

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pabrik kimia merupakan susunan berbagai unit pengolahan yang terintegrasi untuk mengubah bahan baku menjadi produk yang memiliki nilai guna. Dalam pengoperasiannya, kondisi proses dapat mengalami perubahan akibat perubahan kondisi operasi maupun gangguan (*disturbance*) dari lingkungan eksternal. Untuk menjaga kondisi operasi tersebut, diperlukan pengawasan (*monitoring*) secara terus-menerus terhadap operasi pabrik kimia serta tindakan pengendalian untuk mencapai tujuan operasi. Hal ini dapat terlaksana melalui suatu rangkaian peralatan, seperti alat ukur, pengendali, dan komputer, serta keterlibatan manusia seperti *plant managers* dan *plant operators* yang secara bersama-sama membentuk *control system*.

Pengendalian proses dilakukan dengan mengatur variabel operasi berdasarkan kondisi proses yang terukur. Sistem pengendalian umumnya terdiri atas sensor atau alat ukur, kontroler, dan aktuator yang bekerja secara terintegrasi untuk mempertahankan variabel proses pada nilai yang diinginkan (*set point*). Pada sistem pengendalian umpan balik (*feedback control*), kondisi aktual proses diukur dan dibandingkan dengan *set point* untuk menghasilkan *error*. *Error* tersebut kemudian digunakan oleh kontroler untuk menentukan tindakan pengendalian sehingga respons sistem dapat dikoreksi terhadap perubahan kondisi proses (Liu *et al.*, 2023).

Salah satu variabel proses yang penting untuk dikendalikan adalah *level* atau ketinggian fluida dalam tangki. Pengendalian *level* diperlukan untuk menjaga kondisi operasi tangki tetap berada pada rentang yang diinginkan serta menjaga keseimbangan antara aliran masuk dan aliran keluar. Salah satu metode pengendalian yang dapat digunakan adalah kontrol PID (*Proportional Integral Derivative*), yang bekerja berdasarkan tiga parameter pengendalian, yaitu *proportional band*, *integral time*, dan *derivative time*. Perubahan parameter tersebut dapat memengaruhi karakteristik respons sistem, seperti kecepatan respons, *overshoot*, dan kestabilan sistem. Oleh karena itu, pemahaman mengenai pengoperasian sistem pengendalian *level* serta pengaruh parameter kontrol terhadap respons sistem diperlukan untuk memahami penerapan pengendalian proses pada industri.

1.2 Rumusan Masalah

Dalam proses produksi di industri, pengendalian variabel proses diperlukan untuk menjaga kestabilan, keamanan, kualitas, dan efisiensi operasi. Pengendalian tersebut dilakukan dengan memantau kondisi proses dan menyesuaikan variabel operasi agar tetap berada pada kondisi yang diinginkan. Salah satu variabel penting yang harus dikendalikan dalam proses industri adalah *level* (ketinggian fluida), karena perubahan *level* dapat memengaruhi kondisi operasi suatu unit proses. Oleh karena itu, praktikum ini dilakukan untuk mengkaji pengoperasian sistem pengendalian *level* menggunakan PCT 50 *Level Control*, pengaruh mode pengoperasian terhadap waktu pencapaian *level* maksimum, serta pengaruh perubahan *proportional band*, *integral time*, dan *derivative time* terhadap respons sistem.

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, dan PID.

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, dan PID.

BAB II

TINJAUAN PUSTAKA

2.1 Pengendalian Proses

Sistem pengendalian adalah susunan komponen fisik yang dirancang sedemikian rupa untuk mengendalikan sistem yang berhubungan dengan sebuah proses dalam keadaan *open loop* atau *close loop* (Benner, 1993). Sistem *open loop* merupakan sistem pengendalian yang tidak menggunakan *feedback* dari keluaran proses untuk menentukan aksi pengendalian. Sebaliknya, sistem *closed loop* menggunakan *feedback* dari keluaran proses untuk dibandingkan dengan *set point*. Selisih antara nilai aktual dan *set point* disebut *error*, yang digunakan oleh *controller* untuk menentukan aksi pengendalian. Dengan demikian, sistem *closed loop* dapat melakukan koreksi terhadap perubahan kondisi proses. Prinsip pengendalian tersebut kemudian diterapkan dalam pengendalian proses, yaitu 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.

