SR04 mengukur ketinggian air dalam Aquascape. Hasil dari sistem ini adalah pemanas akan menyala ketika sensor suhu mendeteksi suhu di bawah 22°C dan pendingin akan menyala ketika suhu melebihi 27°C. Pengujian sensor DS18B20 dengan alat pengukur suhu menunjukkan rata-rata kesalahan sebesar 1,26%, dan pengujian sensor HC-SR04 dengan mistar menunjukkan rata-rata kesalahan sebesar 2,27%, dan modul RTC dapat merencanakan pemberian makanan ikan yang dikendalikan oleh motor servo, lamanya pencahayaan, dan jadwal pergantian air [15].

Penelitian yang dilakukan oleh Reynaldi Mahendra Putra dan rekannya pada tahun 2020 berjudul "Kontrol dan Pengawasan pH Air Pada Budidaya Lobster Air Tawar Dengan Metode PID Berbasis *Internet Of Things*" membahas bagaimana secara otomatis mengontrol kadar keasaman (pH) air kolam lobster, serta bagaimana menggunakan metode kontrol PID melalui aplikasi Blynk untuk

memantau kadar pH air secara realtime. Sistem ini menggunakan ESP32 sebagai mikrokontroler, sensor pH, dosing pompa peristaltik, dan LCD untuk menampilkan data. Dengan parameter K_p 0,76, K_i 0,001, dan K_d 92,72, perancangan ini menghasilkan grafik respons sistem yang menunjukkan bahwa motor dengan cairan asam membutuhkan waktu sekitar 325 detik untuk mencapai set point. Waktu yang diperlukan untuk motor dengan cairan basa adalah sekitar 276 detik. Nilai *error* rata-rata 3,38% dimiliki oleh sensor pH analog DFROBOT SEN0161-V2 [16].

Penelitian yang dilakukan oleh Aztisyah D, et al. pada tahun 2022 dengan judul "Implementasi Logika *Fuzzy Mamdani* Pada pH Air dalam Sistem Otomatisasi Suhu dan pH Air Aquascape Ikan Guppy" berhasil mencapai tujuan untuk menggunakan logika *fuzzy mamdani* untuk mengontrol suhu dan pH air di Aquascape ikan guppy. Penelitian ini menghasilkan rancangan sistem dengan menggunakan sensor suhu DS18B20 dan sensor pH DFRobot SKU : SEN0161. Sistem dapat menstabilkan suhu antara 27 dan 30 derajat *Celcius* dan pH antara 6,5 dan 7,2 derajat *Celcius*. Sistem dapat melakukan pembacaan suhu menggunakan termometer raksa manual dengan persentase *error* 1.4136%, pembacaan pH menggunakan pH meter dengan persentase *error* 0.6813%, dan pembacaan nilai *fuzzy* menggunakan matlab dengan persentase *error* 0.7246% [17].

Tabel 2. 1 Kajian penelitian sebelumnya

Tahun	Penulis	Metode Yang Diusulkan	Hasil Pengujian
2020	Agung Setya Wicaksana	PID	<ol style="list-style-type: none"> 1. Rata-rata sensor pH 7,77 2. Rata-rata pH meter 7,8 3. Nilai <i>error</i> pH 0,03 4. $K_p = 18,5$, $K_i = 1,78$, $K_d = 99,34$ 5. $Ess = 0,1\%$, $td = 9,889S$, $ts = 42,804S$, $tr = 31,35S$, $Overshoot = 1,18\%$
2020	Dendi Ramdani, dkk.	-	<ol style="list-style-type: none"> 1. Rata-rata <i>error</i> sensor DS18B20 2,105% 2. Rata-rata <i>error</i> sensor pH 0,216% 3. NTP server membuat jadwal lampu <i>Aquascape</i>
2020	Yessi Triawan, dkk.	-	<ol style="list-style-type: none"> 1. Rata-rata <i>error</i> sensor DS18B20 dengan <i>thermo gun</i> 1,26% 2. Rata-rata <i>error</i> sensor HC-SR04 dengan mistar 2,27% 3. RTC membuat jadwal pakan ikan
2020	Reynaldi Mahendra Putra, dkk.	PID	<ol style="list-style-type: none"> 1. waktu mencapai set point cairan asam 325 detik dan basa 276 detik. 2. Rata-rata <i>error</i> sensor pH 3,38%

2022	Diaz Aztisyah, dkk.	<i>Fuzzy</i> Mamdani	1. Suhu yang didapat 27 – 30 °C 2. Rata-rata <i>error</i> sensor pH 0.6813%.
------	------------------------	-------------------------	---

Berdasarkan penelitian yang sudah pernah dilakukan dan digunakan sebagai kajian pustaka, peneliti akan melakukan penelitian dengan mengendalikan pH air aquarium dengan mengadaptasi metode PID dari penelitian referensi [13] yang dibuktikan dengan hasil data yang diperoleh dalam melakukan pengendalian pH air terdapat analisa tanggapan waktu dibandingkan referensi [16]. Sedangkan perancangan alat, peneliti mengadaptasi dari penelitian referensi [17] yang dimana menggunakan arduino sebagai pusat pengendalian sistem, sensor DFRobot SKU : SEN0161 sebagai pengukur pH air aquarium.

