

**TUGAS AKHIR - EE184801**

# **DESAIN KONTROLER CASCADE PID GENETIC ALGORITHM UNTUK PENGATURAN LEVEL-FLOW TANGKI PCT-100**

**HAKHI GYA YEKTianto**

**NRP 07111940000022**

Dosen Pembimbing

**Dr. Trihastuti Agustinah, S.T., M.T.**

**NIP 196808121994032001**

**Eka Iskandar, ST., MT.**

**NIP 196808121994032001**

**Program Studi Sarjana Teknik Elektro**

Departemen Teknik Elektro

Fakultas Teknik Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya

Tahun 2023



**TUGAS AKHIR - EE184801**

# **DESAIN KONTROLER CASCADE PID GENETIC ALGORITHM UNTUK PENGATURAN LEVEL-FLOW TANGKI PCT-100**

**HAKHI GYA YEKTianto**

**NRP 07111940000022**

**Dosen Pembimbing**

**Dr. Trihastuti Agustinah, S.T., M.T.**

**NIP 196808121994032001**

**Eka Iskandar, S.T., M.T.**

**NIP 198005282008121001**

**Program Studi Sarjana Teknik Elektro**

**Departemen Teknik Elektro**

**Fakultas Teknik Elektro dan Informatika Cerdas**

**Institut Teknologi Sepuluh Nopember**

**Surabaya**

**Tahun 2023**



**FINAL PROJECT - EE184801**

# **CASCADE PID CONTROLER DESIGN USING GENETIC ALGORITHM FOR LEVEL-FLOW CONTROLLER OF PCT-100 TANK**

**HAKHI GYA YEKTianto**

**NRP 07111940000022**

Advisor

**Dr. Trihastuti Agustinah, S.T., M.T.**

**NIP 196808121994032001**

**Eka Iskandar, ST., MT.**

**NIP 198005282008121001**

**Study Program Bahelor of Electrical Engineering**

Department of Electrical Engineering

Faculty of Intelligent Electrical and Information Technology

Institut Teknologi Sepuluh Nopember

Surabaya

2023



## LEMBAR PENGESAHAN

### DESAIN KONTROLER CASCADE PID GENETIC ALGORITHM UNTUK PENGATURAN LEVEL-FLOW TANGKI PCT-100

#### TUGAS AKHIR


Diajukan untuk memenuhi salah satu syarat  
memperoleh gelar Sarjana Teknik pada  
Program Studi S-1 Teknik Sistem Pengaturan  
Departemen Teknik Elektro  
Fakultas Teknologi Elektro dan Informatika Cerdas  
Institut Teknologi Sepuluh Nopember

Oleh : **HAKHI GYA YEKTianto**


NRP. 07111940000022

Disetujui oleh Tim Penguji Tugas Akhir :

1. Dr. Trihastuti Agustinah, S.T., M.T

Pembimbing 

2. Eka Iskandar, ST., MT

Ko-pembimbing 

3. Ir. Ali Fatoni, MT.

Penguji 

4. Mohamad Abdul Hady, S.T., M.T.

Penguji 

5. Prof.Dr.Ir. Achmad Jazidie, M.Eng.

Penguji 

**SURABAYA**

**Juli, 2023**

*(Halaman ini sengaja dikosongkan)*

## PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini:

Nama mahasiswa / NRP : Hakhi Gya Yektianto / 07111940000022  
Program studi : Teknik Sistem Pengaturan  
Dosen Pembimbing / NIP : Dr. Trihastuti Agustinah, S.T., M.T / 196808121994032001  
Eka Iskandar, ST., MT /. 198005282008121001

dengan ini menyatakan bahwa Tugas Akhir dengan judul “Desain Kontroler Cascade PID Menggunakan Genetic Algorithm Untuk Pengaturan Level-Flow Tangki PCT-100” adalah hasil karya sendiri, bersifat orisinal, dan ditulis dengan mengikuti kaidah penulisan ilmiah.

Bilamana di kemudian hari ditemukan ketidaksesuaian dengan pernyataan ini, maka saya bersedia menerima sanksi sesuai dengan ketentuan yang berlaku di Institut Teknologi Sepuluh Nopember.

Surabaya, 17 Juli 2023

Mengetahui  
Dosen Pembimbing



Dr. Trihastuti Agustinah, S.T., M.T  
NIP. 196808121994032001

Mengetahui  
Dosen Ko-pembimbing



Eka Iskandar, ST., MT  
NIP. 198005282008121001

Mahasiswa



Hakhi Gya Yektianto  
NRP. 07111940000022

*(Halaman ini sengaja dikosongkan)*



## ABSTRAK

### DESAIN KONTROLER CASCADE PID GENETIC ALGORITHM UNTUK PENGATURAN LEVEL-FLOW TANGKI PCT-100

Nama Mahasiswa / NRP : Hakhi Gya Yektianto / 07111940000022  
Departemen : Teknik Elektro FTEIC - ITS  
Dosen Pembimbing : Dr. Trihastuti Agustinah, S.T., M.T  
Eka Iskandar, ST., MT

#### Abstrak

Dalam industri banyak yang menggunakan fluida (cairan) sebagai bahan pada proses pengolahannya. Pada proses yang berhubungan fluida sering dijumpai sistem pengaturan *level* dan *flow* untuk fluida. Dalam pengaturan *level*, untuk mendapatkan kinerja yang optimal banyak proses industri yang menggunakan PID kontroler. Akan tetapi kontroler PID konvensional kurang mampu menangani perubahan pada variabel manipulasi yaitu berupa gangguan debit *input* pada sistem pengaturan *level*. Sehingga dibutuhkan konfigurasi *cascade flow* dan *level* untuk menangani permasalahan tersebut. Sistem pengaturan *cascade* pada umumnya masih menggunakan kontroler PID untuk mengatur *flow* dan *level*. Untuk dapat mendapatkan parameter PID pada konfigurasi *cascade* diperlukan perhitungan yang rumit jika menggunakan metode *tuning* konvensional. Selain itu hasil *tuning* secara konvensional belum tentu menghasilkan performa yang diinginkan karena sistem *cascade* yang kompleks. Oleh karena itu digunakan metode pengaturan PID berbasis *Genetic Algorithm* (GA) untuk melakukan optimisasi tuning parameter kontroler PID *cascade*. Kontroler *cascade* PID-GA mampu memenuhi spesifikasi *level* air yang diinginkan, yaitu *overshoot* < 5%, *error steady state* 0% dan *settling time* < 120 detik. Selain itu sistem dengan konfigurasi *cascade* memiliki dampak gangguan lebih kecil daripada sistem tanpa konfigurasi *cascade*.

**Kata kunci:** *Cascade, Flow, Genetic Algorithm, Level, PID.*

*(Halaman ini sengaja dikosongkan)*

## **ABSTRACT**

### **CASCADE PID CONTROLER DESIGN USING GENETIC ALGORITHM FOR LEVEL-FLOW CONTROLLER OF PCT-100 TANK**

**Student Name / NRP** : Hakhi Gya Yektianto / 07111940000022  
**Department** : Teknik Elektro FTEIC - ITS  
**Advisor** : Dr. Trihastuti Agustinah, S.T., M.T  
Eka Iskandar, ST., MT

#### **Abstract**

In industry, many use fluid (liquid) as an ingredient in the processing process. In fluid-related processes, level and flow control systems are often found for fluids. In the control level, to get optimal performance, many industrial processes use PID controllers. However, conventional PID controllers are less able to handle changes in the manipulated variables, namely in the form of input discharge disturbances in the level setting system. So that cascade flow and level configurations are needed to handle this problem. Cascade control systems generally still use a PID controller to control flow and level. To get the PID parameters in the cascade configuration, complicated calculations are needed when using conventional tuning methods. In addition, conventional tuning results may not necessarily get the desired performance because of the complexity cascade system. Therefore, a Genetic Algorithm (GA) based PID adjustment method is used to optimize the cascade PID controller parameter tuning. The PID-GA cascade controller can meet the desired water level specifications, namely 5% overshoot, 0% steady state error and settling time < 150 seconds. In addition, a system with a cascade configuration has a better response than a system without a cascade configuration.

**Keywords:** *Cascade, Flow, Genetic Algorithm, Level, PID*

*(Halaman ini sengaja dikosongkan)*

## KATA PENGANTAR

Puji dan syukur kami panjatkan kehadiran Allah SWT, karena berkat rahmat dan karunia-Nya, penulis dapat menyelesaikan penyusunan tugas akhir dengan judul “**Desain Kontroler Cascade PID Menggunakan Genetic Algorithm Untuk Pengaturan Level-Flow Tangki PCT-100**” untuk memenuhi syarat kelulusan pada Bidang Studi Teknik Sistem Pengaturan, Departemen Teknik Elektro, Fakultas Teknik Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember (ITS). Laporan tugas akhir ini dapat diselesaikan oleh penulis berkat bantuan, bimbingan, dan dukungan dari berbagai pihak yang membantu penulis selama pengerjaan yang perlu penulis ucapkan terimakasih kepada:

1. Kedua orang tua penulis beserta adik penulis yang selalu menyemangati penulis selama pengerjaan tugas akhir, serta memberikan doa kepada penulis.
2. Ibu Dr. Trihastuti Agustinah, S.T., M.T dan Bapak Eka Iskandar, ST., MT selaku dosen pembimbing tugas akhir yang selalu membantu penulis dengan memberikan bimbingan dan arahan selama pengerjaan tugas akhir.
3. Prof.Ir. Abdullah Alkaf, Ph.D. selaku dosen wali penulis selama berkuliah di departemen teknik elektro yang telah membantu proses akademik penulis selama masa perkuliahan.
4. Bapak dan ibu dosen bidang studi teknik sistem pengaturan serta bapak dan ibu dosen teknik elektro ITS yang telah memberikan materi perkuliahan yang berguna dalam penyusunan tugas akhir.
5. Teman-teman PCT-100, Muhammad Faris Zuhairi, Muhammad Usama Sahal, dan Reno Dari Khalifah Panarung atas bantuan dan kerja samanya selama mengerjakan tugas akhir PCT bertsama, sehingga penulis dapat menyelesaikan tugas akhir ini.
6. Keluarga besar asistem Laboratorium Kontrol dan Otomasi dan asistem laboratorium Sistem dan Sibernetika yang telah membantu, memberi dukungan, dan memfasilitasi penelitian tugas akhir ini.
7. Semua pihak yang telah membantu dan tidak dapat penulis sebutkan satu persatu

Penulis berharap laporan ini dapat bermanfaat bagi pembaca serta penulis pada khususnya. Dalam penyusunan tugas akhir ini, penulis sadar bahwa hasil yang telah dikerjakan masih belum sempurna dan masih perlu diperbaiki, sehingga penulis mengharapkan kritik dan saran dari pembaca yang bersifat membangun.

Surabaya, Juli 2023

Pemulis

*(Halaman ini sengaja dikosongkan)*

## DAFTAR ISI

LEMBAR PENGESAHAN.....	i
PERNYATAAN ORISINALITAS .....	ii
ABSTRAK.....	iv
ABSTRACT.....	vii
KATA PENGANTAR .....	ix
DAFTAR ISI.....	xi
DAFTAR GAMBAR .....	xiii
DAFTAR TABEL.....	xiv
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	1
1.3 Batasan Masalah .....	1
1.4 Tujuan.....	2
1.5 Manfaat.....	2
BAB 2 TINJAUAN PUSTAKA .....	3
2.1 Hasil Penelitian Terdahulu .....	3
2.2 Dasar Teori .....	3
2.2.1 Pengenalan <i>Plant</i> PCT 100 .....	3
2.2.2 Sistem Pengaturan Cascade.....	5
2.2.3 Sistem Pengaturan Cascade <i>Level-Flow</i> Tangki PCT-100 .....	5
2.2.4 Kontroler PID.....	8
2.2.5 Identifikasi Sistem dan Aturan Kontrol PID Ziegler Nichols.....	9
2.2.6 Genetic Algorihm (GA) .....	10
2.2.7 Kontroler PID-GA.....	11
2.2.8 Indeks Performa .....	12
BAB 3 PERANCANGAN SISTEM.....	13
3.1 Metode yang digunakan.....	13
3.2 Bahan dan peralatan yang digunakan .....	13
3.2.1 Modul Process Control Technology 100.....	14
3.2.2 STM32F407VG.....	14
3.2.3 Rangkaian Modul Mikrokontroler .....	15
3.2.4 MATLAB dan Simulink 2020b.....	18
3.3 Urutan pelaksanaan penelitian.....	18

3.3.1	Pemodelan Level Transmitter.....	18
3.3.2	Pemodelan Flow Transmitter.....	19
3.3.3	Pemodelan Sistem Pengaturan Flow .....	21
3.3.4	Pemodelan Sistem Pengaturan Level.....	23
3.3.5	Desain kontroler PID Ziegler-Nichols.....	24
3.3.6	Desain Kontroler PID-GA .....	26
BAB 4	Hasil dan Pembahasan .....	29
4.1	Hasil penelitian .....	29
4.1.1	Hasil pengujian Model Flow .....	29
4.1.2	Hasil Pengujian Respon Sistem Cascade tanpa Kontroler .....	30
4.1.3	Hasil Pengujian Respon Sistem dengan Kontroler PID Ziegler Nichols ...	30
4.1.4	Hasil Pengujian Respon Sistem dengan Kontroler PID GA.....	31
4.1.5	Hasil Pengujian Pada Disturbance Flow.....	35
4.1.6	Hasil Pengujian pada Pembebanan Flow.....	36
BAB 5	Kesimpulan dan Saran .....	39
5.1	Kesimpulan .....	39
5.2	Saran.....	39
DAFTAR PUSTAKA .....		40
LAMPIRAN.....		41
BIODATA PENULIS .....		47