Pada sistem kendali *closed loop*, terdapat beberapa komponen utama yang saling berhubungan, yaitu *controller*, *actuator*, *plant*, dan sensor. *Controller* berfungsi menentukan aksi pengendalian berdasarkan selisih antara *set point* dan nilai keluaran, sedangkan *actuator* mengubah sinyal pengendalian menjadi aksi pada proses. *Plant* merupakan proses yang dikendalikan, sementara *sensor* mengukur keluaran proses dan mengirimkan sinyal *feedback* ke *controller*.

Terdapat berbagai parameter proses yang perlu dikendalikan untuk menjaga proses berjalan sesuai dengan kondisi yang diinginkan. Di antaranya yang paling umum adalah tekanan (*pressure*) pada 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.

Dalam praktiknya, sistem pengendalian otomatis memiliki peran penting karena dapat meningkatkan kinerja sistem dinamis, memperbaiki kualitas produk, menurunkan biaya produksi, dan meningkatkan kecepatan produksi.

Peran operator manual kini sebagian besar telah digantikan oleh alat yang disebut *controller*. Misalnya, tugas membuka dan menutup katup (*valve*) dapat dilakukan secara otomatis menggunakan *controller*. Agar dapat dikendalikan secara otomatis, katup dilengkapi dengan *actuator* sehingga berfungsi sebagai *control valve* dalam sistem instrumentasi pengendalian proses.

Salah satu metode pengendalian yang umum digunakan adalah pengendalian PID (*Proportional Integral Derivative*). Pengendalian PID berfungsi untuk memperbaiki performa sistem melalui peningkatan respons terhadap perubahan kondisi. Dengan penerapan kontrol PID, sistem dapat mengurangi bahkan menghilangkan kesalahan pada kondisi tunak (*steady-state error*) serta meningkatkan kecepatan respons dalam mencapai nilai *set point* yang diinginkan.

2.2 Pengendalian *Proportional* (P)

Pengendalian proporsional 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

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

Berdasarkan pendekatan tersebut, K_p adalah *proportional gain* yang menunjukkan *responsiveness controller* terhadap proses *setting up* atau koreksi yang diberikan untuk mencapai *set point*. *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 *proportional*. Daerah kerja efektif *controller* dicerminkan oleh *proportional band*, sedangkan konstanta proporsional menunjukkan nilai faktor penguatan terhadap sinyal kesalahan, K_p . Hubungan antara *proportional band* (PB) dengan konstanta *proportional* (K_p) ditunjukkan sebagai berikut:

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

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 *steady state error* yang besar.

Ciri-ciri pengontrol proporsional:

1. Jika nilai K_p kecil, pengontrol proporsional memberikan koreksi *error* yang kecil sehingga respons sistem menjadi lebih lambat dan *rise time* meningkat.
2. Jika nilai K_p diperbesar, pengontrol proporsional memberikan koreksi yang lebih besar sehingga respons sistem menjadi lebih cepat dan *rise time* menurun.
3. Jika nilai K_p terlalu besar, sistem menjadi tidak stabil, ditandai dengan terjadinya osilasi atau *overshoot*.
4. Nilai K_p dapat diatur untuk mengurangi *steady state error*, tetapi tidak dapat menghilangkan sepenuhnya.

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.

$$p(t) = \bar{p} + K_i \int_0^t \text{error}(\tau) d\tau \quad (2.3)$$

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 *control valve* yang sudah dalam kondisi jenuh (terbuka penuh), sehingga sistem tidak lagi merespons perintah kendali. Hal ini dapat menyebabkan *overshoot* atau bahkan membuat *controller* kehilangan fungsinya sebagai pengendali. Untuk mengatasi hal ini, biasanya digunakan metode tambahan seperti *anti-reset windup* agar respon sistem tetap stabil dan terkendali.

2.4 Pengendalian Derivative (D)

Pengendalian *Derivative* berfungsi untuk mengantisipasi 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 respons sistem sehingga akan mengurangi *overshoot* (sinyal melebihi *set point*).