2.2 DASAR TEORI

Pada subbab ini, dasar teori yang terkait dengan penelitian ini digunakan sebagai pengetahuan untuk penelitian ini. Beberapa dasar teori ini termasuk Ikan *Oscar*, akuarium, metode PID, metode tuning *cohen coon*, arduino nano, sensor pH, driver L298N, pompa DC 12V, LCD 16x2 I2C, dan catu daya 12V.

2.2.1 Ikan *Oscar* (*Astronotus ocellatus*)

Ikan *Oscar* (*Astronotus ocellatus*) adalah ikan air tawar yang populer di dunia akuatik hobi. Ikan ini terkenal karena ukurannya yang besar, warna cerah, serta perilaku dan kepribadiannya yang unik. Ikan *Oscar* dapat bertahan dengan rentang pH 6,5 hingga 8. Ikan *Oscar* berasal dari Sungai Amazon dan sungai-sungai terkait di Amerika Selatan [18].

Ciri-ciri Ikan *Oscar*:

1. **Ukuran:** Ikan *Oscar* dapat tumbuh hingga ukuran yang cukup besar, biasanya sekitar 25 hingga 35 cm, tetapi dalam kondisi ideal dan tangki yang cukup besar, mereka bisa lebih besar lagi.
2. **Warna:** Warna dasar ikan *Oscar* adalah abu-abu atau coklat dengan bercak-bercak hitam di seluruh tubuhnya. Seleksi keturunan telah menghasilkan variasi warna yang luas, termasuk merah, kuning, albino, dan kombinasi warna lainnya.
3. **Perilaku:** Ikan *Oscar* dikenal memiliki kepribadian yang kuat. Mereka memiliki interaksi sosial yang menarik dan bisa mengenali pemiliknya. Ikan ini bisa menjadi cukup interaktif dan bisa dijinakkan [18].



Gambar 2. 1 Ikan *Oscar* [18]

Tabel 2. 2 Nama Variasi Berdasarkan Penampilan dan Karakteristik Ikan *Oscar*

Nama	Penjelasan
<i>Astronotus ocellatus</i>	Nama ilmiah dari ikan <i>Oscar</i> .
<i>Oscar Fish</i>	Nama umum yang paling umum digunakan.
<i>Tiger Oscar</i>	Merujuk pada variasi warna dengan pola garis-garis yang mirip dengan motif harimau.
<i>Albino Oscar</i>	Variasi yang memiliki warna putih karena kurangnya pigmen.
<i>Red Oscar</i>	Variasi dengan warna merah yang cerah.
<i>Lutino Oscar</i>	Variasi dengan warna kuning cerah.
<i>Wild Oscar</i>	Merujuk pada ikan <i>Oscar</i> dengan penampilan alami seperti di habitat alaminya.
<i>Marble Oscar</i>	Merujuk pada variasi warna dengan pola bercak-bercak yang khas seperti marmer.
<i>Black Oscar</i>	Variasi dengan warna hitam bisa berubah seiring pertumbuhan dan perubahan lingkungan

2.2.2 *Oscar Tiger*

Dinamakan *oscar tiger* karena coraknya seperti harimau. Warnanya hitam dan coraknya berwarna kuning kemerahan. *Oscar tiger* sangatlah agresif akan tetapi jenis ikan ini yang paling populer di dunia. Karena pertumbuhannya yang sangat cepat bisa mencapai 20 hingga 30cm lebih baik untuk tidak menggabungkan ikan hias lainnya dengan ikan *Oscar* [19].



Gambar 2. 2 *Oscar Tiger* [19]

Gambar 2.2 merupakan ikan *Oscar tiger*. Penting untuk diingat bahwa warna tubuh dan jenis ikan *Oscar* dapat memiliki karakteristik fisik dan perilaku yang sedikit berbeda, meskipun mereka masih berasal dari spesies yang sama [19].

2.2.3 Akuarium

Akuarium dapat digunakan untuk budidaya ikan air tawar dan air laut, dan juga dapat menjadi penghias ruangan. Akuarium juga sangat mudah dalam perawatannya [20]. Dapat dilihat pada Gambar 2.1 merupakan akuarium pada ikan hias air tawar.

Jenis-Jenis Akuarium:

1. Akuarium Air Tawar: Didesain untuk menampung ikan dan organisme air tawar seperti ikan hias, kura-kura, amfibi, dan tanaman air tawar.
2. Akuarium Air Laut: Merupakan lingkungan air asin yang memelihara ikan dan organisme laut, seperti karang, ikan karang, dan invertebrata laut.
3. Akuarium Paludarium: Kombinasi dari akuarium air dan terrarium darat, menciptakan lingkungan untuk ikan dan hewan yang membutuhkan akses ke daratan.
4. Akuarium Rawa: Meniru lingkungan rawa-rawa alami dengan air dangkal dan tumbuhan rawa [20].



Gambar 2. 3 Akuarium Ikan *Oscar* [20]

Gambar 2.3 merupakan akuarium dengan ikan hias *Oscar*. Masalah pH dalam akuarium yang di isi ikan *Oscar* bisa memiliki dampak signifikan pada kesehatan dan kenyamanan ikan serta keseluruhan stabilitas lingkungan akuarium [21]. Berikut adalah beberapa masalah pH yang sering ditemukan pada akuarium ikan *Oscar*:

1. pH Terlalu Tinggi (Alkalinitas Tinggi): Pengaruh pada Organisme: Jika pH terlalu tinggi, ikan *Oscar* dan organisme akuatik lainnya bisa mengalami stres, karena sebagian besar spesies air tawar lebih nyaman dalam lingkungan yang

sedikit asam hingga netral. Selanjutnya pada Kelarutan Nutrisi: pH yang tinggi dapat mengurangi kelarutan beberapa nutrisi penting dalam air, sehingga tanaman air mungkin mengalami defisiensi nutrisi.