## DAFTAR GAMBAR

Gambar 2.1 Tampilan PCT-100.....	4
Gambar 2.2 Skema Pengaturan cascade (mathworks, 2023).....	5
Gambar 2.3 Skema Proses Pengaturan Cascade Level pada Tangki.....	6
Gambar 2.4 Ilustrasi Sistem Pengaturan Level pada Tangki PCT-100 .....	6
Gambar 2.5 Ilustrasi Pemodelan Sistem Pengaturan Level Tangki (Ogata, 2010).....	7
Gambar 2.6 Respon sistem yang menyerupai huruf S (Ogata, 2010).....	9
Gambar 2.7 Flowchart Genetic Algorithm .....	10
Gambar 2.8 Diagram Blok Sistem pengaturan PID-GA.....	11
Gambar 3.1 Flowchart diagram penelitian.....	13
Gambar 3.2 Module Control PCT-100 .....	14
Gambar 3.3 Mikrokontroler STM32F407VG.....	15
Gambar 3.4 Rangkaian Power Supply .....	15
Gambar 3.5 Rangkaian Modul ADC .....	16
Gambar 3.6 Rangkaian Modul DAC .....	16
Gambar 3.7 Rangkaian Amplifier dan Regulator .....	17
Gambar 3.8 Rangkaian Mikrokontroler dan Data Logger.....	17
Gambar 3.9 Respon tegangan level transmitter terhadap ketinggian air .....	18
Gambar 3.10 Gambar Tegangan Flow Meter terhadap Variasi Input Pompa .....	19
Gambar 3.11 Laju perubahan ketinggian level terhadap variasi input pompa.....	20
Gambar 3.12 Perbandingan tegangan $V_{FT}$ terhadap perubahan ketinggian level tangki.....	20
Gambar 3.13 Hasil Pengujian Step Respon Model Flow .....	22
Gambar 3.14 Penentuan Titik Inflection Point .....	22
Gambar 3.15 Diagram Blok Konfigurasi cascade .....	24
Gambar 3.16 Penyederhanaan inner loop cascade.....	25
Gambar 3.17 Step Response Gout .....	25
Gambar 3.18 Diagram Blok Sistem pengaturan PID-GA.....	27
Gambar 4.1 Hasil Validasi Model dan Step Respon Sistem.....	29
Gambar 4.2 Step Respon Sistem Tanpa Kontroler .....	30
Gambar 4.3 Step Respon Sistem Kontroler PID Ziegler Nichols.....	31
Gambar 4.4 Step Respon Sistem Kontroler PID GA .....	32
Gambar 4.5 Respon Sistem kontroler PID GA dengan Variasi Populasi .....	32
Gambar 4.6 Respon Sistem kontroler PID GA dengan Variasi Generasi.....	33
Gambar 4.7 Respon Sistem kontroler PID GA dengan Variasi Probabilitas Mutasi.....	34
Gambar 4.8 Respon Sistem terhadap Disturbance pada Flow .....	35
Gambar 4.9 Respon Sistem terhadap Pembebanan pada Flow.....	36
Gambar 4.10 Perbandingan Respon Pembebanan pada PID ZN dan PID GA.....	37

## DAFTAR TABEL

Tabel 2.1 Bagian Process Rig PCT 100 .....	4
Tabel 2.2 Penjelasan Fault Switch Control Module PCT .....	5
Tabel 3.1 Parameter Fisis Tangki PCT-100 .....	23
Tabel 3.2 Parameter PID inner loop .....	24
Tabel 4.1 Parameter Kontroler PID Cascade .....	30
Tabel 4.2 Nilai ISE kontroler PID ZN.....	30
Tabel 4.3 Nilai ISE kontroler PID GA dengan Variasi Populasi .....	32
Tabel 4.4 Nilai ISE kontroler PID GA dengan Variasi Generasi.....	33
Tabel 4.5 Nilai ISE kontroler PID GA dengan Variasi Probabilitas Mutasi.....	34

# BAB 1 PENDAHULUAN

## 1.1 Latar Belakang

Dalam industri banyak yang menggunakan fluida (cairan) sebagai bahan pada proses pengolahannya. Pada proses yang berhubungan fluida sering dijumpai sistem pengaturan *level* dan *flow* untuk fluida (Raharjo, 2014). Sistem pengaturan *level* fluida merupakan salah satu aplikasi sistem kontrol pada proses industri. Fluida dipompa dan ditampung dalam tangki/bejana khusus yang memerlukan pengaturan batas *level* yang telah ditentukan. Akan tetapi sering terjadi gangguan pada instrumen pengukur dan gangguan tak terduga. Hal ini tentu saja mempengaruhi operasi sistem kontrol *level* dan aliran cairan. Untuk mencapai kinerja yang baik dalam sistem pengaturan level dan aliran, diperlukan pengontrol yang mampu menangani masalah yang muncul dalam sistem, yaitu kontroler PID (Pritandi, 2016).

Dalam pengaturan *level*, untuk mendapatkan kinerja yang optimal banyak proses industri yang menggunakan PID kontroler, kontroler PID dipilih karena bentuknya yang sederhana dan mudah dalam pengimplementasian. Akan tetapi kontroler PID konvensional kurang mampu menangani perubahan pada variabel manipulasi yaitu berupa gangguan debit *input* pada sistem pengaturan *level*. Sehingga dibutuhkan konfigurasi *cascade flow* dan *level* untuk menangani permasalahan tersebut (Tridianto et al., 2017). Sistem pengaturan *cascade* pada umumnya masih menggunakan kontroler PID untuk mengatur *flow* dan *level*. Untuk dapat mendapatkan parameter PID pada konfigurasi *cascade* diperlukan perhitungan yang rumit jika menggunakan metode *tuning* konvensional. Selain itu hasil *tuning* secara konvensional belum tentu menghasilkan performa yang diinginkan karena sistem *cascade* yang kompleks.

Penelitian tentang metode pengaturan untuk meningkatkan performa kontroler PID banyak dilakukan oleh peneliti. Salah satunya adalah metode pengaturan PID berbasis *Genetic Algorithm* (GA) untuk melakukan optimisasi tuning parameter kontroler PID konvensional. Salah satu penelitian tentang penentuan parameter PID menggunakan *Genetic Algorithm* adalah penentuan parameter PID pada *steam drum* PLTU menggunakan *Genetic Algorithm* (Yusuf et al., 2017). Oleh karena itu diperlukan cara tuning PID *cascade* menggunakan *Genetic Algorithm* untuk mendapatkan performa kontroler PID *cascade* yang lebih baik

## 1.2 Rumusan Masalah

Pada pengaturan proses industri banyak dijumpai sistem pengaturan *level* pada tangki. salah satu permasalahan pada sistem pengaturan *level* adalah adanya gangguan pada *manipulated variabel* yaitu gangguan pada debit *input*. Penggunaan PID konvensional dirasa kurang mampu untuk mengatasi permasalahan tersebut. Salah satu metode untuk memperbaiki respon sistem terhadap gangguan adalah menggunakan konfigurasi PID *cascade*. akan tetapi proses tuning PID *cascade* menjadi rumit karena perlu melakukan *tuning* pada dua kontroler sekaligus. sehingga penulis berencana menggunakan *Genetic Algorithm* dalam penentuan parameter PID pada konfigurasi *cascade*.

## 1.3 Batasan Masalah

Dalam melaksanakan Tugas Akhir ini, terdapat berbagai batasan masalah yang perlu diperhatikan, diantaranya sebagai berikut:

1. *Plant* yang digunakan pada Tugas Akhir ini adalah PCT 100 dengan sistem pengaturan level yang selanjutnya dimodelkan persamaan model linear fungsi alih casacade antara tangki dan *control valve*.
2. Ruang lingkup yang dibahas yaitu permasalahan respon sistem ketika terjadi disturbance pada *flow*.
3. Optimasi parameter PID dilakukan secara *offline* menggunakan Algoritma *Genetic Algorithm*

#### 1.4 Tujuan

Tujuan dari tugas akhir ini adalah:

1. Mengetahui perbandingan sistem dengan konfigurasi cascade dan tanpa cascade dalam menangani gangguan
2. Mengetahui respon sistem setelah *detuning* dengan PID kontroler menggunakan metode *genetic algorithm*.

#### 1.5 Manfaat

Manfaat penelitian ini diharapkan Tuning PID berbasis *Genetic Algorithm* pada sistem pengaturan *cascade flow* dan *level* dapat menjadi solusi pada permasalahan Tuning PID denag konfigurasi *cascade* dan dapat diaplikasikan dalam industri kontrol proses, serta sebagai referensi untuk penelitian selanjutnya

## BAB 2 TINJAUAN PUSTAKA

### 2.1 Hasil Penelitian Terdahulu

Penelitian mengenai *plant* PCT-100 oleh Adetya Devritama (2016) dilakukan dengan metode PID gain Scheduling untuk mengontrol *level* proses. Identifikasi system dilakukan dengan metode Harriot dan divalidasi menggunakan ISE/Integral Square Error. Hasil yang diperoleh yakni controller PID gain scheduling dapat mengendalikan respon dinamis dalam simulasi dengan rata-rata *error* waktu tunak 0.06% dan 0.12% pada implementasi. Namun, nilai gain PID masih dipengaruhi variable tertentu yang berubah kontinu atau dengan kata lain gain scheduling hanya cocok untuk variasi yang terprediksi.(Devritama, 2016).

Penelitian *level* proses kemudian dilanjutkan oleh penelitian M. Zakki Ghuftron (2016) yang menggunakan self-tuning PID dengan pendekatan adaptif. Controller adaptif dapat memperbaiki kelemahan metode gain scheduling, dimana dapat bekerja pada variasi tak terprediksi. Nilai gamma memengaruhi kecepatan adaptasi controller self-tuning PID. Kriteria square time-square error/STSE unggul karena memberikan *error* terkecil 3.64% saat simulasi dan 0.093% bila diberikan pembebanan. Namun, implementasinya masih memerlukan waktu learning yang lama dan memerlukan otomatisasi penentuan nilai gamma (Ghuftron, 2016).

Penelitian plant PCT-100 dilanjutkan oleh Khoirus Fauzi R. (2022) dilakukan dengan pengaturan level dan flow menggunakan metode cascade. Metode yang digunakan adalah tuning PID dengan algoritma Particel Swamp Optimization (PSO). Pada *Inner Loop* diperoleh besaran  $K_p$ ,  $K_i$ , dan hasil tuning PSO sebesar 20.8441, 10.0045, dan 5.8969 dengan indeks performansi ITAE sebesar 0.6534 dan ISE 0.2022. Memiliki Karakteristik sistem *max overshoot* 0.567%, *Rise time* 0.1544 detik dan *Settling time* 0.2473 detik. Pada *Outer Loop* diperoleh besaran  $K_p$ ,  $K_i$ , dan  $K_d$  yang dituning dengan PSO sebesar 68.4639, 1.8237, dan 10.2855 memiliki indeks performansi error ITAE sebesar 1.51 dan ISE sebesar 0.5841(Rahmadhani, 2022).

Penelitian tentang penggunaan Genetic Algorithm pernah dilakukan oleh Mohamad Yusuf (2017). Metode Genetic Algorithm digunakan untuk autotuning PID pada pengaturan level air tangki *steamdrum* pada PLTU. Hasil dari kontroler PID-GA mampu memenuhi spesifikasi desain yang diinginkan. Selain itu juga dapat meredam gangguan berupa perubahan beban dengan *perturbation peak* kurang dari 0.25(Yusuf et al., 2017)

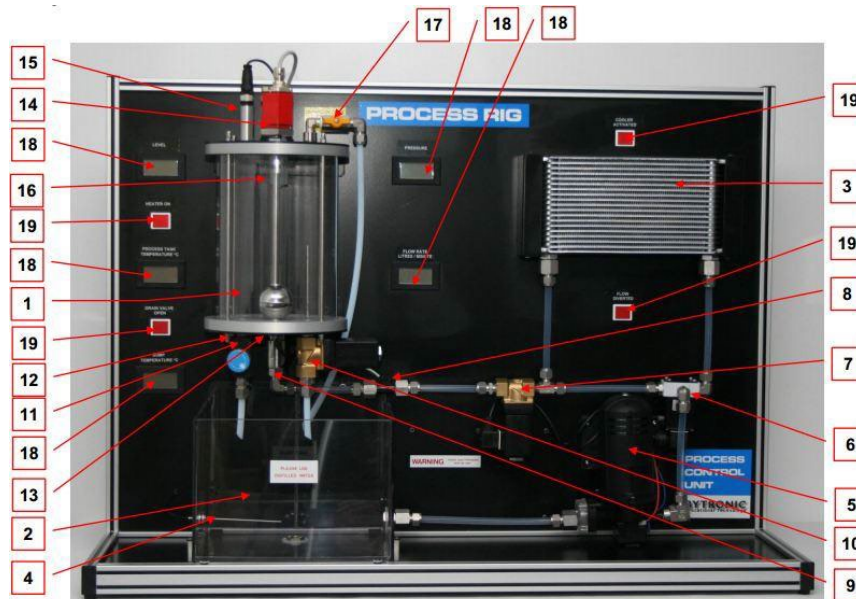
### 2.2 Dasar Teori

#### 2.2.1 Pengenalan *Plant* PCT 100

PCT 100 merupakan sebuah model sistem yang mengimplementasikan proses kontinyu pada fluida yang merepresentasikan proses yang terjadi di industri. Pada PCT 100, terdapat komponen-komponen, baik berupa hardware maupun software yang berfungsi untuk melakukan pengaturan pada proses yang terjadi di fluida, diantaranya pengaturan *level*, *flow*, *temperatur*, dan tekanan pada fluida. PCT 100 terdiri dari dua komponen, yaitu *Process rig* dan control module. Secara umum *Process rig* merupakan tempat berlangsungnya proses, sedangkan control module berfungsi untuk melakukan pengaturan pada sistem.