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

Pengendali *derivative* 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 mengintegrasikan *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

$$p(t) = \bar{p} + K_p(\text{error}(t) + \frac{1}{\tau_i} \int_0^t \text{error}(\tau) d\tau) \quad (2.6)$$

K_i = *integral gain* atau "reset rate" (*repeat/minute*)

K_p = besar koreksi yang diberikan

τ_i = waktu integral atau waktu *reset*

Pengendalian integral dikenal pula sebagai "*reset*" yang mempunyai respons 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_c , τ_i dan τ_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:

$$\Delta Q_{output} = K_d \frac{d \text{error}(t)}{dt} \quad (2.5)$$

Atau

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

K_d = time constant

τ_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_p , τ_i dan τ_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 *input*nya, lalu dari pengamatan itu kita menentukan nilai-nilai parameter PID yang tepat. Tujuannya adalah agar sistem dapat memberikan respons yang cepat dan tetap stabil, dengan nilai *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 respons cepat dan efisien.

Metode Ziegler-Nichols untuk 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 (τ_i) dan diferensial (τ_d) *gain* berdasarkan rumus yang ditentukan oleh Ziegler-Nichols. Umumnya, rumus untuk menetapkan parameter PID adalah sebagai berikut:

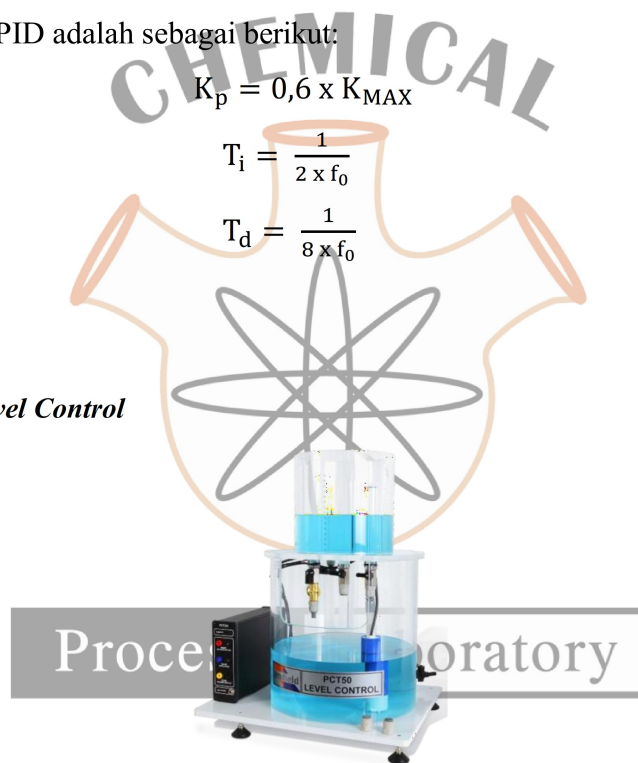
$$K_p = 0,6 \times K_{MAX} \quad (2.9)$$

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

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

(Ellis, 2012)

2.8 PCT 50 Level Control



Gambar 2.1 PCT 50 level control

PCT 50 merupakan alat *level control* yang menggunakan air sebagai fluida kerja. Air dari tangki penampung bawah dialirkan menuju bejana proses atas (*upper process vessel*) menggunakan pompa sentrifugal dengan kecepatan yang dapat diatur. Pada saluran keluaran pompa terdapat konektor pelepas cepat (*quick-release connector*) yang memungkinkan selang dilepas dan dipasang dengan mudah. Konektor ini digunakan untuk memudahkan proses *priming*, yaitu pengisian air ke dalam pompa dan salurannya sebelum pompa dioperasikan. Saluran masuk pada bejana proses dirancang secara vertikal sehingga aliran air yang masuk tetap dapat diamati pada berbagai ketinggian air. Selain itu, terdapat *check valve* yang berfungsi mencegah air mengalir kembali

ke tangki penampung bawah ketika pompa diperlambat atau dihentikan. Katup bola (CV1) digunakan untuk mengatur besar aliran air yang masuk ke bejana proses secara manual, sehingga kondisi aliran dapat disesuaikan dengan kebutuhan percobaan.

