

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Pabrik kimia merupakan susunan atau rangkaian berbagai unit pengolahan yang terintegrasi satu sama lain secara sistematis dan rasional. Tujuan pengoperasian pabrik kimia secara keseluruhan adalah mengubah bahan baku menjadi produk yang lebih bernilai guna. Dalam pengoperasiannya pabrik akan selalu mengalami permasalahan seperti regulasi atau pengontrolan berbagai variabel proses dan gangguan (*disturbance*) dari lingkungan eksternal. Selama beroperasi, pabrik diharuskan mempertimbangkan aspek keteknikan, keekonomisan, dan kondisi sosial agar proses yang terjadi di dalam pabrik tidak terpengaruh secara signifikan oleh perubahan-perubahan eksternal tersebut dan tercapai proses yang stabil.

Agar proses selalu stabil, dibutuhkan instalasi alat-alat pengendalian. Alat-alat pengendalian yang dimaksud antara lain kontroler, sensor, aktuator, sistem alarm, pengendali jarak jauh, dan lain sebagainya. Alat-alat pengendalian ini dipasang dengan tujuan menjaga keamanan dan keselamatan kerja. Selain itu, dapat berfungsi untuk memenuhi spesifikasi produk yang diinginkan, menjaga peralatan proses dapat berfungsi sesuai yang diinginkan dalam desain, menjaga operasi pabrik tetap ekonomis, dan memenuhi persyaratan lingkungan.

Untuk memenuhi persyaratan di atas, diperlukan pengawasan (*monitoring*) yang terus menerus terhadap operasi pabrik kimia dan intervensi dari luar (*external intervention*) untuk mencapai tujuan operasi. Hal ini dapat terlaksana melalui suatu rangkaian peralatan (alat ukur, pengendali, dan komputer) serta intervensi manusia (*plant managers* dan *plants operators*) yang secara bersama membentuk *control system*. Dalam pengoperasian pabrik, diperlukan berbagai prasyarat dan kondisi operasi tertentu sehingga diperlukan usaha-usaha pemantauan terhadap kondisi operasi pabrik dan pengendalian proses supaya kondisi operasinya stabil (Liu *et al.*, 2023).

### 1.2 Rumusan Masalah

Dalam keberjalanan suatu proses produksi di industri dibutuhkan penjagaan untuk kualitas dan kuantitas dari suatu proses tersebut. Penjagaan tersebut merupakan suatu sistem yang disebut sebagai pengendalian proses. Pengendalian proses dilakukan untuk menekan atau mengurangi *human error* dan meningkatkan efisiensi dalam keakuratan reaksi sebuah alat yang bekerja

secara otomatis terhadap suatu gangguan dibandingkan dengan alat yang digerakkan secara manual. Maka dari itu diperlukan pemahaman dalam pengoperasian sistem pengendali dan variasi dari segi *proportional* (P), *proportional integral* (I), dan *proportional integral derivative* (PID).

### 1.3 Tujuan Praktikum

1. Mampu mengoperasikan suatu proses dengan sistem pengendali.
2. Mampu mengevaluasi proses dengan variasi sistem pengendali umpan balik atau *feedback controller* (*Proportional* (P), *Proportional Integral* (I), dan *Proportional Integral Derivative* (PID)).
3. Mampu mengkaji grafik hubungan antara *level* dan *elapsed time* terhadap *set point*, interval data, PID, dan variasi *orifice*.

### 1.4 Manfaat Praktikum

1. Mahasiswa diharapkan mampu mengetahui pengoperasian suatu proses dengan sistem pengendali.
2. Mahasiswa dapat mampu mengevaluasi proses dengan variasi sistem pengendali umpan balik atau *feedback controller* (*Proportional* (P), *Proportional integral* (I), dan *Proportional Integral Derivative* (PID)).
3. Mahasiswa diharapkan mampu mengkaji grafik hubungan antara *level* dan *elapsed time* terhadap *set point*, interval data, PID, dan variasi *orifice*.

Process

Laboratory

## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Pengendalian Proses

Sistem pengendalian adalah susunan komponen fisik yang dirancang sedemikian rupa untuk mengendalikan sistem atau sistem lain yang berhubungan dengan sebuah proses dalam keadaan *open loop* atau *close loop* (Benner, 1993). Secara umum, pengendalian proses adalah pengendalian terhadap suatu atau beberapa besaran sehingga tetap berada pada suatu harga atau *range* tertentu. Sebagai contoh, ketinggian fluida dalam reaktor harus dipertahankan pada angka tertentu sehingga diperlukan control valve untuk mengatur besar kecil laju alir yang masuk pada reaktor. Hasil yang presisi mustahil untuk dilakukan secara manual sehingga diperlukan peralatan otomatis untuk menyelesaikan masalahnya. Hampir semua proses dalam dunia industri membutuhkan peralatan-peralatan otomatis untuk mengendalikan parameter-parameter prosesnya. Otomatisasi tidak saja diperlukan demi kelancaran operasi, keamanan, ekonomi, maupun mutu produk, tetapi lebih merupakan kebutuhan pokok. Dalam sebuah industri tidak mungkin jika tidak melibatkan pengendalian proses, contohnya pengendalian disuatu proses pengilangan minyak.