*Process rig* pada PCT 100 berfungsi sebagai tempat berlangsungnya proses pengaturan fluida. Terdapat beberapa komponen pada *Process rig* mulai dari pompa

untuk mensirkulasikan air, hingga heater dan fan untuk mengatur temperatur air. Tampilan dari *Process rig* ditampilkan pada Gambar 2.1.



Gambar 2.1 Tampilan PCT-100

*Process rig* pada PCT 100 terdiri dari komponen-komponen yang berfungsi sebagai tempat berlangsungnya pengaturan *level*, *flow*, *temperature*, dan *pressure*. Pada Gambar 2.1, terdapat Sembilan belas komponen yang mempunyai fungsi masing-masing. Komponen utama yang terdapat pada Process Rig, seperti pompa, *control valve*, pemanas, dan sebagai aktuator, dan beberapa jenis transmitter. Bagian-bagian dari *Process rig* yang terdapat pada Gambar 2.3 ditampilkan pada Tabel 2.1.

Tabel 2.1 Bagian Process Rig PCT 100

No	Keterangan
1	<i>Process Tank</i>
2	<i>Sump Tank</i>
3	<i>Cooler Unit</i>
4	<i>Thermocouple Sump Tank</i>
5	<i>Pump</i>
6	<i>3/2 Diverter Valve</i>
7	<i>2/2 Proportional Drain Valve</i>
8	<i>Transmitter Flow</i>
9	<i>One way Check valve</i>
10	<i>2/2 Proportional drain valve</i>
11	<i>Needle valve</i>
12	<i>Pressure relief valve</i>
13	<i>Heater</i>
14	<i>Transmitter Level</i>
15	<i>Pressure transducer</i>
16	<i>Float switch</i>
17	<i>Overflow/Vent valve</i>

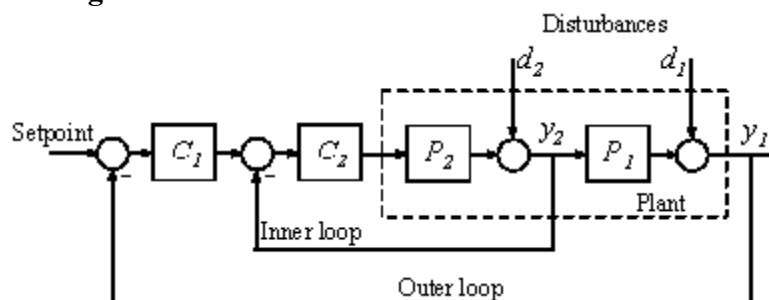
18	<i>Digital LCD displays</i>
19	<i>Indicator lights</i>

Control Module pada PCT 100 berisi rangkaian–rangkaian elektronik untuk menghubungkan *Process rig* dengan kontroler. Control module pada PCT 100 dilengkapi dengan tombol fault, di mana penjelasan untuk setiap fungsi dari setiap tombol fault ditampilkan pada Tabel 2.2 (Bytronics, 2015).

*Tabel 2.2 Penjelasan Fault Switch Control Module PCT*

Fault	Fungsi	Efek
F1	A/D Fault	<i>ADC lock up</i>
F2	Pump	Pompa tidak dapat aktif
F3	Temperature	<i>Display Pembacaan Temperature</i> maksimal
F4	Cooler	<i>Cooler</i> menyala terus menerus
F5	Heater	Display <i>heat</i> menampilkan pembacaan yang salah
F6	Flow meter	Tidak ada <i>flow feedback</i>

### 2.2.2 Sistem Pengaturan Cascade



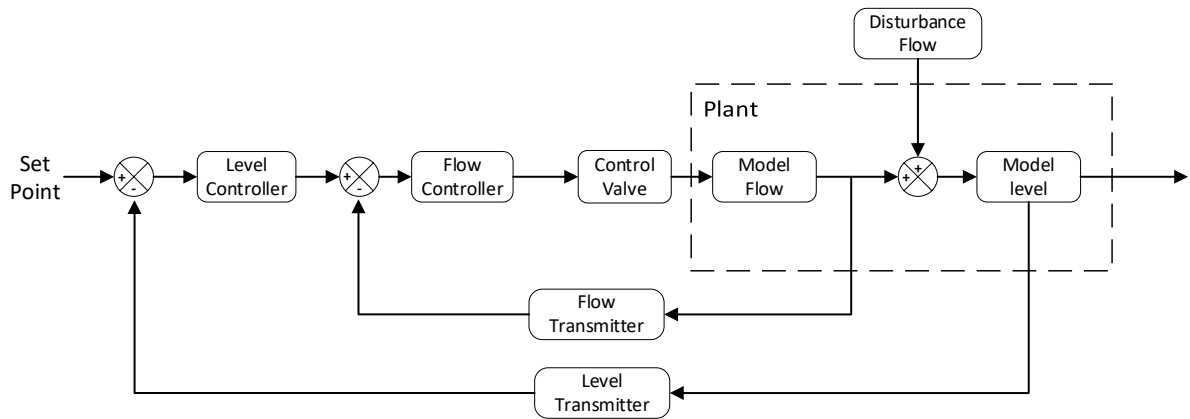
*Gambar 2.2 Skema Pengaturan cascade (mathworks, 2023)*

*Cascade control* adalah strategi kontrol yang digunakan dalam sistem otomasi untuk mengontrol proses yang kompleks. Strategi ini melibatkan penggunaan dua atau lebih *loop* kontrol yang saling terhubung secara hierarkis seperti tampak pada gambar 2.2. Dalam cascade control, terdapat dua tingkat loop kontrol, yaitu loop dalam (*inner loop*) dan loop luar (*outer loop*). Loop dalam yang bertanggung jawab untuk mengontrol variabel proses yang lebih cepat berubah, seperti debit aliran. Output dari loop dalam kemudian digunakan sebagai masukan untuk loop luar. Loop luar berada di tingkat yang lebih tinggi dan bertanggung jawab untuk mengontrol variabel proses yang memiliki perubahan lambat atau memiliki dampak yang lebih besar terhadap sistem secara keseluruhan (Johnson, 2014).

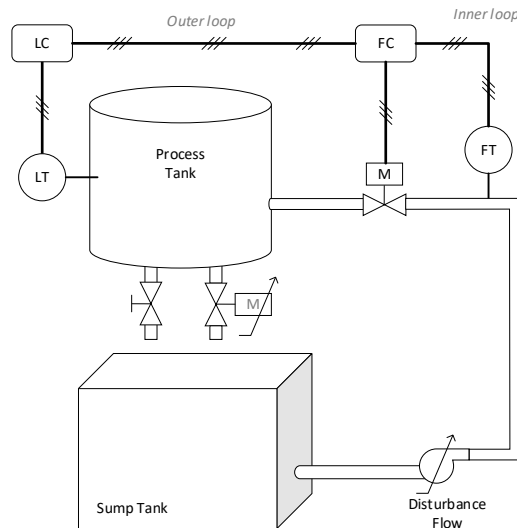
### 2.2.3 Sistem Pengaturan Cascade *Level-Flow* Tangki PCT-100

Perancangan sistem selanjutnya dilakukan dengan melakukan pemodelan matematis dari sistem pengaturan *level* cairan pada tangki. Secara umum prinsip dari pengaturan *level* pada tangki adalah dengan menjaga debit air yang masuk ke tangki bernilai sama dengan debit air yang keluar dari tangki. Sistem pada pengaturan *level* air tangki pada dasarnya berupa sistem orde satu. Akan tetapi jika dirangkai secara *cascade* maka sistem dapat menjadi memiliki orde yang lebih tinggi seperti pada gambar 2.3. Proses yang

terjadi pada pengaturan *level* air tangki dapat dilihat seperti ilustrasi pada Gambar 2.4 (Rahmadhani, 2022).



Gambar 2.3 Skema Proses Pengaturan Cascade Level pada Tangki



Gambar 2.4 Ilustrasi Sistem Pengaturan Level pada Tangki PCT-100

### 2.2.3.1 Pemodelan Sistem Pengaturan Level Tangki

Pemodelan sistem pengaturan level dapat dilakukan dengan melakukan pemodelan matematis. prinsip dari pengaturan level pada tangki. adalah dengan menjada debit air yang masuk ke tangki bernilai sama dengan debit air yang keluar dari tangki seperti pada gambar 2.5. Aliran air yang masuk dan keluar pada tangki dapat diturunkan dari hukum kekekalan massa. Karena massa jenis fluida yang masuk dan keluar pada tangki sama, maka, nilai  $\rho$  dapat diabaikan dari persamaan sehingga hanya dipengaruhi debit fluida yang masuk dan keluar pada tangki.

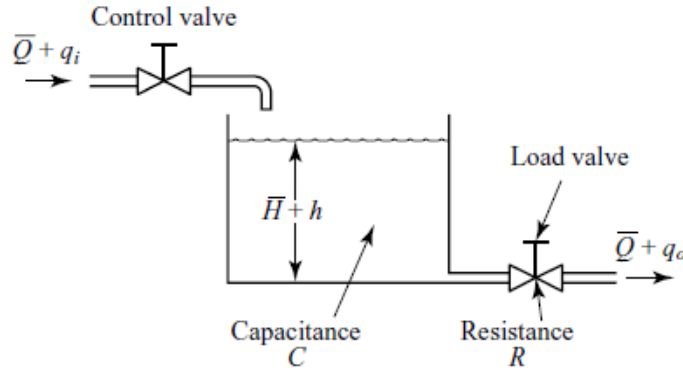
$$Q_{in} - Q_{out} = \frac{dV}{dt} \quad (2.1)$$

Volume (V) tangki merupakan perkalian antara luas alas (A) dengan tinggi (h). Bentuk tangki yang digunakan berupa tabung sehingga nilai luas alas selalu konstan



karena dinding tangki vertikal dan sejajar. Oleh karena itu,  $h$  adalah satu-satunya variabel yang nilainya dapat berubah.

$$Q_{in} - Q_{out} = A \frac{dh}{dt} \quad (2.2)$$



Gambar 2.5 Ilustrasi Pemodelan Sistem Pengaturan Level Tangki (Ogata, 2010)

Sistem pengaturan level pada tangki dapat disederhanakan dengan mengibaratkan seperti rangkain resistor dan kapasitor. Dimana kapasitor dianalogikan menjadi kapasitansi tangki dirumuskan dengan persamaan 2.3.

$$C = \frac{\Delta V_{tangki}}{\Delta h_{tangki}} = A_{tangki} \quad (2.3)$$

Nilai hambatan  $R$  dianalogikan sebagai perubahan ketinggian pada tangki terhadap perubahan debit output pada tangki untuk debit air yang keluar dianggap laminar yang dirumuskan dengan persamaan 2.4 (Ogata, 2010)

$$R = \frac{\Delta h}{\Delta Q_{out}} = \frac{h}{Q_{out}} \quad (2.4)$$

Dengan menggabungkan persamaan 2.2, 2.3, dan 2.4 diperoleh

$$RC \frac{dh}{dt} + h = RQ_{in} \quad (2.5)$$

Dengan menggunakan transformasi laplace, diperoleh transfer function ketinggian tangki terhadap debit masukan seperti persamaan 2.6.

$$\frac{H(s)}{Q_{in}(s)} = \frac{R}{RCs + 1} \quad (2.6)$$

### 2.2.3.2 Pemodelan Level Transmitter

*Level Transmitter* merupakan suatu transduser yang berfungsi untuk mengukur besaran level atau ketinggian dan mengubahnya menjadi sinyal tegangan. Pemodelan *level* bertujuan untuk memperoleh nilai penguatan yang disimbolkan dengan  $G_{LT}$ .

Penguatan *level Transmitter* dimodelkan dengan persamaan garis lurus seperti persamaan 2.7 dimana  $V_{LT}(h)$  adalah tegangan terukur dan  $c$  adalah *offset* pengukuran level

$$V_{LT}(h) = G_{LT} h + c \quad (2.7)$$

### 2.2.3.3 *Pemodelan Flow Transmitter*

*Flow Transmitter* merupakan suatu transduser yang berfungsi untuk mengukur besaran debit dan mengubahnya menjadi sinyal tegangan. Pemodelan *flow Transmitter* bertujuan untuk memperoleh nilai penguatan yang disimbolkan dengan  $G_{FT}$ . *Flow Transmitter* dimodelkan dengan menghitung perubahan volume air pada tangki terhadap perubahan waktu. Karena luas penampang dari tangki selalu tetap sehingga perubahan volume hanya bergantung dengan perubahan level pada tangki.

$$Q_{in} = \frac{\Delta V}{\Delta t} = A_{tabung} \frac{\Delta h}{\Delta t} \quad (2.8)$$

*Input Flow Transmitter* yang berupa debit air dirubah menjadi tegangan 0-10V. dapat dimodelkan dengan penguatan  $G_{FT}$  seperti pada persamaan 2.9

$$V_{FT} = G_{FT} Q_{in} \quad (2.8)$$

### 2.2.3.4 *Pendekatan Waktu Tunda Sistem Orde Satu*

Salah satu metode pendekatan waktu tunda yang digunakan adalah pendekatan deret orde pertama seperti pada Persamaan 2.9. dimana  $N$  merupakan orde sistem dan  $\theta$  adalah nilai waktu tunda sistem (Pritandi, 2016).

$$e^{\theta s} = \frac{1}{\left(\frac{\theta s}{N} + 1\right)^N} \quad (2.9)$$

## 2.2.4 **Kontroler PID**

Kontroler PID merupakan salah satu kontroler yang banyak digunakan untuk memperbaiki performa sistem di industri. Struktur kontroler PID yang sederhana menjadi salah satu faktor kontroler tersebut diminati. Parameter kontroler PID adalah penguatan proporsional  $K_p$ , waktu integral  $T_i$ , dan waktu derivatif  $T_d$ . Algoritma kontroler PID dapat dijelaskan pada Persamaan 2.11

$$u(t) = K_p \left( e(t) + \frac{1}{T_i} \int_0^\infty e(t) dt + T_d \frac{de(t)}{dt} \right) \quad (2.11)$$

$u(t)$  merupakan sinyal kontrol dan  $e(t)$  adalah sinyal kesalahan. Sinyal kontrol dari kontroler tersebut merupakan penjumlahan dari 3 aksi yaitu, aksi proporsional ( $P$ ), integral ( $I$ ) dan derivatif ( $D$ ). Kontroler PID memiliki fungsi untuk memperbaiki kesalahan melalui aksi proporsional, menghilangkan kesalahan pada keadaan tunak melalui aksi integral dan mengantisipasi kesalahan akan datang melalui aksi derivatif (Ghufron, 2016).