Ketinggian air (*level*) di dalam bejana proses diukur menggunakan sensor tekanan elektronik yang dipasang pada bagian bawah atau sisi bejana. Salah satu sisi sensor terhubung dengan bagian dalam bejana, sedangkan sisi lainnya terbuka terhadap atmosfer. Perbedaan tekanan antara air di dalam bejana dan tekanan atmosfer digunakan untuk menentukan ketinggian air. Selain menggunakan sensor tekanan, ketinggian air juga dapat diamati melalui skala yang terdapat pada sisi bejana proses. Setelah berada di dalam bejana proses, air mengalir kembali menuju tangki penampung bawah melalui dua saluran keluar (*outlet*) yang terdapat di bagian dasar bejana. Salah satu *outlet* mengalirkan air secara kontinu, sedangkan aliran pada *outlet* lainnya dapat diatur menggunakan katup solenoida (*solenoid valve/SOL*) yang dikendalikan dari jarak jauh. Kedua *outlet* dilengkapi dengan katup bola (CV2 dan CV3) untuk mengatur besar aliran secara manual.

Selain katup bola, kedua saluran keluar dilengkapi *orifice* yang dapat digunakan untuk menetapkan besar aliran pada kondisi tertentu. *Orifice* merupakan lubang dengan ukuran tertentu yang berfungsi membatasi aliran air. Ukuran *orifice* dapat diganti sesuai kebutuhan dengan melepas penutup plastik dan menggantinya dengan ukuran *orifice* yang diinginkan. Penggantian dilakukan secara manual menggunakan segel cincin *O-ring* dan cukup dikencangkan dengan tangan. *Orifice* dengan ukuran lain disimpan pada bagian depan pelat dasar alat ketika tidak digunakan.

Bejana proses juga dilengkapi saluran *overflow* yang berfungsi mengalirkan kelebihan air kembali ke tangki penampung bawah sehingga mencegah bejana proses terisi secara berlebihan. Sensor tekanan, pompa sentrifugal, dan katup solenoida terhubung dengan *electrical interface* yang berfungsi mengolah sinyal dari setiap komponen agar sistem dapat dioperasikan melalui komputer menggunakan satu koneksi USB. Perangkat lunak PCT 50 dapat digunakan untuk mengendalikan *level* secara langsung sekaligus merekam respons sistem selama percobaan. Selain itu, perangkat lunak dapat digunakan untuk merekam data ketika sistem dioperasikan menggunakan pengontrol PID. Dengan demikian, PCT 50 dapat digunakan untuk mempelajari karakteristik dan respons sistem *level control* terhadap perubahan kondisi operasi maupun parameter pengendalian.

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

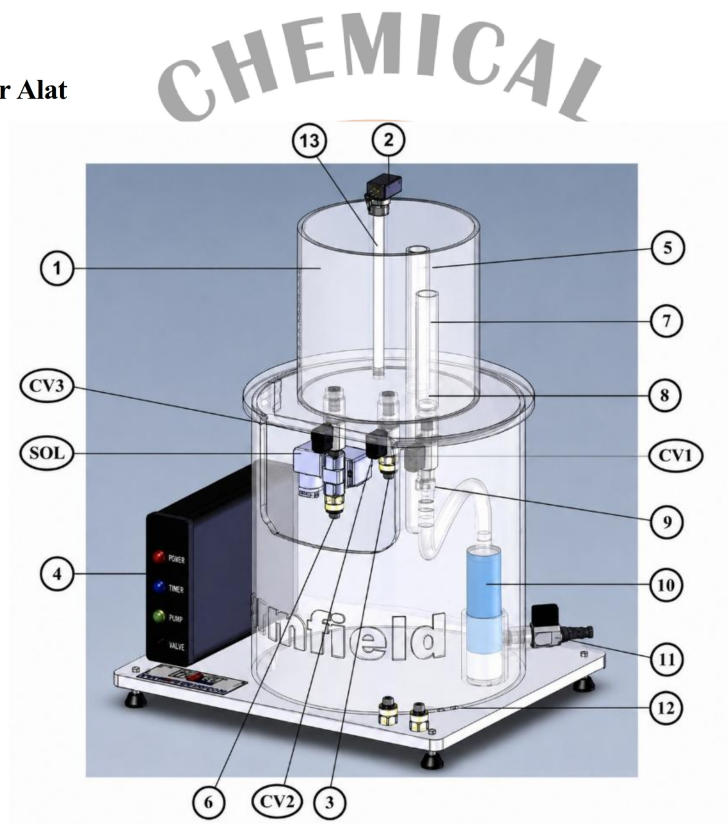
3.2.1 Bahan

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 Connector</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. *Install 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 *install* dengan 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.