Ada banyak parameter yang harus dikendalikan di dalam suatu proses. Di antaranya yang paling umum adalah tekanan (*pressure*) di dalam sebuah *vessel* atau pipa dan reaktor, aliran (*flow*) di dalam pipa, suhu (*temperature*) di unit proses seperti *heat exchanger*, atau permukaan zat cair (*level*) di sebuah tangki, dan konsentrasi bahan kimia (Dong *et al.*, 2024). Selain itu, ada beberapa parameter lain diluar keempat parameter di atas yang cukup penting dan juga perlu dikendalikan karena kebutuhan spesifik proses, diantaranya: pH di industri petrokimia, *water cut* (BS & W) di ladang minyak mentah, warna produk di suatu fasilitas pencairan gas (NGL) dan sebagainya.

Pemahaman terhadap proses merupakan salah satu faktor utama dalam keberhasilan pengendalian parameter yang harus dikendalikan. Dalam praktiknya, sistem pengendalian otomatis memiliki peran penting karena dapat meningkatkan kinerja sistem dinamis, memperbaiki kualitas produk, menurunkan biaya produksi, meningkatkan kecepatan produksi, serta mengurangi pekerjaan rutin yang sebelumnya dilakukan oleh manusia. Peran operator manual kini sebagian besar telah digantikan oleh alat yang disebut *controller*. Misalnya, tugas membuka dan menutup katup (*valve*) tidak lagi dilakukan secara manual, melainkan dikendalikan oleh *controller*. Agar

pengendalian ini berjalan otomatis, katup perlu dilengkapi dengan alat yang disebut *actuator*, sehingga katup tersebut berubah fungsi menjadi *control valve*. Seluruh peralatan yang digunakan dalam proses pengendalian ini, seperti *controller* dan *control valve*, dikenal sebagai bagian dari sistem instrumentasi pengendalian proses.

*Controller* merupakan sebuah sistem yang ditambahkan pada *plant* untuk mendapatkan karakteristik sistem kinerja sistem secara keseluruhan sesuai dengan yang diharapkan. Salah satu metode pengendalian yang umum digunakan adalah pengendalian PID (*Proportional-Integral-Derivative*). Pengendalian PID berfungsi untuk memperbaiki performa sistem melalui peningkatan respon terhadap perubahan kondisi. Dengan penerapan kontrol PID, sistem dapat mengurangi bahkan menghilangkan kesalahan pada kondisi tunak (*steady-state error*) serta meningkatkan kecepatan respon dalam mencapai nilai *set point* yang diinginkan.

## 2.2 Pengendalian *Proportional* (P)

Pengendalian proposional merupakan suatu pengendali yang isyarat keluarannya ( $p(t)$ ) proporsional terhadap kesalahan ( $e(t)$ ), yaitu beda antara *set point*-nya dengan hasil pengukuran yang secara matematis dapat dinyatakan sebagai berikut:

$$p(t) = \bar{p} + K_c \times \text{error}(t) \quad (2.1)$$

Berdasarkan pendekatan tersebut dimana  $K_c$  adalah *proportional gain* yang menunjukkan *responsiveness controller* terhadap proses *setting up* atau koreksi yang diberikan untuk mencapai *set poin*. Proportional gain disesuaikan untuk mengatur seberapa responsif sistem terhadap perbedaan antara *set point* dan variabel proses.

Pengendali proporsional memiliki 2 parameter yaitu: pita proporsional (*proportional band*) dan konstanta proporsional. Derah kerja efektif *controller* dicerminkan oleh *proportional band*, sedangkan konstanta proporsional menunjukkan nilai faktor penguatan terhadap sinyal kesalahan,  $K_c$ . Hubungan antara *proportional band* (PB) dengan konstanta *proportional* ( $K_c$ ) ditunjukkan sebagai berikut:

$$PB = \left( \frac{1}{K_c} \right) \times 100\% \quad (2.2)$$

Sistem pengendalian ini merupakan bentuk sistem pengendalian proses yang sangat sederhana dengan proses yang sangat cepat terhadap *set point* dan gangguan pada proses, tetapi mempunyai karakteristik besaran *steady state error*

yang besar.

Ciri-ciri pengontrol proporsional:

1. Apabila nilai  $K_C$  kecil, pengontrol proporsional hanya mampu melakukan koreksi kesalahan yang kecil sehingga akan menghasilkan respon sistem yang lambat (menambah *rise time*).
2. Apabila nilai  $K_C$  dinaikkan, respon/tanggapan sistem akan semakin cepat mencapai keadaan mantapnya (mengurangi *rise time*).
3. Namun, jika nilai  $K_C$  diperbesar sehingga mencapai harga yang berlebihan, akan mengakibatkan sistem bekerja tidak stabil atau respon sistem akan berosilasi/terjadinya *overshoot*.
4. Nilai  $K_C$  dapat diatur sedemikian sehingga mengurangi *steady state error*, tetapi tidak menghilangkannya.