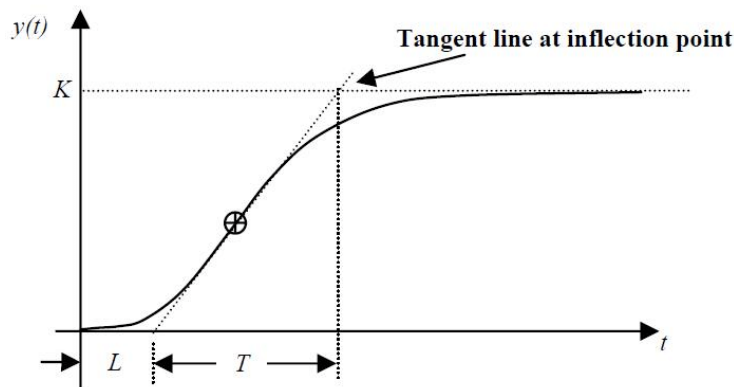
### 2.2.5 Identifikasi Sistem dan Aturan Kontrol PID Ziegler Nichols

Identifikasi sistem merupakan metode yang dilakukan untuk mendapatkan model matematis dari suatu *plant* berdasarkan data hasil pengujian. Pengujian ini dilakukan dengan memberikan sinyal uji pada masukan suatu *plant* dan mengukur hasil keluaran dari *plant*. Model matematis yang diperoleh dapat merepresentasikan hubungan antara masukan dan keluaran pada plant (Rizki, 2016).

Salah satu metode identifikasi sistem adalah metode Zeigler – Nichols. Metode ini dilakukan dengan cara memberikan sinyal step pada *input plant*. Sistem yang diidentifikasi menggunakan Zeigler – Nichols nantinya dimodelkan dalam persamaan orde satu dengan time delay. Seperti persamaan berikut.

$$G(s) = \frac{K e^{-Ls}}{Ts + 1} \quad (2.10)$$

Dalam perhitungan persamaan Zeigler – Nichols diperlukan nilai L dan T pada model sistem. Nilai L dan T merupakan nilai yang didapatkan dari garis singgung pada inflection point. Nilai tersebut diperoleh dengan menggambar garis singgung yang memotong sumbu waktu dan garis keluaran pada kondisi tunak seperti pada gambar 2.6. Untuk nilai L sendiri merupakan nilai *delay* dan nilai T merupakan nilai *time constant* dari sistem.



Gambar 2.6 Respon sistem yang menyerupai huruf S (Ogata, 2010)

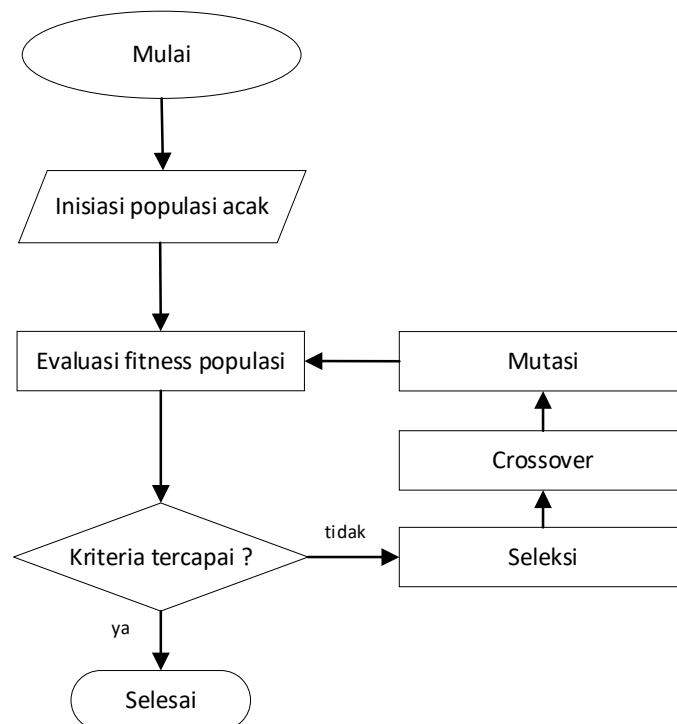
Identifikasi Zeigler – Nichols juga dapat langsung digunakan untuk mendesain kontroler PID. Nilai parameter kontroler PID ditentukan dengan mensubstitusikan L dan T yang telah diperoleh ke dalam Tabel 2.3.

Tabel 2.3 Aturan Tuning Kontroler PID Ziegler-Nichols Metode Open Loop

Kontroler	$K_p$	$T_i$	$T_d$
P	$\frac{T}{L}$	$\infty$	0
PI	$0,9 \frac{T}{L}$	$\frac{L}{0,3}$	0
PID	$1,2 \frac{T}{L}$	$2L$	$0,5 L$

### 2.2.6 Genetic Algorithm (GA)

*Genetic Algorithm* (GA) merupakan metode optimisasi yang didasari oleh proses seleksi alam dan evolusi pada makhluk hidup. GA akan berulang kali memodifikasi populasi individu yang menyatakan suatu solusi. Pada setiap langkah, algoritma genetika memilih individu dari populasi saat ini untuk menjadi orang tua dan menggunakannya untuk menghasilkan anak-anak untuk generasi berikutnya. Selama beberapa generasi berturut-turut, populasi akan "berevolusi" menuju solusi optimal (Melanie, 2007).



Gambar 2.7 Flowchart Genetic Algorithm

#### 2.2.6.1 Inisiasi populasi

GA bekerja dengan cara mencari nilai optimal dari sekumpulan data atau nilai yang disebut populasi. Inisiasi diawali dengan mengatur nilai batas atas dan batas bawah dari suatu nilai. Batas tersebut nantinya digunakan untuk menginisiasi populasi awal yang bernilai acak sesuai dengan jumlah populasi yang diinginkan yang berada dalam rentang yang telah ditentukan.

#### 2.2.6.2 Evaluasi fitness

Fungsi objektif ( $J$ ), digunakan untuk mengukur performa individu sebagai solusi suatu permasalahan. Pada kasus permasalahan minimisasi, individu yang sesuai memiliki nilai fungsi objektif yang paling rendah. Nilai performa yang diperoleh dari fungsi objektif digunakan untuk menentukan nilai ketahanan relatif individu. Fungsi *fitness*,  $f=g(J)$ , pada umumnya digunakan untuk merubah nilai fungsi objektif untuk mengukur nilai ketahanan relatif individu.

### 2.2.6.3 Seleksi

Seleksi merupakan salah satu komponen yang memegang peran penting dalam mendesain GA. Seleksi berfungsi untuk menentukan individu yang akan berperan untuk membentuk populasi baru. Tujuan dari seleksi adalah memastikan bahwa individu yang terpilih mengikuti proses reproduksi memiliki nilai *fitness* yang tinggi.

### 2.2.6.4 Crossover

*Crossover* meniru mekanisme alami pindah silang yang terjadi pada suatu kromosom untuk diaplikasikan secara acak pada individu yang terpilih dari hasil operator seleksi. Prinsip dari *crossover* adalah menukar sebagian informasi dari individu dengan individu pasangannya sehingga dapat diperoleh individu yang baru. Salah satu jenis *Crossover* yang banyak digunakan adalah *arithmetic crossover* yang biasa digunakan pada data berupa bilangan bulat dengan mengambil rata-rata dari kedua orang tua yang diberi bobot tertentu antara nol sampai satu. Adapun untuk rumus yang digunakan yaitu

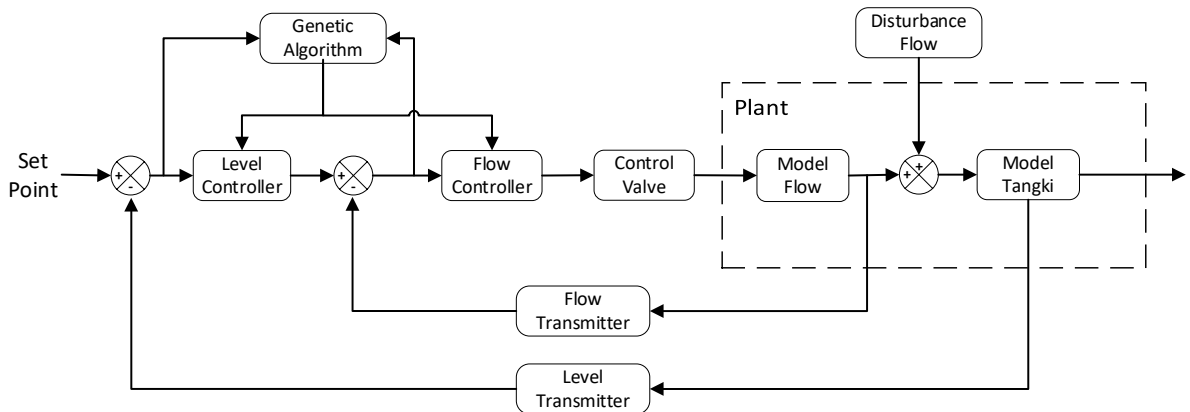
$$\begin{aligned} child1 &= \alpha * parent1 + (1 - \alpha) * parent2; \\ child2 &= (1 - \alpha) * parent1 + \alpha * parent2; \end{aligned} \quad (2.12)$$

### 2.2.6.5 Mutasi

Mutasi pada evolusi makhluk hidup merupakan proses pengantian satu alel suatu gen dengan alel gen lain untuk menghasilkan struktur genetika yang baru. Operator mutasi pada GA dapat menghasilkan individu baru yang tidak mungkin dapat diperoleh dari operator seleksi atau *crossover*.

## 2.2.7 Kontroler PID-GA

Kontroler PID-GA merupakan kontroler PID yang menggunakan mekanisme optimisasi GA pada *tuning* parameternya. Algoritma tersebut digunakan untuk mencari nilai parameter optimal berdasarkan fungsi objektif yang ditentukan. Pada kasus ini, fungsi objektifnya adalah meminimalkan kesalahan antara nilai *setpoint* dengan keluaran aktual. Nilai parameter kontroler PID pada kondisi awal ditentukan secara acak yang nantinya terus dievaluasi performanya tiap generasi menggunakan algoritma GA untuk memperoleh *fitness* terbaik (Yusuf et al., 2017). Adapun



Gambar 2.8 Diagram Blok Sistem pengaturan PID-GA

## 2.2.8 Indeks Performa

Dalam mencari suatu fungsi optimal tidak dapat didefinisikan dengan tepat oleh karena itu diperlukan Indeks performa untuk menilai suatu solusi dalam menangani permasalahan. Indeks performa banyak digunakan oleh kalangan praktisi dan juga akademisi untuk membantu dalam menentukan kualitas dari sebuah sistem. Indeks performansi sendiri sebenarnya adalah fungsi hubungan di mana beberapa karakteristik sistem seperti kondisi optimal dari sistem didefinisikan (Khuriati R.S., 2014).

### 2.2.8.1 *Integral Square Error*

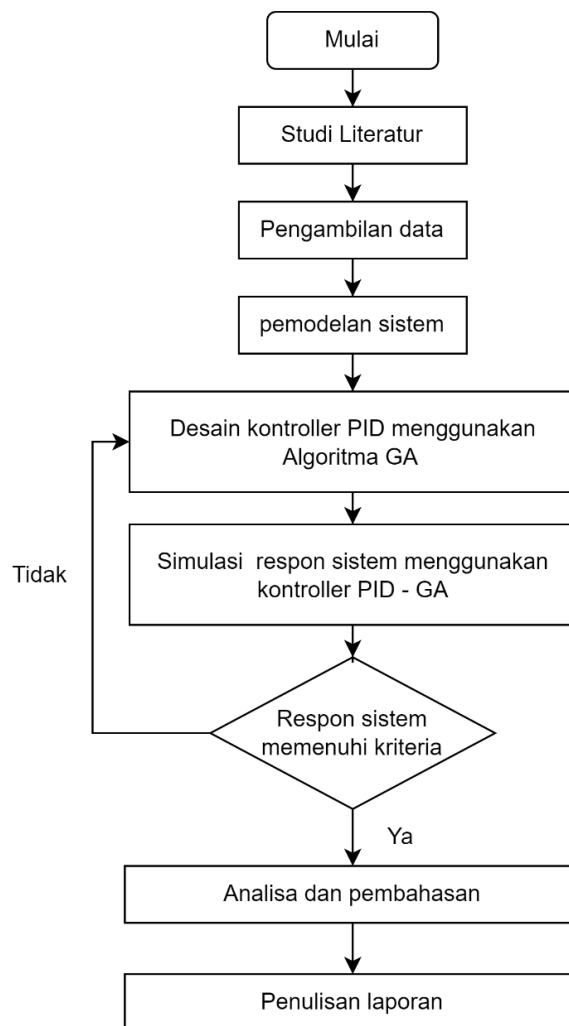
*Integral Square Error (ISE)* merupakan salah satu metode yang digunakan untuk mengukur kualitas dari suatu model kontrol atau identifikasi. ISE mengukur seberapa besar eror atau selisih antara sinyal yang dihasilkan oleh model dengan sinyal aktual yang ada pada sistem. ISE mengintegrasikan kuadrat kesalahan dari waktu ke waktu. *error* yang kecil saja akan menghasilkan ISE yang besar. Dengan meminimalkan ISE cenderung menghilangkan kesalahan besar dengan cepat, tetapi akan mentolerir kesalahan kecil yang bertahan untuk jangka waktu yang lama,.

$$ISE = \int_0^t e^2(t) dt \quad (2.11)$$

## BAB 3 PERANCANGAN SISTEM

### 3.1 Metode yang digunakan

Dalam tugas akhir ini, dilakukan dengan simulasi dan implementasi. Langkah tersebut dimulai dari pengambilan data dari respon sistem, identifikasi sistem, pemodelan sistem, perancangan kontroler, pengambilan data simulasi, dan penulisan laporan. Pengambilan data respon sistem dilakukan dengan memberikan sinyal step pada *control valve* sehingga diketahui karakteristik sistem dari *control valve* tersebut. Sedangkan untuk sistem pengaturan level dimodelkan secara matematis dari penurunan rumus secara fisis berdasarkan parameter fisik pada tangki PCT 100. Desain kontroler akan dilakukan dengan mensimulasikan algoritma GA mencari parameter kontroler PID serta dibandingkan dengan *tuning* PID *Ziegler Nichols*. Adapun metodologi penelitian ini dapat dijelaskan dalam *flowchart* pada gambar 3.1.