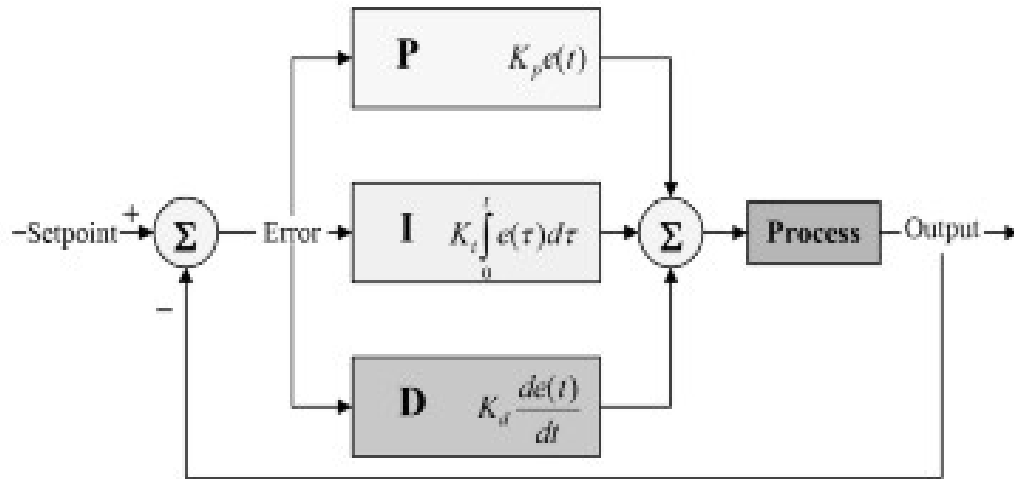
2. pH Terlalu Rendah (Asamitas Tinggi): Tingkat Toksisitas: pH yang terlalu rendah dapat menyebabkan peningkatan toksisitas amonia, yang dapat merusak kesehatan ikan dan organisme akuatik lainnya. Selanjutnya pada Masalah Pencernaan: pH yang rendah juga dapat mempengaruhi pencernaan ikan dan mengganggu kesehatan mereka secara keseluruhan.
3. Instabilitas pH: Tidak Konsisten: Fluktuasi pH yang tidak konsisten dapat membuat ikan *Oscar* kesulitan beradaptasi dan menyesuaikan diri dengan lingkungan. Ini bisa mempengaruhi kesehatan dan perilaku mereka.
4. Perubahan pH yang Tiba-tiba: Perubahan Stres: Perubahan pH yang cepat dan drastis dapat menyebabkan stres pada ikan *Oscar*, yang dapat memicu masalah kesehatan lebih lanjut.
5. Kekurangan *Buffer* Alami: Penurunan pH Mendalam: Jika kekurangan buffer alami di dalam air, penurunan pH yang tajam dapat terjadi dengan cepat, menyebabkan stres pada ikan.
6. Kontaminasi Air: Zat Kimia Berlebihan: Pemakaian bahan kimia atau bahan-bahan lain yang dapat merubah pH (seperti penggunaan batu kapur) secara berlebihan bisa menyebabkan masalah pH yang tidak diinginkan[21].

2.2.4 PID (*Proportional Integral Derivatif*) Controller

Pengontrol PID (*Proportional Integral Derivatif*) berfungsi untuk menentukan presisi sistem instrumentasi berdasarkan karakteristiknya dan adanya umpan balik (*feedback*) pada sistem yang bersangkutan. Sistem kontrol PID memiliki tiga metode pengaturan. Pengontrol proporsional P menunjukkan sinyal deviasi sistem kontrol proporsional. Ketika ada sinyal kesalahan, kontroler PID akan menghasilkan efek kontrol untuk mengurangi kesalahan. Dalam perancangan sistem pengontrol PID, parameter P, I, dan D diatur untuk tanggapan sinyal. I (integral) menghilangkan kesalahan statis dan meningkatkan stabilitas sistem, dan D (derivatif) menunjukkan perubahan penyimpangan sinyal dan mempercepat respons sistem. Kontrol proporsional dapat mempercepat naik waktu, kontrol

integral dapat mengurangi kesalahan (*error*), dan kontrol derivatif dapat mengurangi *overshoot* atau *undershoot* [22].

Diagram blok dari kontroler PID menunjukkan bagaimana sinyal kontrol dihasilkan dan digunakan untuk mengendalikan sistem instrumentasi. Pada Gambar 2.10 merupakan diagram blok pengendali PID, yaitu:



Gambar 2. 4 Blok diagram pengendali PID [22]

Dari Gambar 2.4 diatas didapatkan fungsi alih kontroler PID dapat dilihat pada persamaan 2.1 dibawah ini.

$$u(t) = K_p e(t) + K_i \int_0^t e(x) dx + K_d \frac{de(t)}{dt} \quad (2.1)$$

Keterangan:

$u(t)$ = 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/T_d)

$e(t)$ = sinyal *error* = referensi – keluaran plant = *set point*

Pada persamaan 2.1 menjelaskan bahwa nilai keluaran $G_s(s)$ merupakan jumlah dari kontrol proporsional (K_p), kontrol integral (K_i), dan kontrol derivatif (K_d), yang masing-masing dipengaruhi oleh *error* (e) dan waktu (t) tertentu [22]. Setiap konstanta pengendali PID memiliki karakteristik masing-masing, dengan karakteristik sebagai berikut:

Tabel 2. 3 Karakteristik Pengendali PID [23]

Parameter	<i>Rise Time</i>	<i>Overshoot</i>	<i>Settling Time</i>	<i>Error steady state</i>
Proporsional	Menurunkan	Meningkatkan	Perubahan Kecil	Menurunkan / mengurangi
Integral	Menurunkan	Meningkatkan	Meningkatkan	Mengeliminasi
Derivatif	Perubahan Kecil	Menurunkan	Menurunkan	Perubahan Kecil

Berdasarkan tabel 2.3 dapat disimpulkan bahwa, kontrol proporsional berfungsi untuk mempercepat *rise time*, kontrol integral berfungsi untuk menghilangkan *error steady state*, kontrol derivatif berfungsi untuk meningkatkan stabilitas sistem dan mengurangi *overshoot* [23].

2.2.4.1 Kontrol Proporsional

Parameter pengendali proporsional (P) merupakan bagian dari sistem kontroler umpan balik yang memiliki keluaran yang sebanding dengan besarnya sinyal *error* yang terjadi pada sistem. Pada kontrol proporsional sinyal kesalahan $e(t)$ merupakan masukan kontrol, sedangkan keluaran kontrol adalah sinyal $u(t)$ [24]. Hubungan antara masukan $u(t)$ dan keluaran $u(t)$ adalah $u(t)$:

$$u(t) = K_p \cdot e(t) \quad (2.2)$$

Keterangan:

$u(t)$ = sinyal output pengendali PID

K_p = Konstanta Proporsional

$e(t)$ = sinyal *error* = referensi – keluaran plant = *setpoint*

Pada kontrol proporsional dapat mempercepat *rise time* dan mengurangi *error* suatu *plant*. Kontrol proporsional mempengaruhi kecepatan respon kontroler untuk mencapai nilai *setpoint*. Semakin besar nilai K_p , maka sistem akan merespon lebih cepat, namun jika nilai K_p terlalu besar, sistem dapat mengalami *overshoot*, yaitu respon yang berlebihan dan melebihi nilai *setpoint*.

2.2.4.2 Kontrol Integral

Pada pengontrol integral digunakan untuk memperbaiki respon sistem dan memastikan keluaran sistem mencapai nilai yang diinginkan. Pengontrol integral menggunakan integral untuk meningkatkan respon sistem secara kontinyu dengan batasan tertentu. Dengan menggunakan pengontrol integral dapat meminimalkan

nilai *error steady state* dan performa sistem dapat ditingkatkan [24]. Nilai masukan $u(t)$ diubah pada kontrol proporsional dari sinyal kesalahan $e(t)$ sehingga hubungan antara masukan kontrol $e(t)$ dan keluaran $u(t)$ yaitu:

$$u(t) = K_i \int_0^t e(t) dt \quad (2.3)$$

Keterangan:

$u(t)$ = sinyal output pengendali PID

K_i = Konstanta Integral

$e(t)$ = sinyal *error* = referensi – keluaran plant = *setpoint*

dt = Nilai perubahan waktu (detik)

2.2.4.3 Kontrol Derivatif

Pada pengontrol derivative memiliki sinyal keluaran yang sebanding dengan perubahan *error*, semakin cepat *error* berubah maka semakin besar sistem kontrol yang dihasilkan. sistem dapat menjadi lebih responsif terhadap perubahan *error*, yang pada akhirnya dapat meningkatkan kestabilan sistem. Selain itu, parameter derivatif juga dapat memberikan efek redaman pada sistem yang berosilasi, sehingga memungkinkan nilai K_p yang lebih besar untuk diberikan. Untuk mencapai nilai parameter PID (K_p , K_i , dan K_d) yang optimal, berbagai metode *tuning* PID telah diusulkan [24]. Hubungan antara masukan kontrol $e(t)$ dengan keluaran kontrol $u(t)$ yaitu:

$$u(t) = K_d \frac{de(t)}{dt} \quad (2.4)$$

Keterangan:

$u(t)$ = sinyal output pengendali PID

K_d = Konstanta Derivatif

$e(t)$ = sinyal *error* = referensi – keluaran plant = *setpoint*

dt = Nilai perubahan waktu (detik)

2.2.5 Metode Tuning Cohen Coon

Terdapat beberapa metode untuk menentukan nilai yang akan digunakan untuk proporsional, integral dan turunan dari pengontrol parameter PID. Salah satu metode tersebut adalah metode tuning *Cohen coon*. Respons sistem dimodelkan ke perubahan langkah sebagai respons orde pertama ditambah waktu mati,

menggunakan metode *Cohen-Coon* (Gambar 2.9). K adalah keadaan tunak keluaran dibagi dengan perubahan langkah masukan, m adalah konstanta waktu efektif dari respons orde pertama, dan d adalah waktu mati [25].