### 2.3 Pengendalian *Integral* (I)

Pengendalian integral berfungsi untuk menghilangkan kesalahan (*error*) secara keseluruhan, terutama *steady-state error*. Tujuan utama dari komponen integral adalah memastikan bahwa kesalahan antara *set point* dan output sistem menjadi nol. Cara kerjanya yaitu dengan menghitung kesalahan secara terus-menerus seiring waktu, lalu menjumlahkannya (integrasi) untuk menghasilkan sinyal kendali.

$$p(t) = \bar{p} + \frac{1}{\tau_i} \int_0^t \text{error}(t_i) dt_i \quad (2.3)$$

Dengan mengintegrasikan error, nilai keluaran sistem dapat dikoreksi secara akumulatif, terutama saat error terjadi dalam jangka waktu lama. Namun, pengendalian integral jarang digunakan secara tunggal karena memiliki beberapa kelemahan. Salah satu masalah umum yang muncul adalah reset windup, yaitu kondisi ketika akumulasi error menyebabkan sinyal kendali melebihi batas kemampuan kontrol valve (*control valve*) yang sudah dalam kondisi jenuh (terbuka penuh), sehingga sistem tidak lagi merespons perintah kendali. Hal ini dapat menyebabkan *overshoot* atau bahkan membuat kontroler kehilangan fungsinya sebagai pengendali. Untuk mengatasi hal ini, biasanya digunakan metode tambahan seperti anti-reset windup agar respon sistem tetap stabil, responsif dan terkendali.

### 2.4 Pengendalian *Derivative* (D)

Pengendalian *Derivative* berfungsi untuk mengantisipasi kelakukan isyarat kesalahan (*error*) yang akan terjadi dengan memperhatikan kecepatan perubahan

dan memprediksi perubahan sehingga sistem bereaksi lebih cepat dan cerdas terhadap perubahan. *Derivative controller* dikenal dengan aksi kecepatan, pre-act, atau pengendali antisipatif. Fungsinya adalah mengantisipasi kelakukan isyarat kesalahan yang akan terjadi dengan memperhatikan kecepatan perubahan dan memprediksi perubahan. Pengendali *derivative* dapat mempercepat respon sistem sehingga akan mengurangi *overshoot* (sinyal melebihi *set point*).

$$p(t) = \bar{p} + \tau_i \frac{d \text{error}(t)}{dt} \quad (2.4)$$

Pengendali derivatif membantu sistem merespons lebih cepat di awal dan mencegah sinyal melebihi nilai yang diinginkan (*set point*). Cara kerjanya adalah dengan melihat seberapa cepat kesalahan (*error*) berubah. Jika error meningkat dengan cepat, pengendali akan segera bereaksi sebelum error menjadi terlalu besar. Sebaliknya, jika error turun dengan cepat, pengendali akan mengurangi reaksinya agar tidak turun secara berlebihan. Dengan begitu, sistem menjadi lebih stabil, tidak mudah melewati batas, dan lebih cepat mencapai kondisi stabil.

## 2.5 Pengendalian *Proportional Integral* (PI)

Pengendali PI adalah sistem pengendali gabungan antara pengendali proporsional dan integral. Pengendali integral digunakan untuk mengeliminasi *error* dengan mengintegralkan *error* pada periode tertentu hingga *error* “nol” atau tidak ada *error* sama sekali. Bentuk keluaran pengendali tergantung pada integral dari kesalahan isyarat pada seluruh waktu, dimana:

$$\Delta Q_{\text{output}} = K_i \int i. dt \quad (2.5)$$

atau

$$p(t) = \bar{p} + K_c(\text{error}(t) + \frac{1}{\tau_i} \int_0^t \text{error}(t_i) dt_i) \quad (2.6)$$

$K_i$  = *integral gain* atau “*reset rate*” (*repeat/minute*)

$K_c$  = besar koreksi yang diberikan

$\tau_i$  = waktu integral atau waktu *reset*

Pengendalian integral dikenal pula sebagai “*reset*” yang mempunyai respon yang relatif lambat tapi cukup efektif untuk pengendalian proses yang berlangsung cepat, mengandung unsur gangguan yang besar dan didominasi oleh adanya sifat *deadtime* pada transportasi produk. Pengaruhnya terhadap *steady state error* relatif kecil. Pengendalian ini biasa digunakan untuk mengurangi adanya *offset* antara *set point* dan *process variable*.



## 2.6 Pengendalian *Proportional Integral Derivative* (PID)

Pengendalian *Proportional Integral Derivative* (PID) adalah suatu jenis pengendalian sistem yang menggabungkan tiga elemen kontrol yang berbeda, yakni proporsional, integral, dan derivatif. Kontrol PID ini digunakan untuk menentukan nilai  $K_p$ ,  $T_i$ , dan  $T_d$ . Derivatif kontrol ini umumnya dikenal juga sebagai “rate”. Nilai dari parameter ini pada dasarnya berarti seberapa jauh di detik kedepannya ingin memprediksi laju perubahannya. Model persamaan yang digunakan adalah:

$$\Delta Q_{\text{output}} = K_d \frac{d \text{error}(t)}{dt} \quad (2.7)$$

$$p(t) = \bar{p} + K_c \left( \text{error}(t) + \frac{1}{\tau_i} \int_0^t \text{error}(t_i) dt_i + \tau_D \frac{d \text{error}(t)}{dt} \right) \quad (2.8)$$

$K_d$  = *time constant*

$\tau_D$  = waktu *derivative*

Model *derivative* tidak pernah berdiri sendiri tetapi selalu bersama proporsional atau proporsional integral karena kontrol *derivative* hanya akan berubah saat ada perubahan *error*, sehingga saat *error* statis (tetap), kontrol ini tidak akan bereaksi. Dalam pengendalian PID ada tiga parameter yang dapat diatur yaitu  $K_c$ ,  $\tau_i$  dan  $\tau_D$ . Dalam kontrol PID sendiri perlu dihindari terjadinya *derivative kick* (gangguan yang besar) dengan mengatur parameter-parameter yang ada.