Gambar 3.1 Flowchart diagram penelitian

### 3.2 Bahan dan peralatan yang digunakan

Pengerjaan tugas akhir ini memerlukan bahan dan peralatan sebagai berikut.

### 3.2.1 Modul Process Control Technology 100

PCT-100 merupakan produk dari *Bytronic* yang digunakan untuk mendemonstrasikan prinsip kontrol proses industri. Pada peralatan ini terdapat proses kontrol variabel proses yang dapat diatur yaitu tekanan (*pressure*), ketinggian air (*level*), debit air (*flow*), dan suhu (*temperature*). Untuk menjalankan proses kontrol terdapat aktuator pada PCT -100 antara lain pompa, *flow control valve*, *drain valve* dan pemanas seperti pada gambar 3.2. Pompa digunakan untuk menyalurkan air dari tangki penyimpanan menuju tangki utama, *flow control valve* untuk mengatur jumlah debit didalam tangki, *drain valve* untuk mengatur ketinggian level pada tangki, Pemanas digunakan dalam proses sistem pengaturan suhu.

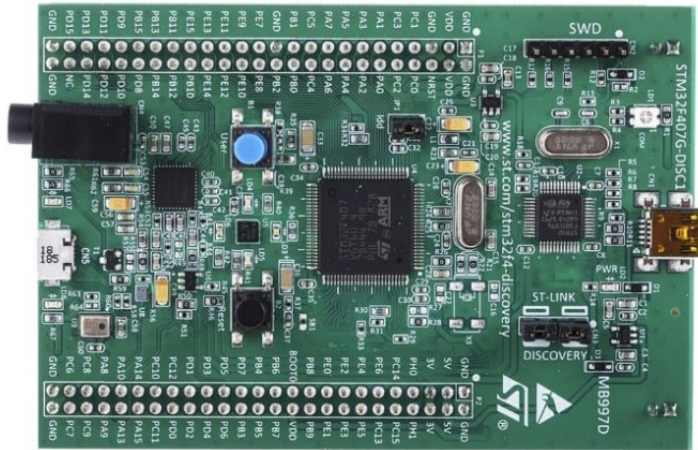


Gambar 3.2 Module Control PCT-100

### 3.2.2 STM32F407VG

STM32F407VG adalah mikrokontroler 32-bit dari jenis STM32F4 yang diproduksi STMicroelectronics seperti pada gambar 3.3. Perangkat menggunakan inti core ARM Cortex-M4, yang menawarkan kinerja tinggi dengan kecepatan hingga 168 MHz. Mikrokontroler ini memiliki berbagai fitur seperti *Flash memory* internal sebesar 1 MB, SRAM sebesar 192 KB, dan berbagai periferal seperti komunikasi SPI, I2C, USART, USB, ADC, dan lain-lain (STMicroelectronics, 2020).



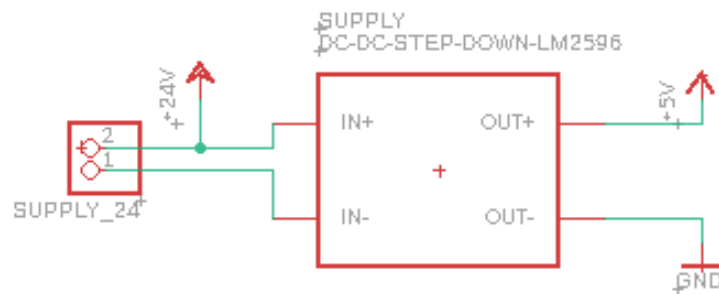


Gambar 3.3 Mikrokontroler STM32F407VG

### 3.2.3 Rangkaian Modul Mikrokontroler

#### 3.2.3.1 Rangkaian Power Supply

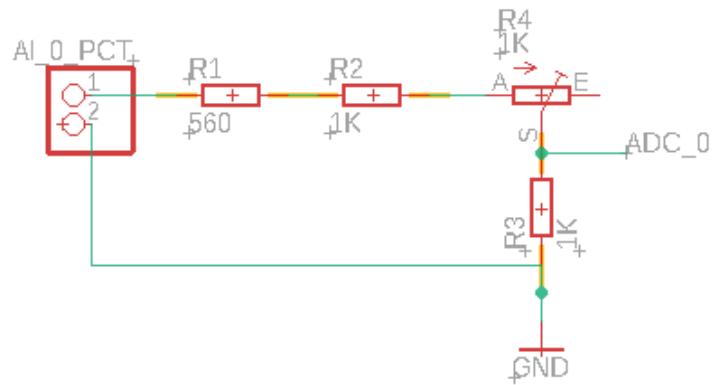
Rangkaian *Power supply* digunakan untuk mengatur tegangan *input* berupa tegangan DC 24 V menjadi beberapa tegangan antara lain tegangan *supply* mikrokontroler dan tegangan *supply* rangkaian *amplifier regulator*. Untuk *supply* tegangan 24 V untuk rangkaian *amplifier regulator* akan langsung dihubungkan dari *input power supply*. Sedangkan tegangan *supply* mikrokontroler 5V akan disupply dari modul DC *stepdown* LM2596. Adapun untuk rangkaian *power supply* dapat dilihat pada gambar 3.4.



Gambar 3.4 Rangkaian Power Supply

#### 3.2.3.2 Modul ADC

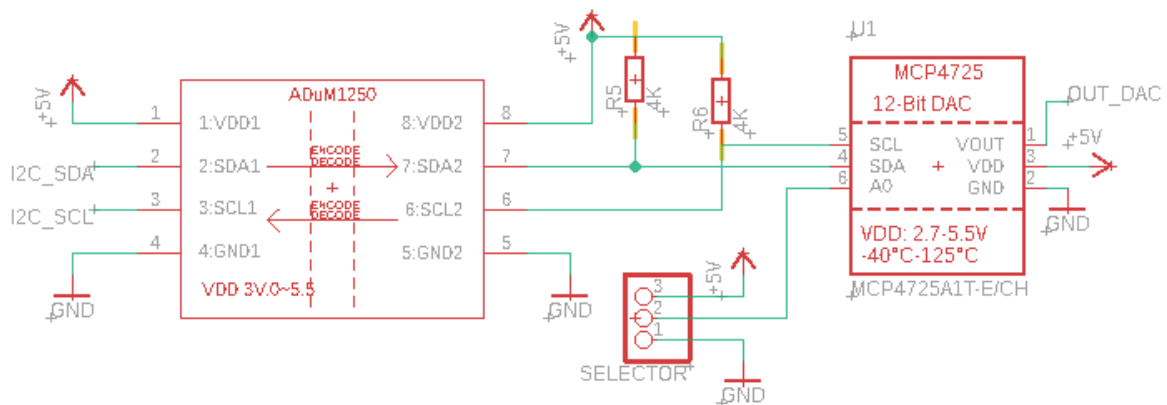
Modul ADC digunakan untuk mengubah output tegangan pembacaan PCT-100 yang berupa sinyal tegangan 0-10 V menjadi sinyal tegangan 0-5 V sebelum masuk ke Mikrokontroler. Pada modul ADC ini digunakan rangkaian pembagi tegangan seperti pada gambar 3.3.



Gambar 3.5 Rangkaian Modul ADC

### 3.2.3.3 Modul komunikasi DAC

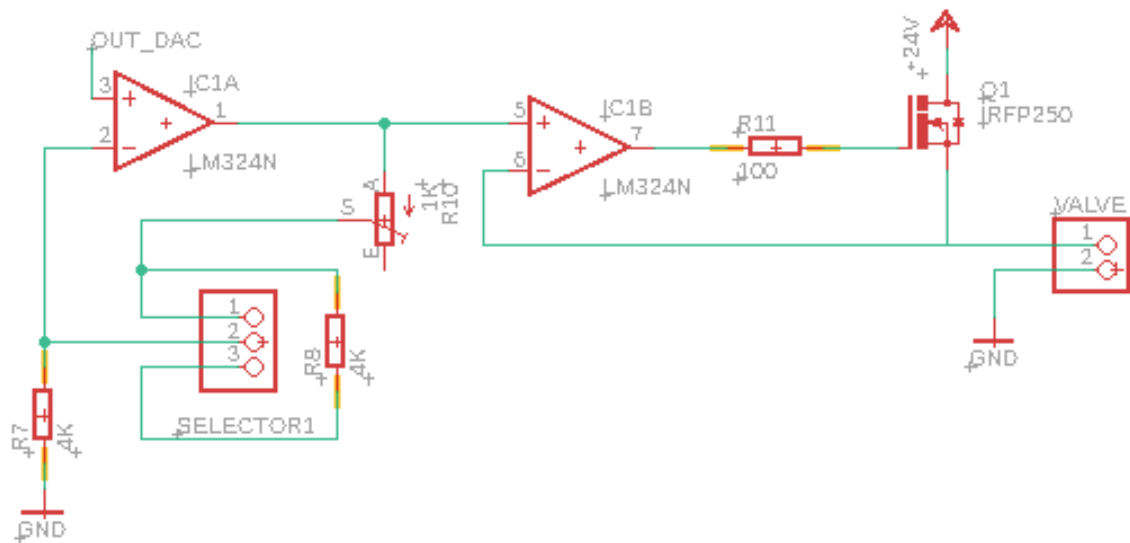
Rangkaian modul komunikasi DAC menggunakan komunikasi I2C untuk mengirim data dari *transmitter* ke mikrokontroler dan sebaliknya. Dalam rangkaian modul DAC digunakan modul ADUM 1250 untuk komunikasi duplex antara mikrokontroler dengan aktuator. Untuk berkomunikasi dengan aktuator perlu mengetahui *address* dari masing-masing aktuator. Aktuator bawaan dari PCT tidak memiliki adress yang diperlukasn dalam komunikasi I2C sehingga digunakan modul MCP4725 untuk memberikan *address* kepada aktuator. Selain Itu MCP4725 juga dapat mengubah bit keluaran mikrokontroler 0-4096 menjadi sinyal tegangan 0-5 volt kepada aktuator. Adapun rangkaian komunikasi DAC dapat dilihat pada gambar 3.6.



Gambar 3.6 Rangkaian Modul DAC

### 3.2.3.4 Rangkaian Amplifier dan Voltage Follower

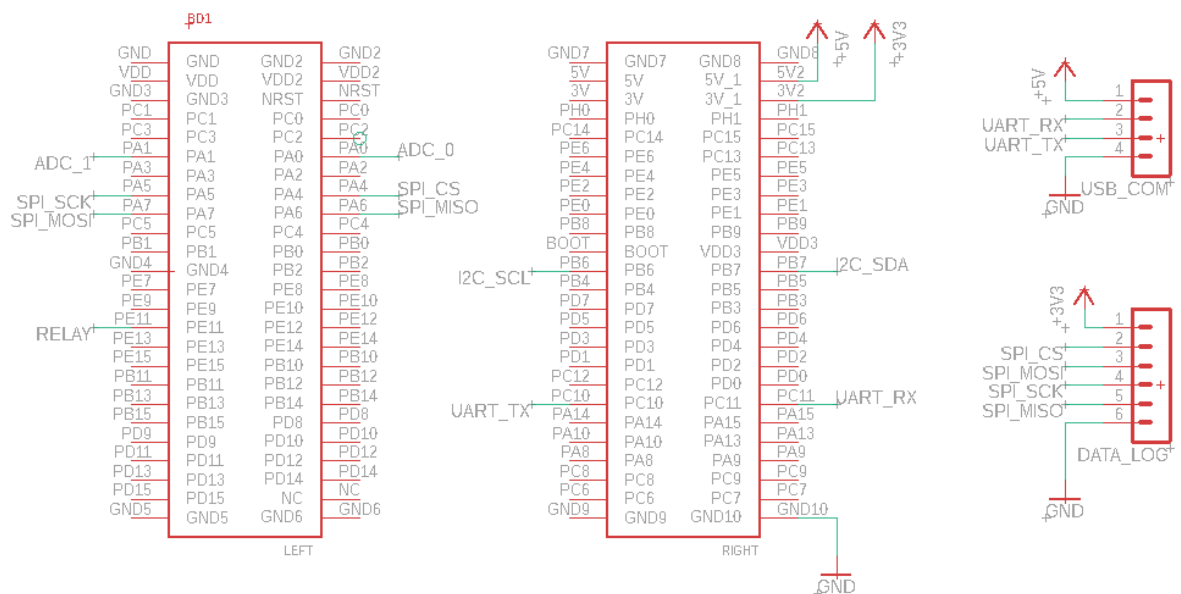
Rangkaian Amplifier digunakan untuk menguatkan sinyal keluaran DAC dari 0-5 V menjadi 0-24V menuju *control valve*. Rangkaian amplifier menggunakan jenis non inverting yang terhubung dengan op-amp sebelah kiri. Setelah sinyal diberi penguatan sinyal akan masuk kedalam rangkaian Voltage Follower. Rangkaian Voltage follower berfungsi untuk penyangga isolasi untuk mencegah adanya drop sinyal tegangan aktuator ketika terhubung dengan mikrokontroller.