Tabel 2. 4 Parameter PID Menggunakan Metode *Cohen-Coon*

Tipe Pengontrol	K_p	T_i	T_D
P	$K_p = \frac{1}{K} \left(\frac{T}{L} \right) \left[1 + \frac{1}{4} \left(\frac{L}{T} \right) \right]$	-	-
PI	$K_p = \frac{1}{K} \left(\frac{T}{L} \right) \left[0,9 + \frac{1}{4} \left(\frac{L}{T} \right) \right]$	$T_i = L \left[\frac{30 + 3 \left(\frac{L}{T} \right)}{9 + 20 \left(\frac{L}{T} \right)} \right]$	-
PD	$K_p = \frac{1}{K} \left(\frac{T}{L} \right) \left[\frac{5}{4} + \frac{1}{6} \left(\frac{L}{T} \right) \right]$	-	$T_D = L \left[\frac{6 - 2 \left(\frac{L}{T} \right)}{22 + 3 \left(\frac{L}{T} \right)} \right]$
PID	$K_p = \frac{1}{K} \left(\frac{T}{L} \right) \left[\frac{4}{3} + \frac{1}{4} \left(\frac{L}{T} \right) \right]$	$T_i = L \left[\frac{32 + 6 \left(\frac{L}{T} \right)}{13 + 8 \left(\frac{L}{T} \right)} \right]$	$T_D = L \left[\frac{4}{11 + 2 \left(\frac{L}{T} \right)} \right]$

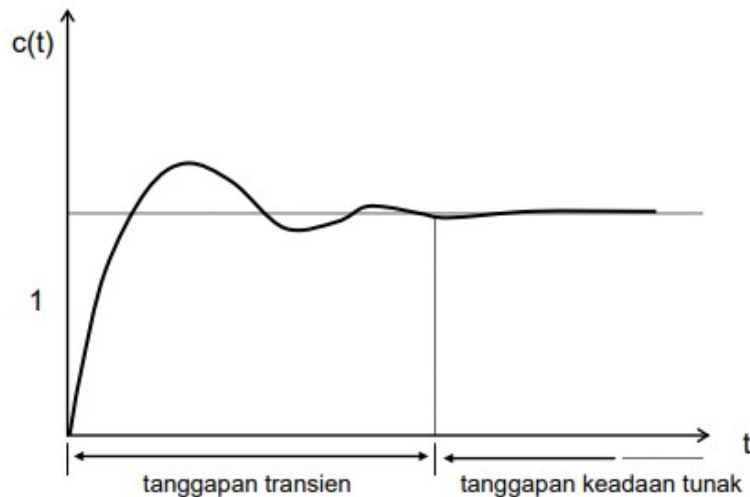
2.2.6 Analisa Tanggapan Waktu (Respon Transien)

Tahapan pertama dalam menganalisa suatu sistem kontrol adalah dengan menurunkan model matematik dari sistem tersebut. Sekali suatu model telah ditentukan, maka berbagai metode dapat diterapkan untuk menganalisa kinerja sistem tersebut. Salah satu cara untuk menguji dan menganalisa suatu sistem adalah dengan memberikan suatu sinyal uji (*test signal*) sebagai masukan dan mengamati serta menganalisa keluarannya. Berbagai sinyal masukan dapat digunakan untuk keperluan analisa yang berbeda-beda. Jika sistem yang digunakan untuk keperluan masukan dengan kenaikan gradual sepanjang waktu, maka digunakan sinyal uji fungsi ramp. Keluaran yang dihasilkan merupakan tanggapan (*response*) dari sistem yang diberikan sinyal uji. Bila analisa yang dilakukan merupakan analisa dalam lingkup waktu dan masukan yang diberikan bukan merupakan fungsi periodik (mempunyai frekuensi), maka analisa tersebut merupakan analisa tanggapan waktu (*time response*).

2.2.6.1 Tanggapan Transien dan Tanggapan Keadaan Tunak

Tanggapan waktu dari suatu sistem kontrol dibagi menjadi dua bagian, yaitu tanggapan transien (*transient response*) dan tanggapan keadaan tunak (*steadystate response*). Tanggapan transien berlangsung dari saat mulai hingga tanggapan sistem mencapai nilai akhir yang diinginkan (*final state*). Tanggapan keadaan tunak

dimulai pada saat tanggapan mulai pertama kali mendekati nilai akhir hingga waktu yang tak terhingga [26].



Gambar 2. 5 Tanggapan transien dan tanggapan keadaan tunak [26]

Tanggapan transien digunakan untuk menganalisa sifat naik atau permulaan dari suatu sistem bila diberikan sinyal uji. Sedangkan tanggapan keadaan tunak digunakan untuk menganalisa karakteristik sistem pada saat mencapai nilai akhirnya [26]. Spesifikasi tanggapan transien dalam domain waktu yang dimaksud adalah:

1. Waktu Tunda (*Delay Time*) : waktu yang dibutuhkan tanggapan untuk mencapai setengah dari nilai akhir dari tanggapan untuk pertama kali
2. Waktu Naik (*Rise Time*) : waktu yang dibutuhkan untuk naik dari 10% – 90%, 5% – 95%, atau 0% – 100% dari nilai akhir dari tanggapan. Untuk kasus *underdamped*, biasanya digunakan kriteria 0% – 100%. Untuk kasus *overdamped*, biasanya digunakan kriteria 10% – 90%. Dapat dilihat sampai dengan respon memotong sumbu *setpoint* yang pertama.
3. Waktu *Settling* (*Settling Time*) : waktu yang dibutuhkan tanggapan untuk mencapai nilai akhir dari tanggapan dan tetap berada pada nilai tersebut dalam range persentase tertentu dari nilai akhir (biasanya 5% atau 2%).
4. *Overshoot* : nilai puncak maksimum dari tanggapan diukur dari nilai akhir dari tanggapan. Cara pengujian dengan melihat respon yang melewati *setpoint*
5. Kesalahan Keadaan Tunak (*Error steady state*) : kesalahan yang terjadi pada saat keluaran dari sistem mencapai harga akhir. Kesalahan ini diperbandingkan

terhadap masukan dari sistemnya. Bila keadaan akhir dari keluaran tidak sama dengan masukannya, berarti terdapat kesalahan keadaan tunak. Cara pengujian dengan cara mengukur nilai persen *error* posisi pada keadaan *steady state* atau keadaan menuji stabil [26].