## 2.7 Metode Ziegler-Nichols

*Tuning* adalah proses menyetel nilai-nilai parameter dalam kontroler, seperti PID (*Proportional, Integral, Derivative*), agar sistem dapat bekerja dengan baik sesuai yang diinginkan. Dalam sistem kontrol, *tuning* sangat penting karena menentukan seberapa cepat, stabil, dan akurat sistem merespons perubahan atau gangguan. Jika *tuning* dilakukan dengan benar, sistem akan merespons dengan cepat tanpa berosilasi berlebihan, tidak terlalu lambat, dan mampu mencapai target atau set point secara efisien. Proses *tuning* ini biasanya dilakukan dengan cara mencoba-coba nilai parameter, menggunakan metode matematis, atau pendekatan eksperimental, tergantung pada jenis sistem dan informasi yang tersedia.

Salah satu metode *tuning* yang digunakan adalah metode Ziegler-Nichols. Metode Ziegler-Nichols adalah cara praktis untuk menyetel kontroler PID agar sistem bisa bekerja dengan cepat dan stabil, tanpa perlu tahu rumus atau model

matematis dari sistem tersebut. Metode ini dilakukan dengan cara mengamati bagaimana sistem merespons saat kita mengubah inputnya, lalu dari pengamatan itu kita menentukan nilai-nilai parameter PID yang tepat. Tujuannya adalah agar sistem bisa merespons dengan cepat namun tetap stabil, biasanya dengan overshoot sekitar 25%. Karena caranya mudah dan tidak memerlukan perhitungan yang rumit, metode ini sangat berguna di berbagai bidang industri, seperti pengaturan suhu, tekanan, aliran, atau sistem otomatis lainnya yang membutuhkan respon cepat dan efisien.

Metode Ziegler-Nichols untuk penalaan kontroler PID dimulai dengan mengatur kontroler PID dengan nilai integral ( $K_i$ ) dan diferensial ( $K_d$ ) sama dengan nol, hanya gain proporsional ( $K_p$ ) yang aktif. Kemudian, tingkatkan secara bertahap nilai  $K_p$  hingga sistem mulai menunjukkan perilaku osilasi (fluktuasi) yang tidak stabil. Nilai  $K_p$  pada titik ketika sistem mengalami osilasi ini disebut dengan  $K_{MAX}$  (nilai maksimum di mana sistem masih dapat beroperasi tetapi sudah mulai berosilasi). Selama osilasi, frekuensi yang terbentuk dinamakan  $f_0$ . Setelah mencapai  $K_{MAX}$ , turunkan nilai  $K_p$  ke tingkat yang lebih rendah, yang biasanya ditetapkan pada sekitar 50-75% dari  $K_{MAX}$ . Terakhir, gunakan nilai  $f_0$  untuk menetapkan nilai integral ( $T_i$ ) dan diferensial ( $T_d$ ) gain berdasarkan rumus yang ditentukan oleh Ziegler-Nichols. Umumnya, rumus untuk menetapkan parameter PID adalah sebagai berikut:

$$K_p = 0,6 \times K_{MAX} \quad (2.9)$$

$$T_i = \frac{1}{2 \times f_0} \quad (2.10)$$

$$T_d = \frac{1}{8 \times f_0} \quad (2.11)$$

(Ellis, 2012)