Gambar 3.7 Rangkaian Amplifier dan Regulator

### 3.2.3.5 Rangkaian Data Logger dan Komunikasi Serial

Rangkaian Data Logger digunakan untuk menyimpan data yang masuk kedalam mikrokontroler yang merupakan nilai yang telah diatur dan nilai yang diperoleh untuk tiap instrumen. Komunikasi serial digunakan untuk menghubungkan komputer ke mikrokontroler agar pengguna dapat memasukkan perintah ke mikrokontroler. Ada beberapa jenis komunikasi serial yang digunakan pada mikro kontroler yaitu komunikasi I2C yang digunakan untuk komunikasi mikrokontroler dengan aktuator. Komunikasi SPI yang digunakan untuk komunikasi mikrokontroler dengan modul *data logger*. Dan komunikasi UART yang digunakan untuk komunikasi mikrokontroler dengan komputer melalui kabel USB.



Gambar 3.8 Rangkaian Mikrokontroler dan Data Logger

### 3.2.4 MATLAB dan Simulink 2020b

MATLAB adalah merupakan sebuah bahasa pemrograman dan lingkungan komputasi numerikal yang digunakan untuk mengelola data, menghitung, dan mengembangkan algoritma. MATLAB merupakan program yang dikembangkan oleh *MathWorks* yang menyediakan fungsi-fungsi matematis, visualisasi data, dan pemodelan sistem untuk dapat digunakan dalam melakukan simulasi dari sebuah sistem. MATLAB juga dapat digunakan untuk membangun sistem kontrol, memproses sinyal, dan menganalisis data secara *realtime*.

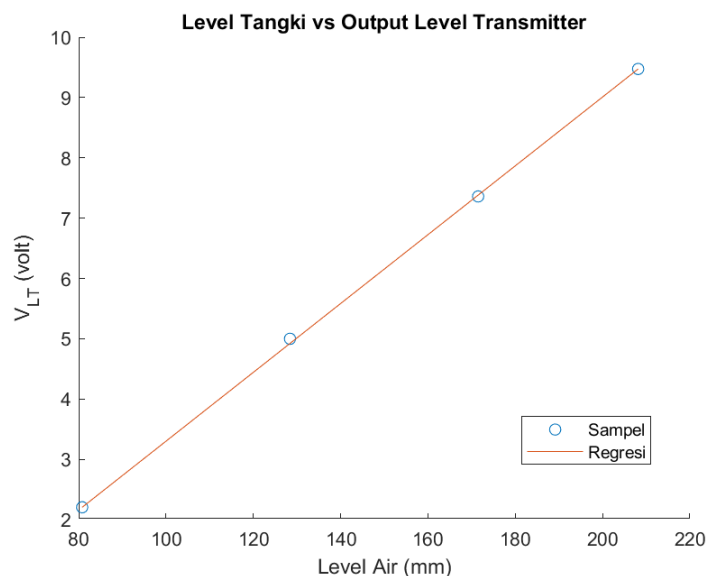
*Simulink* adalah sebuah perangkat lunak yang dikembangkan oleh *MathWorks* yang memungkinkan untuk mensimulasikan sistem dalam bentuk diagram blok. *Simulink* memiliki banyak library yang berisi blok-blok yang umum digunakan dalam simulasi sebuah sistem seperti sumber, fungsi matematis, *transfer function*, dan lainnya. Dalam menggunakan *Simulink* perlu untuk menghubungkan tiap blok yang digunakan untuk membangun model sistem yang diinginkan. Kemudian, *Simulink* digunakan untuk menganalisis dan menjalankan simulasi model tersebut untuk menguji dan mengevaluasi bagaimana sistem bekerja dalam situasi nyata

## 3.3 Urutan pelaksanaan penelitian

### 3.3.1 Pemodelan Level Transmitter

Pemodelan level transmitter dilakukan dengan pengambilan data menggunakan STM32F407VG sebagai *data acquisition module*. Data diperoleh berupa tegangan level transmitter ( $V_{LT}$ ) yang nantinya akan dibandingkan dengan hasil pengukuran ketinggian air ( $h$ ) secara manual menggunakan jangka sorong. Data  $V_{LT}$  dan  $h$  dihubungkan menggunakan regresi linier untuk mendapat persamaan garis lurus seperti persamaan 2.7.

Respon tegangan transmitter terhadap 4 sampel ketinggian air tampak pada gambar 3.9. Perhitungan gradien dan konstanta persamaan garis  $V_{LT}(h)$  digunakan 2 sampel yaitu level terendah  $h_0 = 80.8mm$  dan tertinggi  $h_4 = 208.15mm$ . Nilai tegangan pada kedua hitung menggunakan persamaan garis lurus untuk memperoleh nilai  $G_{LT}$  dan  $c$ .



Gambar 3.9 Respon tegangan level transmitter terhadap ketinggian air

$$G_{LT} = \frac{V_{LT4} - V_{LT1}}{h_4 - h_1} = \frac{9.4739 - 2.2046}{20.815 - 8.08} = 0.571 \frac{\text{Volt}}{\text{cm}}$$

$$c = V_{LT4} - G_{LT} h_4 = 9.4739 - (0.571 * 20.815) = -2.4076 \text{ Volt}$$

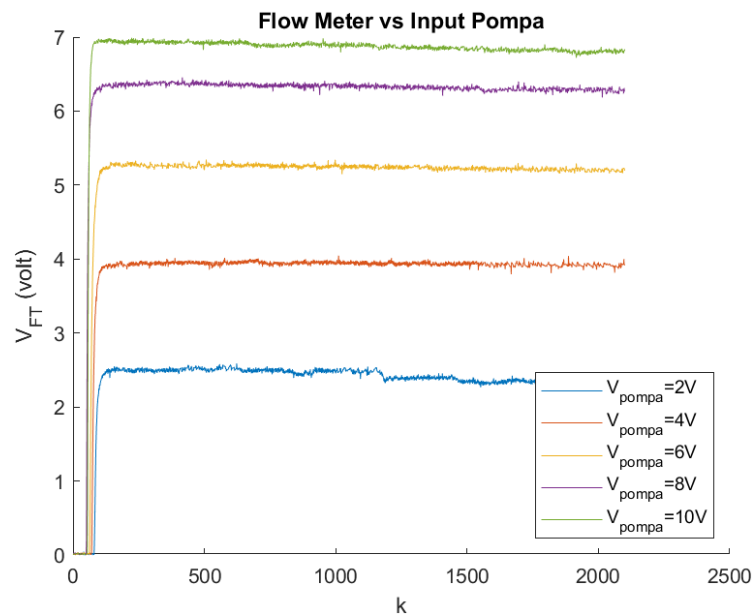
Fungsi tegangan output  $V_{LT}(\text{Volt})$  terhadap *input* ketinggian air level transmitter  $h(\text{cm})$  diperoleh sebagai fungsi persamaan linier berikut.

$$V_{LT}(h) = 0.571 h - 2.4076$$

### 3.3.2 Pemodelan Flow Transmitter

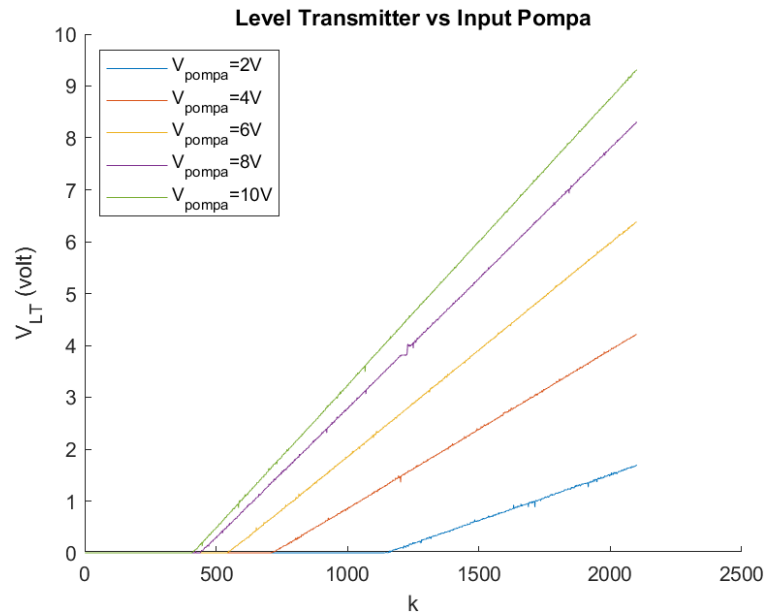
Pemodelan flow transmitter dilakukan dengan pengambilan data menggunakan STM32F407VG sebagai *data acquisition module*. Data diperoleh ketika *manual valve* dan *drain valve* dalam keadaan tertutup. Data yang digunakan adalah tegangan pada *flow transmitter*, durasi pengambilan data, dan ketinggian air. Identifikasi model *flow transmitter* dilakukan dengan membandingkan laju perubahan ketinggian air dengan tegangan *flow transmitter*.

Pemodelan *flow transmitter* Diawalali dengan membaca tegangan *flow transmitter* terhadap variasi tegangan *input* pada pompa ketika *control valve* dalam keadaan terbuka. Respon tegangan  $V_{FT}$  terhadap *input* pompa tampak pada gambar 3.10.



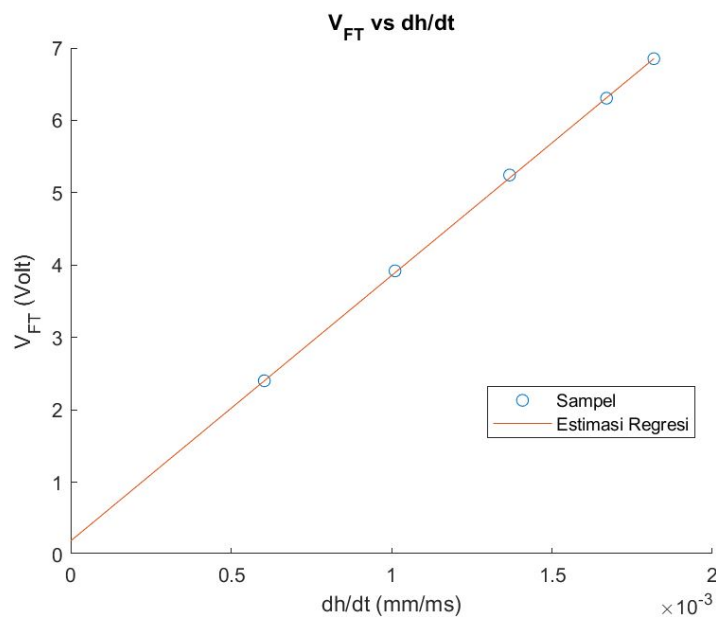
Gambar 3.10 Gambar Tegangan Flow Meter terhadap Variasi Input Pompa

Dari gambar grafik 3.10 terlihat bahwa *flow transmitter* membaca tegangan yang konstan untuk tiap variasi *input* pompa. Dengan asumsi tersebut dapat dianggap bahwa pompa dapat memberikan debit yang konstan. Laju debit *input* yang konstan membuat level air pada tangki meningkat seiring waktu. Tegangan pembacaan tersebut nantinya dibandingkan dengan laju kenaikan level pada tangki. Adapun laju penambahan level tangki untuk tiap variasi *input* pompa tampak pada gambar 3.11.



Gambar 3.11 Laju perubahan ketinggian level terhadap variasi input pompa

Dari grafik gambar 3.11 terlihat bahwa untuk tiap *input* tegangan pompa memiliki gradien kenaikan *level* yang berbeda-beda. Gradien tiap variasi *input* pompa dihitung untuk tiap variasi dan dibandingkan terhadap pembacaan tegangan *flow transmitter* untuk dapat memodelkan tegangan  $V_{FT}$  terhadap perubahan ketinggian *level* tangki.