2.2.7 Arduino IDE

Arduino IDE (*Integrated Development Environment*) adalah perangkat lunak yang digunakan untuk mengembangkan dan memprogram papan pengembangan elektronik yang disebut Arduino. Arduino adalah platform open-source yang memungkinkan pengembang, hobiis, dan mahasiswa untuk membuat berbagai proyek elektronik interaktif dengan mudah. Arduino IDE membantu dalam menulis, mengedit, mengunggah, dan mengelola kode program untuk papan Arduino [27].



Gambar 2. 6 Software Arduino IDE [27]

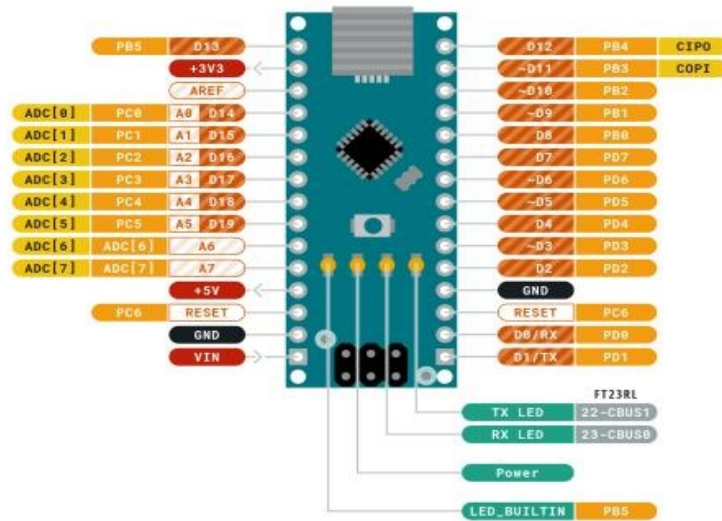
Tabel 2. 5 Komponen utama yang terdapat pada arduino ide

<i>Text Editor</i>	Tempat menulis kode program. Arduino IDE mendukung bahasa pemrograman berbasis <i>Wiring</i> , yang mirip dengan bahasa C/C++. Dapat membuat fungsi, pernyataan, dan variabel seperti dalam bahasa pemrograman lain.
<i>Toolbar</i>	<i>Toolbar</i> menyediakan tombol-tombol pintas untuk tugas umum seperti mengunggah kode ke papan Arduino, membuka atau menyimpan proyek, dan memilih jenis papan Arduino yang ingin digunakan.

<i>Serial Monitor</i>	Untuk berkomunikasi dengan papan Arduino melalui koneksi serial. Dapat mengirim dan menerima data antara komputer dan papan Arduino melalui <i>port</i> serial.
<i>Compiler dan Uploader</i>	Arduino IDE menyediakan kompiler yang mengonversi kode ke dalam bahasa mesin yang dapat dimengerti oleh papan Arduino. Setelah berhasil dikompilasi, dapat menggunakan fitur <i>uploader</i> untuk mengunggah kode ke papan Arduino.
<i>Library Manager</i>	Arduino memiliki berbagai perpustakaan (<i>libraries</i>) yang memuat fungsi-fungsi dan kode yang sudah ada. <i>Library Manager</i> memudahkan pengguna untuk mencari, mengunduh, menginstal, dan mengelola perpustakaan ini.
<i>Examples</i>	Arduino IDE menyediakan contoh kode untuk berbagai sensor, perangkat, dan modul yang sering digunakan dalam proyek elektronik. Dapat melihat contoh-contoh ini untuk memahami cara kerja dan penggunaan berbagai komponen.
<i>Board Manager</i>	Bagian ini memungkinkan untuk menambahkan dukungan untuk jenis papan Arduino yang berbeda. Dengan memilih papan yang sesuai, dapat memastikan bahwa kode cocok dengan spesifikasi perangkat keras papan yang digunakan.

2.2.8 Arduino Nano

Arduino Nano merupakan *board* mikrokontroler berukuran kecil yang menggunakan ATmega 328 untuk versi Arduino Nano 3.0. *Board* ini memiliki semua pinnya dan dilengkapi dengan *port USB* mini untuk mendapatkan sumber daya tambahan dari luar, serta DC *power jack* untuk memungkinkan *bootloader* menggunakan 2 KB memori *flash*. Arduino Nano memiliki empat belas pin digital untuk digunakan sebagai pin *input* dan *output*. Jika diatur sebagai pin *input*, pin ini mengeluarkan tegangan 5 Volt untuk logika *HIGH* dan 0 Volt untuk logika *LOW* [28]. Dapat dilihat pada Gambar 2.7 dibawah ini.



Gambar 2. 7 Pin Layout Arduino Nano [28]

Tabel 2. 6 Bagian - bagian Board Arduino UNO.