## 2.8 PCT 50 Level Control



Gambar 2.1 PCT50 level control

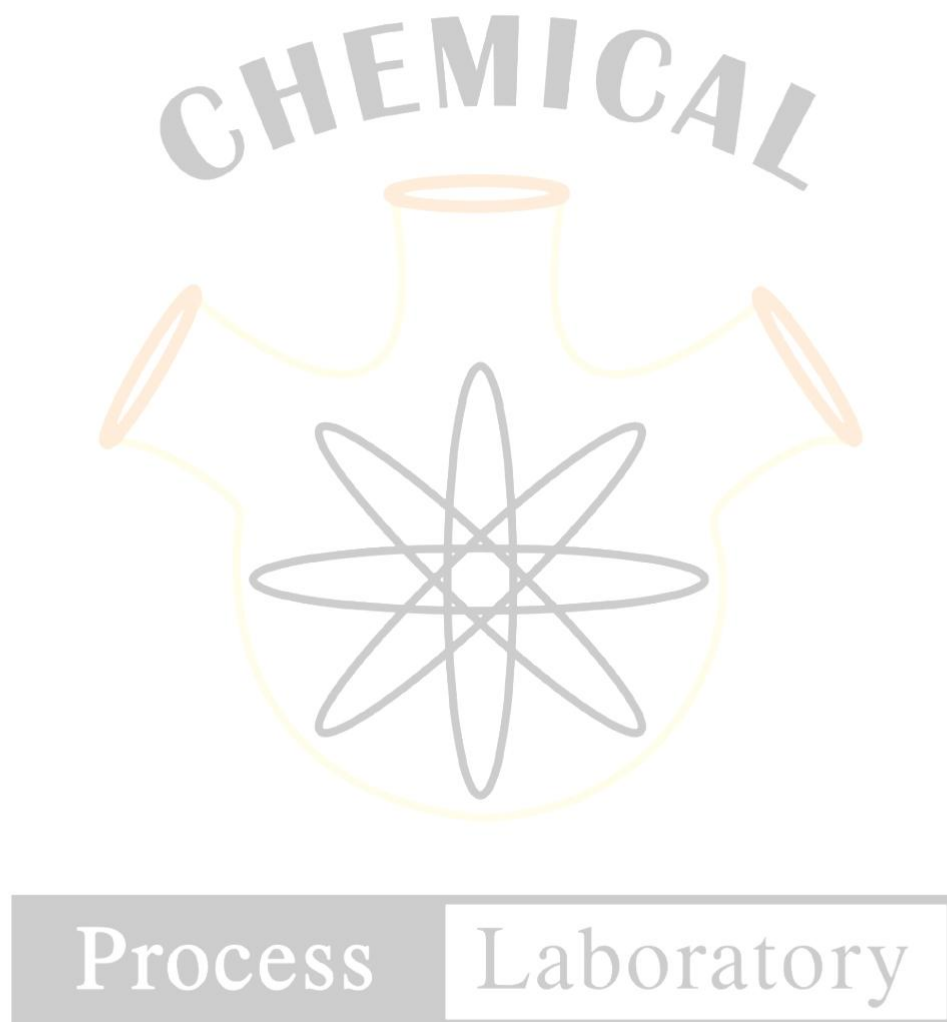


PCT 50 adalah salah satu alat proses kontrol level yang menggunakan air sebagai fluida kerja untuk keamanan dan kenyamanan penggunaan. Air yang disimpan di tangki penampung bawah ditransfer ke bejana proses atas (*upper process vessel*) melalui pompa sentrifugal berkecepatan variabel yang terendam. Konektor pelepas cepat memungkinkan tabung keluaran pompa dilepas dengan mudah, sehingga memudahkan proses priming pompa setelah tangki bawah diisi air. Pengaturan saluran masuk vertikal dalam bejana proses memungkinkan air yang masuk ke bejana tetap terlihat berapapun ketinggian air dan *integral non-valve* (*check valve*) yang bertujuan untuk mencegah air mengalir kembali (*back flow*) ke bejana penampung ketika kecepatan pompa dikurangi atau dihentikan. Katup bola sebaris (CV1) di atas konektor pelepas cepat memungkinkan aliran air yang masuk ke bejana proses divariasikan, tidak tergantung pada kecepatan pompa, agar sesuai dengan demonstrasi tertentu.

Ketinggian air di dalam bejana proses diukur dengan menggunakan sensor tekanan elektronik yang dipasang di tepi bejana. Satu sisi sensor tekanan terhubung didalam bejana proses dan sisi lainnya terbuka ke atmosfer sehingga memungkinkan tekanan di dalam bejana proses diukur secara relatif terhadap atmosfer. Oleh karena itu, sensor ini mengukur tingkat air di dalam bejana proses. Level ketinggian juga ditunjukkan pada skala di sisi bejana proses. Air mengalir dari bejana proses kembali ke tangki penampung bawah melalui dua saluran keluar di dasar bejana proses. Aliran melalui saluran keluar utama bersifat kontinyu. Aliran melalui *outlet* kedua dapat dijalankan dan dihentikan oleh katup solenoid (SOL) dengan kendali jarak jauh. Kedua *outlet* dilengkapi dengan katup bola yang dioperasikan secara manual (CV2 dan CV3) yang memungkinkan aliran air divariasikan secara terus menerus agar sesuai dengan demonstrasi tertentu. Kedua *outlet* juga dilengkapi dengan lubang yang dapat diubah-ubah (3 dan 5) yang memungkinkan aliran ditetapkan pada ukuran yang telah ditentukan. Ukuran lubang diubah dengan membuka tutup plastik yang berisi lubang dan menggantinya dengan ukuran lain yang diperlukan. Pemasangannya menggunakan segel cincin 'O' dan hanya membutuhkan pengencangan dengan tangan. Ukuran alternatif lubang biasa disebut dengan *orifice* disimpan dalam lubang berulir di bagian depan pelat dasar saat tidak digunakan.