Gambar 3.12 Perbandingan tegangan  $V_{FT}$  terhadap perubahan ketinggian level tangki

Dari grafik 3.13 tegangan pompa dapat dimodelkan menjadi persamaan garis lurus dengan menggabungkan persamaan 2.8 dan 2.9 menjadi persamaan

$$V_{FT} \left( \frac{dh}{dt} \right) = \frac{A_{tabung}}{m_{ft}} \times \frac{dh}{dt} + c_{FT}$$

Nilai  $m_{FT}$  dan  $c_{FT}$  dapat diperoleh berdasarkan grafik 3.13. dimana  $m_{FT}$  adalah gradien  $V_{FT}$  terhadap perubahan ketinggian tangki dan  $c_{FT}$  adalah *offset* persamaan garis

$$\frac{A_{tabung}}{m_{ft}} = \frac{V_{FT pompa10V} - V_{FT pompa2V}}{\left(\frac{\Delta h}{\Delta t}\right)_{pompa2V} - \left(\frac{\Delta h}{\Delta t}\right)_{pompa10V}}$$

$$\frac{A_{tabung}}{m_{ft}} = \frac{6.8518 - 2.4011}{0.0018 - 6.0358 \times 10^{-4}} = 3666 \frac{Volt}{mm/ms}$$

$$c_{FT} = V_{FT t=120s} - m_{FT} \left(\frac{\Delta h}{\Delta t}\right)_{t=120s}$$

$$c_{FT} = 6.8518 - (3666 \times 0.0018) = 0.253$$

Substitusikan ke persamaan  $V_{FT} \left(\frac{dh}{dt}\right)$ :

$$V_{FT} \left(\frac{dh}{dt}\right) = 3666 \times \frac{dh}{dt} + 0.0253$$

Untuk mengubah fungsi  $V_{FT}$  terhadap laju pertambahan level menjadi  $V_{FT}$  terhadap debit perlu dikalikan dengan luas penampang tabung. Diketahui  $A_{tabung} = \pi r_{tabung}^2 = 8100\pi \text{ mm}^2$

$$3666 = \frac{A_{tabung}}{m_{FT}}$$

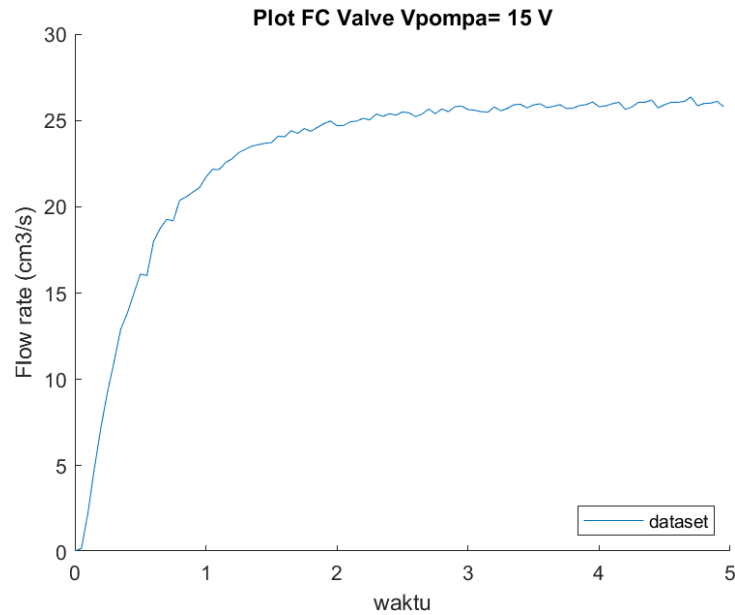
$$m_{FT} = \frac{8100\pi}{3666} = 6.9408 \frac{mm^3}{Volt.ms} = 6.9408 \frac{cm^3/s}{Volt}$$

$m_{FT}$  merupakan konstanta yang mengubah satuan tegangan menjadi debit. Sedangkan *flow transmitter* merupakan instrumen yang mengubah besaran debit menjadi tegangan sehingga hubungan antara  $m_{FT}$  dan penguatan flow transmitter  $G_{FT}$  adalah

$$G_{FT} = \frac{1}{m_{FT}} = \frac{1}{6.9408} = 0.144 \frac{Volt}{cm^3/s}$$

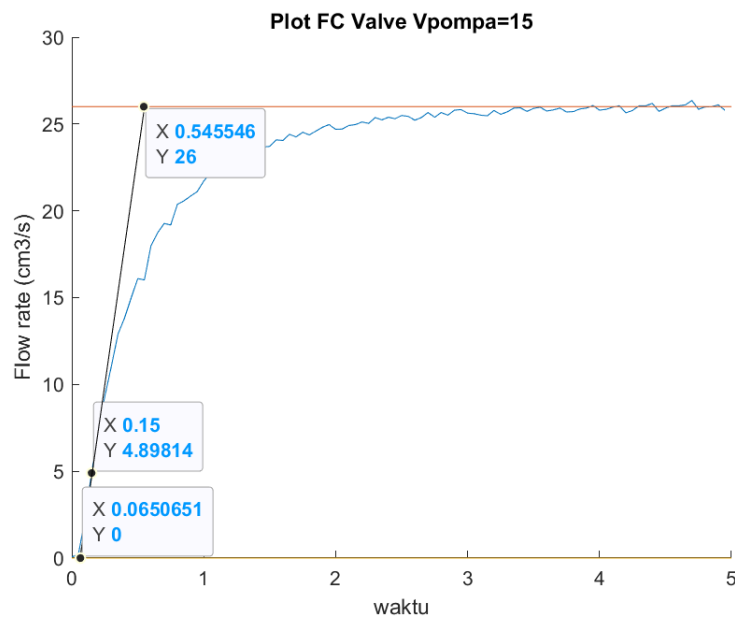
### 3.3.3 Pemodelan Sistem Pengaturan Flow

Prinsip kerja dari pengaturan *flow* adalah menjaga agar aliran air yang berada dalam pipa sesuai dengan debit yang diinginkan dengan cara mengatur bukaan *control valve* sebagai aktuator. Debit yang masuk kedalam pipa dipengaruhi oleh cepat atau lambatnya *control valve* dalam mengatur bukaanya. Untuk mengetahui respon debit air terhadap bukaan *valve* digunakan metode identifikasi sistem. Pengambilan data dilakukan menggunakan STM32F407VG sebagai data acquisition dengan nilai tegangan *input* 15 volt atau *input step* sebesar 3 volt karena *valve* memiliki rentang kerja 12-22 volt. Adapun data pengujian *step respon* tampak pada gambar 3.13



Gambar 3.13 Hasil Pengujian Step Respon Model Flow

Dari respon tersebut digunakan metode pendekatan dari Ziegler-Nichols open loop. Metode tersebut dapat digunakan karena dari kurva respon step open-loop yang diperoleh nampak bahwa sistem berbentuk seperti kurva S. dari kurva tersebut dicari gradient dari titik inflection point pada gambar 3.14.



Gambar 3.14 Penentuan Titik Inflection Point

Dari gambar garis singgung *inflection point* diperoleh nilai L dan sehingga diperoleh parameter sebagai berikut

$$Y_{ss} = 26; \quad X = 3$$

$$K = \frac{Y_{ss}}{X} = \frac{26}{3} = 8.67$$



$$L = 0.065$$

$$L + T = 0.546$$

$$T = 0.481$$

Nilai parameter tersebut dapat disubstitusikan ke persamaan 2.10 sebagai berikut  
Sehingga diperoleh transfer function  $G_{flow}$ .

$$G_{flow}(s) = \frac{8.67}{0.481s + 1} * e^{-0.065s}$$

Nilai *delay* dapat diubah persamaan 2.9 menjadi.

$$e^{-0.065s} = \frac{1}{1 + 0.065s}$$

Sehingga transfer function untuk sistem pengurangan flow menjadi

$$G_{flow}(s) = \left( \frac{8.67}{0.481s + 1} \right) \left( \frac{1}{1 + 0.065s} \right)$$

$$G_{flow}(s) = \frac{8.67}{0.0312s + 0.564s + 1}$$

### 3.3.4 Pemodelan Sistem Pengaturan Level

Prinsip kerja dari pengaturan *level* adalah menjaga agar aliran air yang masuk dan keluar dari tangki sama besarnya sesuai hukum kekekalan massa, yang sesuai dengan Persamaan 2.1. Pemodelan sistem pengaturan level dilakukan dengan pengambilan parameter pada sistem sehingga diperoleh parameter seperti pada tabel 3.1. Nilai  $h$  diperoleh dari pengamatan ketika nilai debit *input* diberi nilai tetap dan nilai  $h$  mencapai kondisi *steady state*. Berdasarkan persamaan 2.2 ketika nilai perubahan ketinggian sama dengan nol maka debit *input* sama dengan debit output pada tangki.

Tabel 3.1 Parameter Fisis Tangki PCT-100

No	Parameter	Nilai
1	$R_{tangki}$	9 cm
2	$h$	9.77 cm
3	$Q_{out}$	16.66 cm <sup>3</sup> /s

Setelah menemukan nilai parameter digunakan persamaan 2.3, 2.4 dan 2.6 untuk memperoleh model fungsi alih dari sistem pengaturan level sebagai berikut.

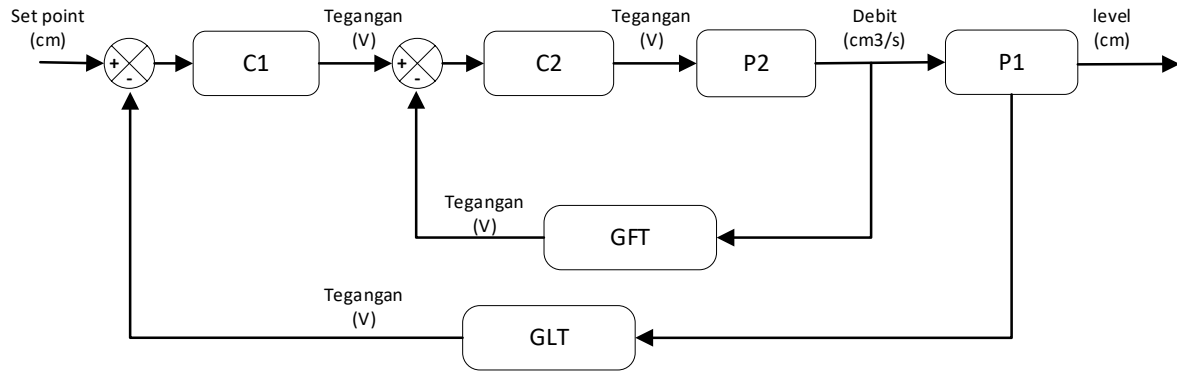
$$C = A_{tangki} = \pi R_{tangki}^2 = 3.14 (9)^2 = 254.47 \text{ cm}^2$$

$$R = \frac{h}{Q_{out}} = \frac{9.77}{16.66} = 0.5864$$

$$\frac{H(s)}{Q_{in}(s)} = \frac{R}{RCs + 1} = \frac{0.5864}{149.22s + 1}$$

### 3.3.5 Desain kontroler PID Ziegler-Nichols

Dalam desain kontroler *cascade* seperti pada gambar 3.15, kontroler *inner loop* perlu dituning terlebih dahulu. Kontroler *inner loop* dapat dituning menggunakan sehingga aturan *tuning* kontroler PID Ziegler Nichols metode pertama karena memiliki respon sistem *open loop* yang seperti kurva S. Berdasarkan pemodelan sistem pengaturan *flow*. Diperoleh nilai konstanta L dan T yang dapat digunakan untuk mencari nilai  $K_p$ ,  $T_i$ ,  $T_d$  seperti pada tabel 2.3. sehingga diperoleh nilai pada tabel 3.2



Gambar 3.15 Diagram Blok Konfigurasi cascade

Tabel 3.2 Parameter PID inner loop

Kontroler Inner	$K_p$	$T_i$	$T_d$
PID	8.88	0.13	0.0325

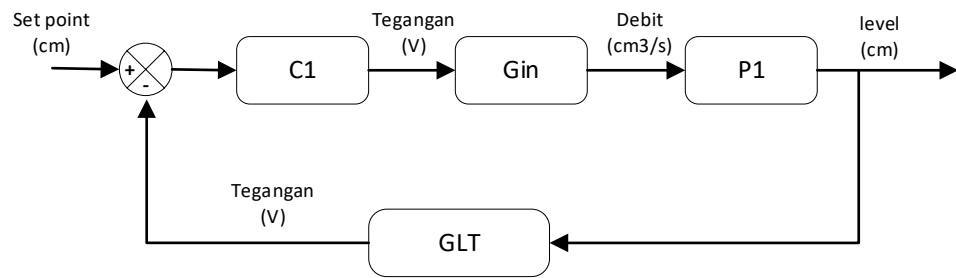
Setelah memperoleh kontroler *inner* sistem close loop *inner* perlu disederhanakan menjadi suatu *transfer function*. Adapun penyederhaan tersebut menjadi sebagai berikut.

$$G_{in} = \frac{C_2 P_2}{1 + C_2 P_2 G_{FT}}$$

Berdasarkan gambar 3.16  $C_2$  merupakan Kontroler PID inner loop sedangkan  $P_2$  adalah *transfer function* sistem pengaturan *flow* dan  $G_{FT}$  merupakan penguatan *flow transmitter*. Dengan memasukkan *transfer function* dari PID *inner*, control valve dan  $G_{FT}$  diperoleh penyederhanaan  $G_{in}$  sebagai berikut:

$$G_{in} = \frac{0.2817 s^2 + 76.96 s + 1.127}{0.03127 s^3 + 0.5866 s^2 + 12.09 s + 0.1623}$$

Penyederhanaan tersebut membuat diagram blok menjadi seperti pada gambar 3.17. Nilai set point yang dimasukkan kepada sistem merupakan besaran ketinggian (cm). Sedangkan kontroler menerima *feedback* berupa tegangan yang mewakili *level* ketinggian. Sehingga nilai set point perlu diubah menjadi tegangan dengan mengalikan set point dengan  $G_{LT}$ . Karena terdapat dua  $G_{LT}$  sebelum *summing point* maka dapat disederhanakan dengan merubah posisi  $G_{LT}$  menjadi setelah *summing point*.



Gambar 3.16 Penyederhanaan inner loop cascade

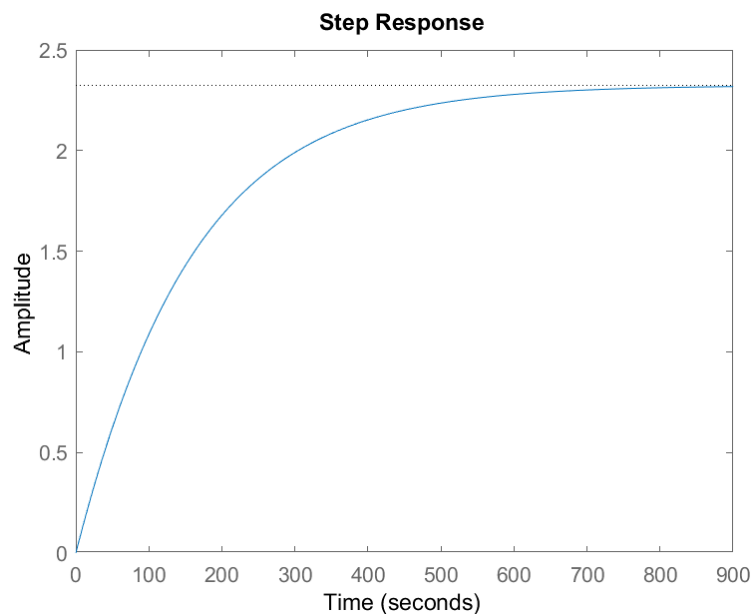
Untuk mendesain kontroler C1 diperlukan *transfer function* dari sistem keseluruhan ( $G_{out}$ ). Adapun untuk transfer function  $G_{out}$  dapat diperoleh dari penyederhanaan berikut.

$$G_{out} = G_{in} P_1 G_{LT}$$

Dengan mensubstitusi  $G_{in}$  yang telah diperoleh serta  $P_1$  yang merupakan transfer function tangki dan  $G_{LT}$  yang merupakan Penguatan *level transmitter* diperoleh *transfer function*  $G_{out}$  sebagai berikut.

$$G_{out} = \frac{0.09431 s^2 + 25.77 s + 0.3772}{4.665 s^4 + 87.56 s^3 + 1804 s^2 + 36.31 s + 0.1623}$$

Sistem yang diperoleh merupakan sistem yang memiliki 4 pole dan 2 zero. Sistem yang memiliki orde tinggi tersebut sulit untuk dilakukan identifikasi karakteristik sistem. Sehingga identifikasi karakteristik sistem dilihat dari respon *open loop* sistem tersebut. Adapun respon *open loop* sistem tersebut tampak pada gambar 3.18.