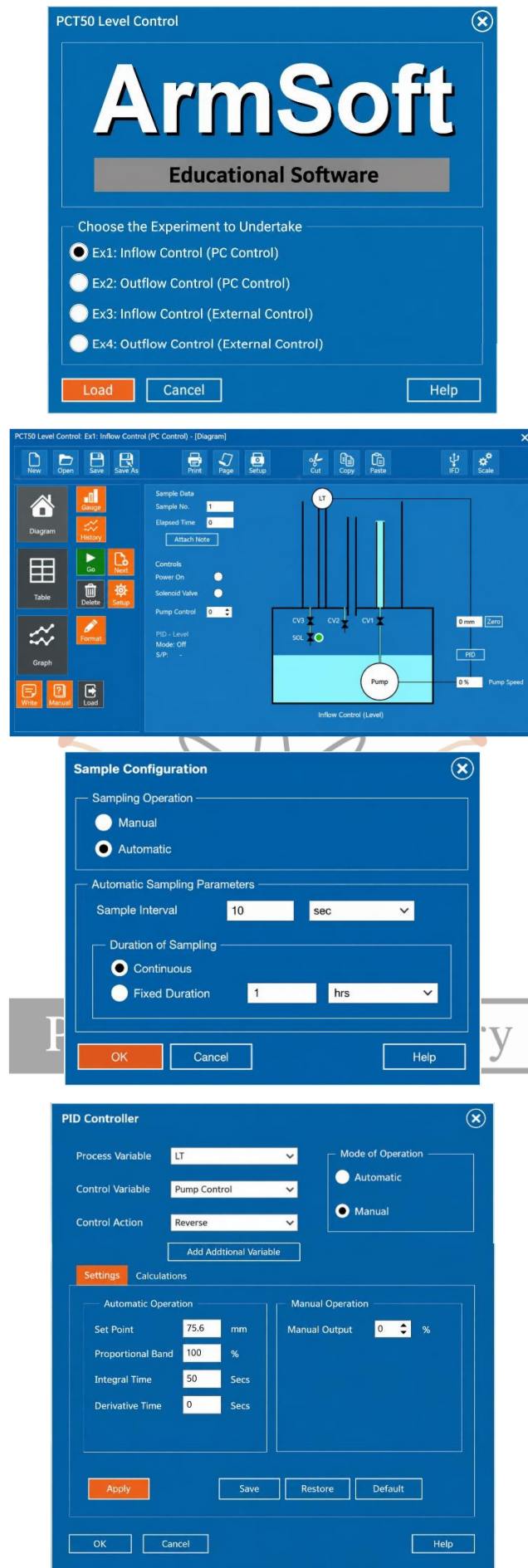
b. Mencari *level* maksimum

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 ditentukan pada menu PID.
4. Operasikan alat sesuai dengan variabel yang ditentukan dengan mengklik *power on* serta *go* untuk *record* 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 dan pengaruh parameter PID

1. Memastikan keadaan tangki proses kosong.
2. Mengatur variabel (*pump speed*, *set point*, *proportional band*, *integral time*, *derivative time*, dan *mode of operation*) 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 *record* data yang didapat.
5. Amati grafik, ambil dan simpan data hasil praktikum sesuai dengan data yang tertera pada aplikasi PCT-50 (excel).
6. Lakukan langkah 2–6 dengan memvariasikan salah satu parameter PID (*proportional band*, *integral time*, atau *derivative time*) sesuai dengan variabel yang ditentukan, sementara parameter PID lainnya dipertahankan.
7. 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.
8. Setelah dibersihkan dan dikeringkan, alat dapat dimatikan

secara bertahap.



Gambar 3.3 Langkah-langkah penggunaan software Armfield PCT 50