*Overflow* dalam bejana proses berfungsi mengembalikan air ke tangki penampung sehingga pengisian berlebih pada bejana proses saat penggunaan dapat dicegah. Sensor tekanan yang mengukur level, pompa sentrifugal, dan katup solenoida terhubung ke *electrifcal interface* yang menggabungkan pengkondisian sinyal yang diperlukan, sehingga proses dapat dioperasikan

langsung dari PC menggunakan satu port USB. Perangkat lunak komputer yang disertakan dengan PCT 50 memungkinkan kontrol proses level dan pencatatan data respons menggunakan PC. Opsi lainnya, perangkat lunak ini memungkinkan pencatatan data hanya saat mengoperasikan proses dari jarak jauh menggunakan pengontrol PID. Ketika diisi dengan air, PCT 50 mandiri hanya membutuhkan pasokan listrik utama ke converter DC *in-line* dan koneksi ke PC melalui port USB. Unit ini dikeringkan menggunakan pembuangan air yang terletak di bagian belakang.



**BAB III**  
**METODE PRAKTIKUM**

**3.1 Rancangan Praktikum**

**3.1.1 Rancangan Praktikum**

Gambar 3.1 Rancangan praktikum

**3.1.2 Penetapan Variabel**

**3.2 Bahan dan Alat yang Digunakan**

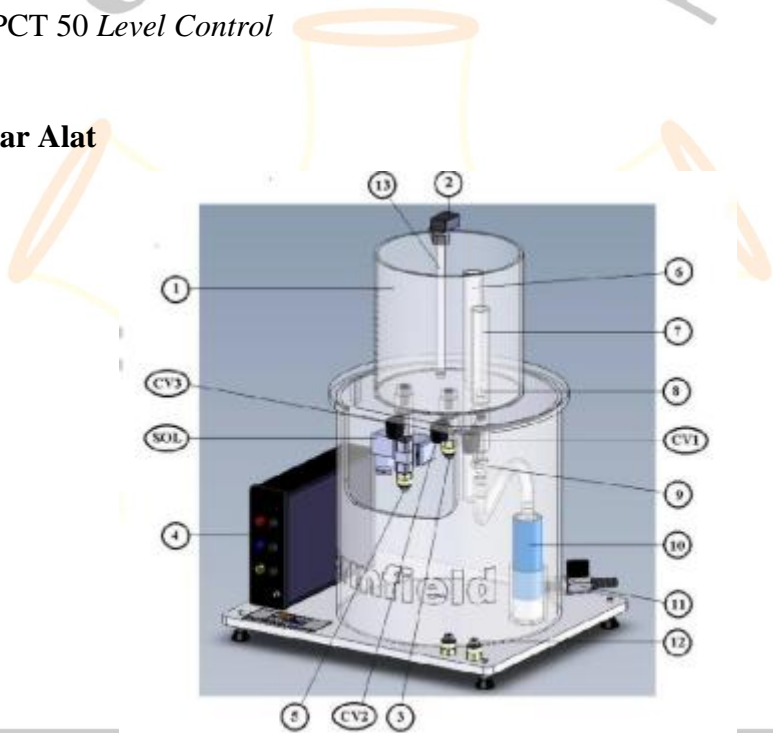
**3.2.1 Bahan**

*Aquadest 8 liter*

**3.2.2 Alat**

*PCT 50 Level Control*

**3.3 Gambar Alat**



Gambar 3.2 Alat PCT 50 level control dan bagian-bagiannya

- |                                      |  |
|--------------------------------------|--|
| (1) <i>Upper Process Vessel</i>      | (9) <i>Quick Release Conenector</i>      |
| (2) <i>Level Sensor</i>              | (10) <i>Centrifugal Pump</i>             |
| (3) <i>Manual Discharge Port</i>     | (11) <i>Water Discharge</i>              |
| (4) <i>Electrical interface</i>      | (12) <i>Alternative sizes of orifice</i> |
| (5) <i>Solenoid Discharge Port</i>   | (CV1) <i>Control Valve 1</i>             |
| (6) <i>Overflow</i>                  | (CV2) <i>Control Valve 2</i>             |
| (7) <i>Inlet</i>                     | (CV3) <i>Control Valve 3</i>             |
| (8) <i>Integral non-return valve</i> | (SOL) <i>Solenoid Drain Valve</i>        |

### 3.4 Cara Kerja

#### 3.4.1 Installing Software Armfield Level Control

1. *Instal driver* terlebih dahulu dengan pergi ke *this PC*, klik kanan lalu pilih *properties* kemudian pilih *device manager*.
2. Masukkan USB ke laptop, setelahnya akan muncul COM 5 dengan tanda seru. Pastikan PCT-50 telah menyala dengan indikator lampu hijau yang menyala.
3. Hilangkan tanda seru dengan meng-*update* atau memperbaharui *driver*. Cari folder PCT-50 lalu pilih ikon urutan kedua dari atas kemudian pilih *install* dengan mengikuti langkah-langkah pada *driver*. Laptop akan secara otomatis *restart* setelah proses *installing* selesai.
4. Pastikan USB terhubung dengan sempurna agar koneksi antara alat PCT-50 dengan laptop tidak terputus tiba-tiba. Pastikan juga saat hendak digunakan tidak dalam keadaan *scanning*.