Keterangan	Parameter
ATMega328	Mikrokontroler
5V	Tegangan Operasi
7-12V	Tegangan Input (rekomendasi)
6-20V	Tegangan Input (limit)
14 pins	Pin Digital I/O
8 pins	Pin Analog <i>Input</i>
40 mA	Arus DC per pin I/O
32 KB	Flash Memory
1 KB	EEPROM
2 KB	SRAM
16 Mhz	Kecepatan Pewaktu

2.2.9 Sensor pH Meter SKU: SEN0161

Sensor pH digunakan untuk mengukur tingkat keasaman air dan tanah. Untuk memastikan keakuratannya, mereka harus dikalibrasi secara berkala. Sensor pH adalah jenis sensor kimia yang mengambil nilai dari reaksi kimia dan kemudian mengubahnya menjadi besaran listrik. Ada dua jenis elektroda dalam sensor pH: elektroda kaca mengukur jumlah ion, dan elektroda referensi mengubah jumlah ion yang terbaca pada elektroda kaca menjadi tegangan analog. Prinsip operasinya adalah bahwa semakin banyak elektron yang ditangkap sampel, semakin asam cairannya, dan jika lebih banyak elektron yang dilihat, sampel cair bersifat basa. Jika nilai pH larutan kurang dari 7 dan lebih tinggi dari 7, larutan tersebut bersifat basa [29]. Pada Gambar 2.8 merupakan bentuk fisik dari sensor pH meter dengan tipe SKU: SEN0161.



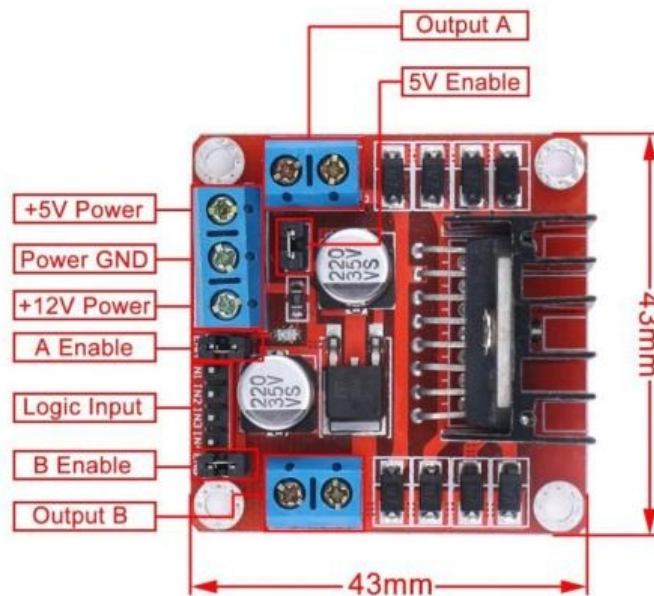
Gambar 2. 8 Sensor pH meter [30]

Spesifikasi sensor pH meter:

1. Mengukur pH dalam *range* 0 – 14
2. Waktu respon \pm 1 menit
3. Tegangan bekerja pada 5V
4. Tingkat akurasi \pm 0.1 pH (25°C)
5. Sensor pH dengan konektor BNC (*Bayonet Neill Concelman*) sebagai penghubung antara modul dengan sensor
6. Mengukur suhu antara 0°C - 60°C
7. Terdapat potensio penyesuaian
8. Terdapat *power indicator LED* [30].

2.2.10 Driver Motor L298N

Driver motor L2986 merupakan module *driver* motor DC yang berfungsi untuk mengontrol kecepatan serta arah perputaran motor DC. IC L298 merupakan sebuah IC tipe *H-bridge* yang mampu mengendalikan beban induktif seperti relay, *solenoid*, motor DC dan motor *stepper*. Pada IC L298 terdiri dari *transistor transistor logic* (TTL) dengan gerbang NAND yang berfungsi untuk memudahkan dalam menentukan arah putaran suatu motor DC maupun motor *stepper*. Kelebihan dari *driver* motor L298N yaitu tingkat kepresisiannya dalam mengontrol motor sehingga motor lebih mudah untuk di kontrol [31]. Dapat dilihat pada Gambar 2.9 dibawah ini.



Gambar 2. 9 Dimensi dan fungsi pin *driver* L298N [31]

Tabel 2. 7 Brief Data *Driver* L298N

Tegangan Masukan	3.2 V – 40 Vdc
<i>Driver</i>	L298N Dual H Bridge DC Motor <i>Driver</i>
Power Supply	DC 5V – 35 V
Arus Maksimal	2 Amp
Arus Masukan	0 – 36mA
Rentang tegangan masukan	Rendah: $-0,3 \text{ V} \leq V_{in} \leq 1,5\text{V}$ (sinyal kontrol tidak valid) Tinggi: $2,3 \text{ V} \leq V_{in} \leq V_{ss}$ (sinyal kontrol aktif)
Daya Maksimal	20 W
Dimensi	43 x 43 x 26mm

2.2.11 Pompa DC (*Direct Current*) 12V

Pompa arus searah DC (*Direct Current*) merupakan jenis pompa yang dimaksudkan untuk bekerja dengan tegangan listrik sebesar 12volt dari sumber arus searah. Pompa ini biasanya digunakan dalam berbagai aplikasi yang membutuhkan aliran fluida dengan menggunakan sumber daya DC. Pompa ini merupakan versi baru yang canggih dengan pompa air membran yang dapat menghisap sendiri yang memiliki daya hisap yang lebih tinggi daripada RS-360, diameter saluran keluar yang lebih besar, dan penahan karet membantu menjaga motor agar terikat pada objek tetap. Pompa ini tidak boleh ditempatkan di dalam air (cairan) saat digunakan, tidak dapat digunakan di dalam air [32]. Dapat dilihat pada Gambar 2.10 dibawah ini.