#### 3.4.2 Pelaksanaan Praktikum

##### a. Kalibrasi Alat PCT-50

1. Pastikan bahwa katup penguras di bagian belakang tangki bak tertutup dengan rapat, lalu isi tangki bak bagian bawah dengan *aquadest* sebanyak 8 L di bawah lubang di bagian depan tangki.
2. Pastikan sensor kosong setiap akan melaksanakan praktikum. Bila ada air terjebak di dalam sensor, gunakan sikat gelas untuk memastikan air sensor kosong.
3. Pastikan *flow control valve* CV1 benar-benar terbuka pada inlet bejana proses sehingga *aquadest* dapat mengalir ke bejana saat pompa sedang berjalan.
4. Pastikan katup keluaran CV2 terbuka penuh di dasar bejana proses untuk memungkinkan *aquadest* kembali ke tangki bak saat bejana proses terisi *aquadest*.
5. Pastikan katup CV3 terbuka penuh di atas *solenoid valve* SOL untuk mengalirkan *aquadest* kembali ke tangki bak saat *solenoid valve* terbuka dan bejana proses berisi *aquadest*.
6. Hidupkan pompa dan pastikan *aquadest* dapat mengalir ke tangki proses. Bila tidak mengalir, lepaskan kaitan selang pompa atas isi dengan *aquadest* dengan cara menenggelamkannya ke *aquadest* yang berada di tangki bak.

7. Setelah *aquadest* dapat terpompa dengan lancar ke tangki bak, kosongkan tangki proses dan lakukan percobaan sesuai variabel.

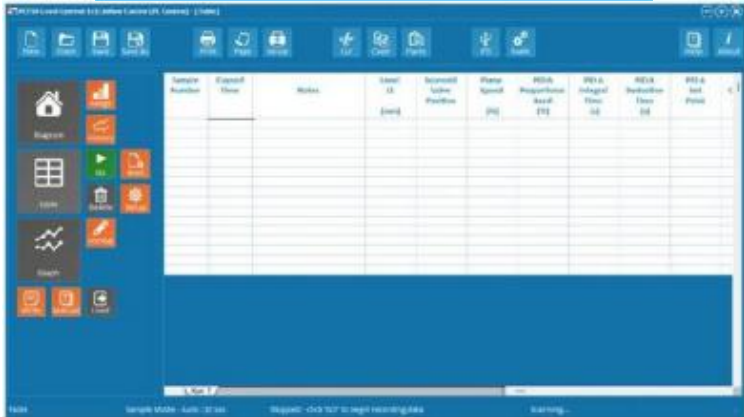
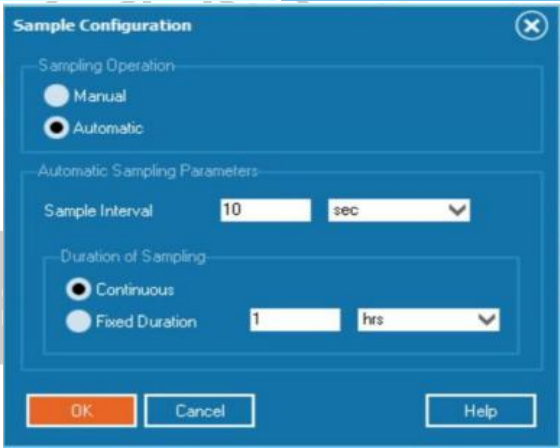
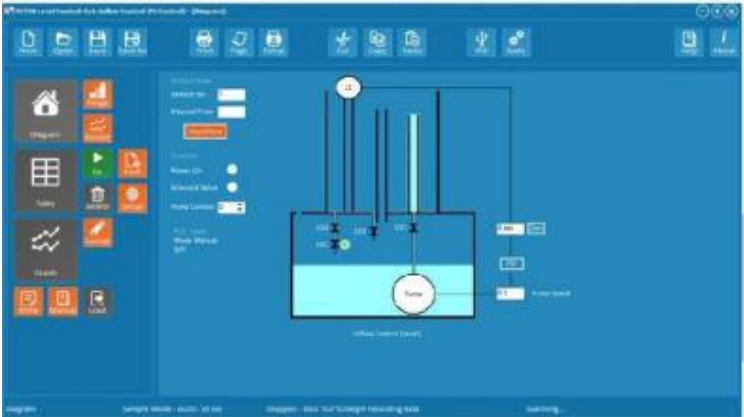
**b. Mencari *level* maksimum**

1. Memastikan tangki proses kosong.
2. Mengatur *sample interval* dan *fixed duration* pada menu *sample configuration* sesuai variabel yang telah ditentukan.
3. Mengatur variabel (*pump speed*, *set point*, *proportional band*, *integral time*, *derivative time*, dan *mode of operation*) yang telah di tentukan pada menu PID.
4. Operasikan alat sesuai dengan variabel yang ditentukan dengan mengklik power on serta go untuk merecord data yang didapat;
5. Amati grafik yang didapat pada menu grafik, ambil dan simpan data hasil praktikum sesuai dengan data yang tertera pada aplikasi PCT-50 (excel).
6. Jika waktu sudah mencapai yang diinginkan, lakukan penurunan *pump speed* untuk mengembalikan air ke tangki.
7. Lakukan langkah 2-6 dengan variabel yang berbeda (mode operasi).

**c. Mencari nilai PID**

1. Memastikan keadaan tangki bejana proses kosong.
2. Mengatur variabel (*pump speed*, *set point*, *proportional band*, *integral time*, dan *derivative time*,) yang telah ditentukan pada menu PID.
3. Mode pengoperasian dilakukan pada mode *automatic*
4. Operasikan alat sesuai dengan variabel yang ditentukan dengan mengklik power on serta go untuk merecord data yang didapat
5. Amati grafik, ambil dan simpan data hasil praktikum sesuai dengan data yang tertera pada aplikasi PCT-50 (excel).
6. Setelah semua dilakukan, lakukan pembersihan alat dengan membuka katup penguras sehingga air pada tangki terkuras keluar, pastikan air pada tangki dan bejana proses benar-benar hilang.

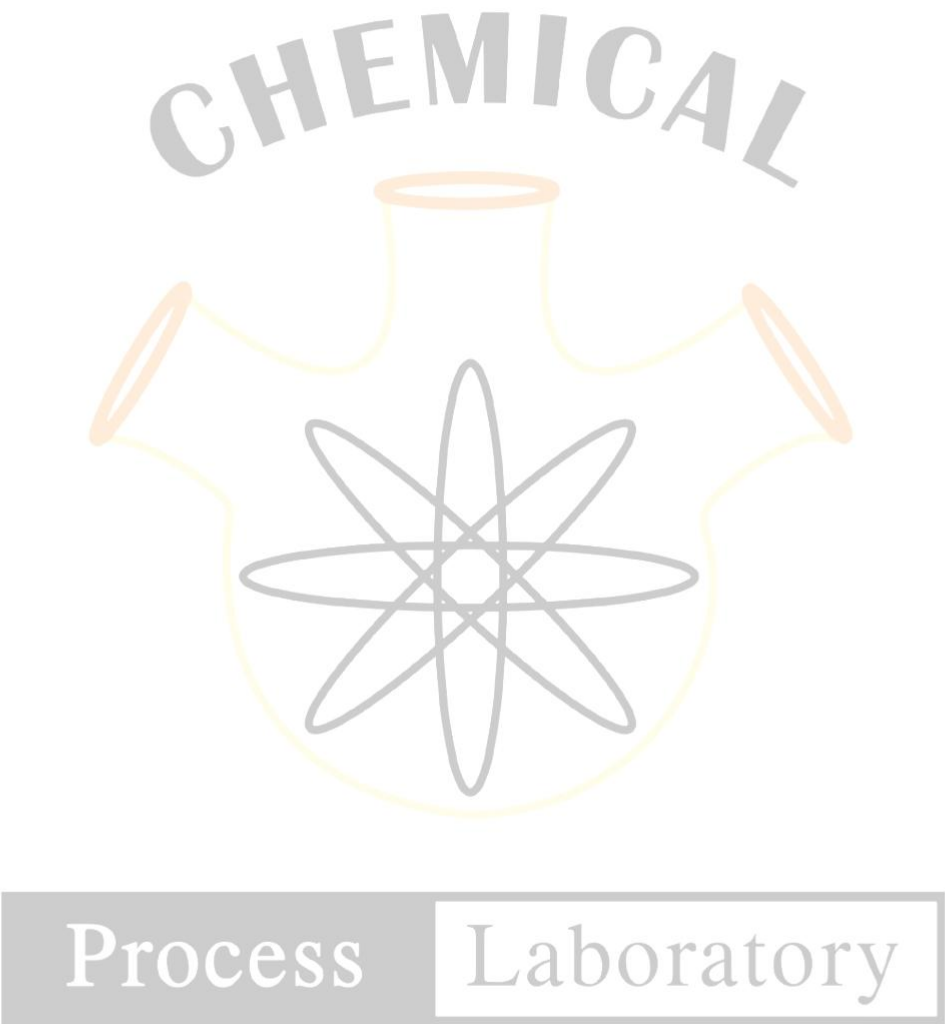
7. Setelah dibersihkan dan dikeringkan, alat dapat dimatikan secara bertahap.







Gambar 3.3 Langkah-langkah penggunaan *software* Armfield PCT 50



## DAFTAR PUSTAKA

- Bennet, S. (1993). Development of the PID controller. *IEEE Control Systems*, 13(6), 58-62. doi: 10.1109/37.248006
- Dong, W., He, F., Wang, J., Wu, N., & Li, X. (2004). Modeling and numerical analysis of PID-controlled phase-change transpiration cooling. *International Journal of Thermal Sciences*, 196, 108729. <https://doi.org/10.1016/j.ijthermalsci.2023.108729>.
- Ellis, G. (2012). Chapter 6 - Four Types of Controllers. *Control System Design Guide (Fourth Edition)*, 97-119. <https://doi.org/10.1016/B978-0-12-385920-4.00006-0>
- Liu, H., Yu, Q., & Wu, Q. (2023). PID control model based on back propagation neural network optimized by adversarial learning-based grey wolf optimization. *Applied Sciences*, 13(8), 4767. <https://doi.org/10.3390/app13084767>.
- Seborg, D. E., Edgar, T. F., Mellichamp, D. A., & Doyle III, F. J. (2016). *Process Dynamics and Control Fourth Edition*. United States: John Willey & Sons, Inc.
- Smith, C.A. & Armando B. Corripio. (1997). *Principles and Practice of Automatic Proses Control Second Edition*. New York: John Willey & Sons, Inc.

Process

Laboratory