Gambar 2. 10 Pompa DC [32]

Tabel 2. 8 Spesifikasi Pompa DC

Spesifikasi	Keterangan
Ukuran	92 x 46 x 35 mm
Tegangan Yang Bekerja	6 – 12 V
Arus Yang Bekerja	0.5 – 0.7A
Maksimal Penghisapan	2m
Maksimal Pengaliran	700ml per 30S
Maksimal Temperature Air	Up to 80°C

2.2.12 LCD (*Liquid Crystal Display*) 16x2 I2C

LCD 16x2 terbuat dari bahan kristal cair dan dapat menampilkan tampilan karakter huruf atau angka. Sangat hemat energi dan memiliki bentuk yang kecil. LCD 16x2 memiliki 2 baris dan 16 karakter, serta 16 pin yang dapat dihubungkan ke komponen elektronika lainnya. Setiap satu dari 16 pin memiliki fungsinya sendiri [33]. Dapat dilihat pada Gambar 2.11 dibawah ini.



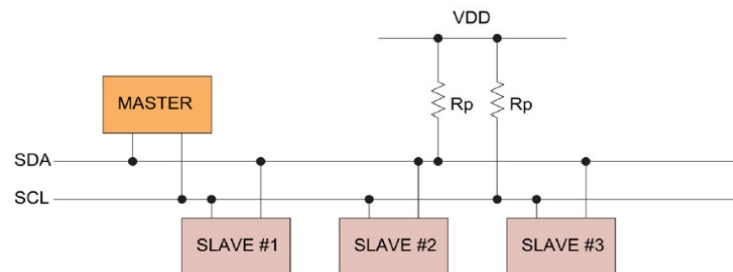
Gambar 2. 11 LCD 16x2 [33]

Tabel 2. 9 Spesifikasi LCD 16x2

Pin	Deskripsi
(Ground/source)	Menghubungkan terminal GND ke sumber daya
(VCC/source pin)	Pin tegangan pada layar, untuk menghubungkan pin sumber tegangan dari sumber listrik
(V0/VEE)	Mengatur kontras atau pencahayaan layar LCD
(Register select/control pin)	Pin yang memilih register command atau register data
(Read/write/pin control)	Mengendalikan tampilan operasi baca atau tulis
(Enable/kontrol pin)	Dihubungkan ke pin mikrokontroler agar mendapat sinyal <i>trigger</i>
(Pin data)	Mengirim data ke tampilan
(Pin +)	Dihubungkan ke VCC (5V)
(Pin -)	Dihubungkan ke GND

2.2.13 Inter Integrated Circuit

I2C (*Inter Integrated Circuit*) merupakan protocol komunikasi IC ke IC secara *serial data* (SDA) dan *serial clock* (SCL) dikenal sebagai I2C. Protokol ini memiliki bagian yang disebut master (perangkat pengendali) dan *slave* (perangkat yang dikendalikan). Master adalah mikrokontroler, sedangkan *slave* adalah IC RTC atau perangkat lainnya [34]. Dapat dilihat pada gambar 2.12 dibawah ini.



Gambar 2. 12 Komunikasi I2C [34]

Dalam kasus LCD 16x2, biasanya mikrokontroler akan berperan sebagai Master memiliki kendali atas jalur *serial clock* (SCL) dan jalur *serial data* (SDA). Mikrokontroler menginisiasi komunikasi dengan mengirimkan sinyal *start* (S) pada jalur SDA saat SCL masih tinggi. Kemudian, mikrokontroler mengirim alamat *Slave* (alamat LCD) beserta bit R or W (*read/write*) yang menunjukkan apakah operasi yang akan dilakukan adalah penulisan atau pembacaan. Setelah itu, Master akan mengirimkan *byte-byte* data atau instruksi kepada Slave (LCD) melalui jalur SDA. Sedangkan LCD akan berperan sebagai Slave, LCD menerima perintah atau data dari Master. LCD akan memantau jalur SDA dan SCL untuk mendeteksi *start condition* (S) dan alamat yang sesuai. Setelah mendeteksi alamat yang cocok, LCD akan menanggapi dengan mengirimkan ACK (*acknowledge*) atau NACK (*not acknowledge*) kepada Master tergantung pada status ketersediaan atau penerimaan data [34].

2.2.14 Power Supply

Switching power supply, juga dikenal sebagai *switched mode power supply* (SMPS), adalah catu elektronik yang memiliki peraturan *switching* yang disediakan untuk memenuhi kebutuhan tegangan keluaran. Sebuah daya pengubah (SMPS) adalah daya pengubah yang dapat mengirimkan daya dari sumber ke beban yang ideal tanpa mengalami kerugian. Pengubah bertanggung jawab untuk memberikan

tegangan keluaran pada tingkat yang berbeda dari tegangan masukan[25]. Pada penelitian ini menggunakan daya 12V, dapat dilihat pada Gambar 2.13.



Gambar 2. 13 Switching Power Supply [35]

Tabel 2. 10

Spesifikasi	Keterangan
Sumber Tegangan Input	220 VAC
Tegangan Output	12 VDC
Daya Maksimal	3A (36 Watt)
Dimensi	9 x 6 x 3.5 cm