Gambar 3.17 Step Response  $G_{out}$

Dari plot respon tersebut nampak bahwa sistem memiliki respon menyerupai sistem orde 1 dengan penguatan K sebesar 2.322 dengan time constant (63%) pada saat 155 detik. Sehingga  $G_{out}$  dapat disederhanakan menjadi.

$$G_{out} = \frac{2.322}{155s + 1}$$

Model penyederhanaan  $G_{out}$  merupakan sistem orde 1 sehingga tidak dapat menggunakan metode Ziegler-Nichols untuk melakukan tuning kontrolernya. Untuk melakukan tuning pada sistem pengaturan level dilakukan secara perhitungan analitik. Adapun untuk karakteristik sistem yang diharapkan sebagai berikut.

$$t_s(5\%) = 120 \text{ detik}$$

$$E_{ss} = 0$$

$$\%overshoot < 5\%$$

Dengan karakteristik tersebut dapat dihitung untuk mencari  $\tau_i$  dan  $K_p$  sebagai berikut

$$\tau_i = \tau_{level} \approx 155$$

$$t_s^*(5\%) = 3\tau^*$$

$$120 = 3\tau^*$$

$$\tau^* = 40$$

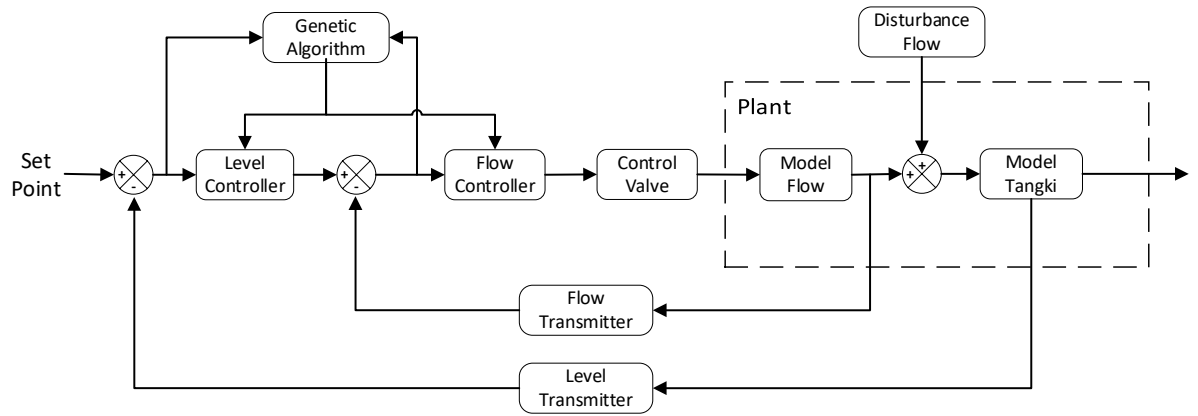
$$Kp_{outer} = \frac{\tau_i}{\tau^*K} = \frac{155}{40 \times 2.32} = 1.67$$

$$Ki_{outer} = \frac{K_p}{\tau_i} = \frac{1.67}{155} = 0.0108$$

### 3.3.6 Desain Kontroler PID-GA

Parameter kontroler PID pada penelitian ini direpresentasikan dalam bentuk bilangan riil dimana sebuah kromosom terdiri dari enam pasang nilai yaitu  $Kp_{outer}$ ,  $Ki_{outer}$ ,  $Kd_{outer}$ ,  $Kp_{iner}$ ,  $Ki_{iner}$ , dan  $Kd_{iner}$ . Tipe operator GA yang digunakan yaitu, seleksi *elitism*, *arithmetic crossover*, dan *boundary mutation*. Adapun nilai probabilitas mutasi yang digunakan adalah 0,25. Nilai parameter PID yang diperoleh dari kromosom akan disimulasikan untuk mencari *fitness* terbaik berdasarkan fungsi objektif. GA akan mensimulasikan kromosom tiap populasi hingga mencapai generasi yang telah ditentukan. Nilai populasi tiap generasi akan terus berubah karena adanya *crossover* dan mutasi pada tiap generasi sehingga berubah tiap generasi.

Fungsi objektif yang digunakan pada algoritma ini adalah ISE dari sistem pengaturan *level* dan sistem pengaturan *flow*. Tujuan dari fungsi objektif ini adalah meminimalkan kesalahan sistem, sehingga semakin kecil nilai *fitness* individu semakin baik dalam menangani permasalahan.



*Gambar 3.18 Diagram Blok Sistem pengaturan PID-GA*

*(Halaman ini sengaja dikosongkan)*

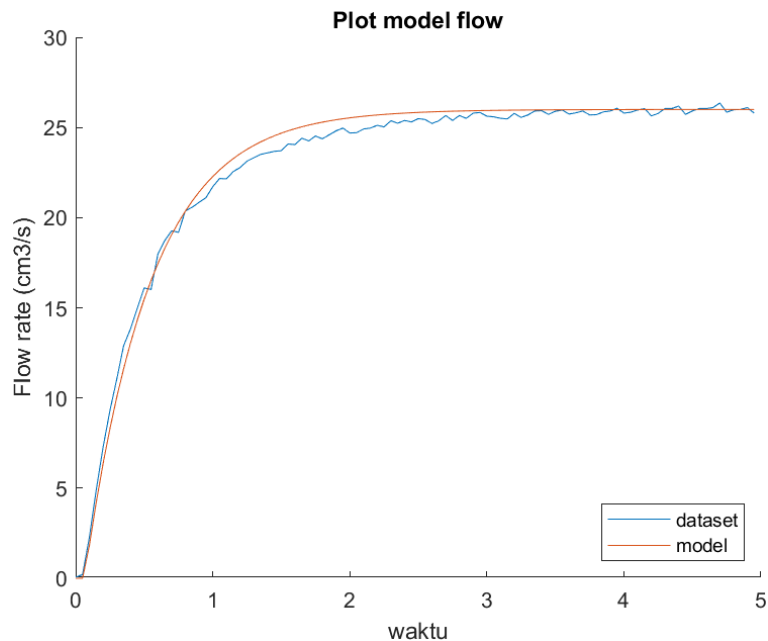
## BAB 4 Hasil dan Pembahasan

### 4.1 Hasil penelitian

#### 4.1.1 Hasil pengujian Model Flow

Berdasarkan hasil pemodelan sistem pada subbab 3.3.3 diperoleh model  $G_{flow}$  sebagai berikut. Hasil pemodelan tersebut perlu dilakukan validasi dengan dibandingkan menggunakan data hasil uji *step respon* untuk mengetahui keakuratan model hasil pendekatan dibandingkan data asli.

$$G_{flow}(s) = \frac{H(s)}{Q_{in}(s)} = \frac{8.67}{0.0312s^2 + 0.564s + 1}$$

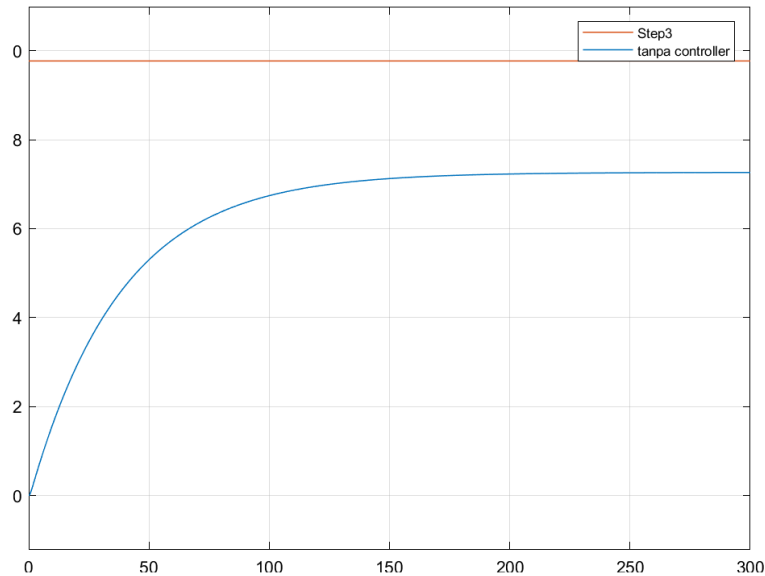


Gambar 4.1 Hasil Validasi Model dan Step Respon Sistem

Pemodelan model *flow* dilakukan mengatur bukaan *control valve* untuk menghasilkan debit tertentu. Pemodelan ini menggunakan konfigurasi tegangan pompa 4 volt dan tegangan *control valve* 15 volt. Hasil *transfer function* menggunakan metode identifikasi *Ziegler Nichols* dibandingkan dengan *dataset* asli dari uji *step respon*. Dataset asli memiliki *time sampling* 0.05 detik sehingga *transfer function* tersebut disimulasikan dengan *time sampling* 0.05 juga. *Transfer function* tersebut disimulasikan dengan *input step* 3 volt karena *control valve* memiliki rentang kerja 12-22 volt. Berdasarkan gambar 4.1 terlihat bahwa hasil pemodelan sudah menyerupai *dataset* asli. Untuk memvalidasi hal tersebut data simulasi dan dataset asli dibandingkan dengan menggunakan perhitungan *mean square error* (MSE) diperoleh nilai mse yaitu 0.2732. Model hasil identifikasi dapat dikatakan valid karena memiliki *mse* yang kecil. Selain itu model hasil identifikasi tersebut hanya bisa digunakan pada tegangan pompa 4 volt dengan tegangan *control valve* 12-22 volt. Jika tegangan pompa berubah maka dianggap model mengalami pembebanan sehingga *transfer function* tersebut dapat berubah.

#### 4.1.2 Hasil Pengujian Respon Sistem Cascade tanpa Kontroler

Pada pengujian kali ini dilakukan dengan menguji sistem secara *close loop* tanpa kontroler atau yang dapat dianalogikan dengan hanya menggunakan gain  $K_p = 1$ . Hal ini dilakukan untuk mengetahui respon sistem secara *close loop*. Adapun respon sistem diperoleh seperti pada gambar 4.2. dari gambar tersebut nampak bahwa sistem memiliki *error steady state* yang besar untuk levelnya. Sehingga perlu dirancang kontroler untuk memperbaiki respon sistem agar sesuai dengan setpoint.



Gambar 4.2 Step Respon Sistem Tanpa Kontroler

#### 4.1.3 Hasil Pengujian Respon Sistem dengan Kontroler PID Ziegler Nichols

Kontroler PID Ziegler-Nichol pada pengujian ini merupakan kontroler hasil desain yang diperoleh pada subbab 3.3.5. Berdasarkan perhitungan pada subbab 3.3.5 diperoleh nilai parameter untuk PID *cascade* seperti pada tabel 4.1. Selain itu juga dihitung nilai performa dari kontroler PID Ziegler-Nichols menggunakan indeks performa ISE untuk sistem pengaturan *flow* dan sistem pengaturan *level plant* yang diperoleh pada tabel 4.1.

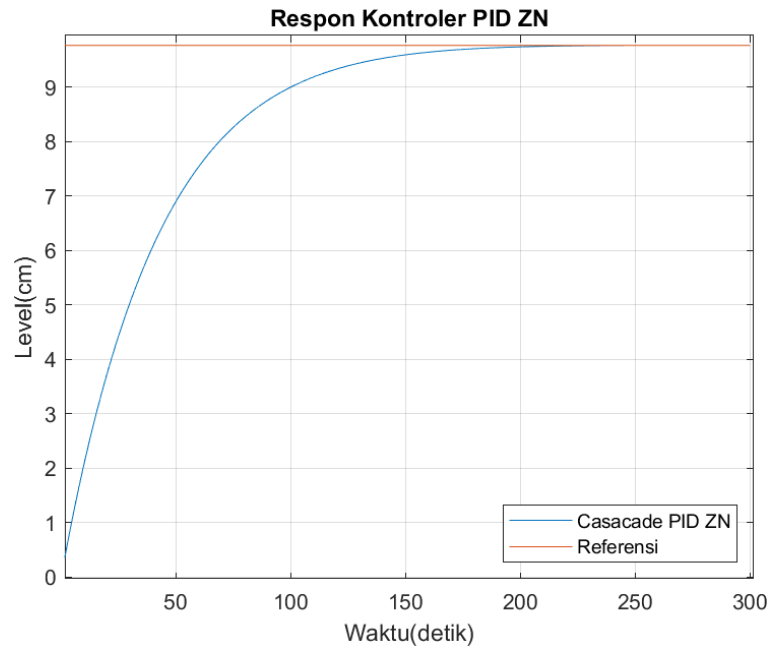
Tabel 4.1 Parameter Kontroler PID Cascade

$K_{p_{out}}$	$K_{i_{out}}$	$K_{d_{out}}$	$K_{p_{in}}$	$T_{i_{in}}$	$T_{d_{in}}$
1.67	0.0108	0	8.88	0.13	0.0325

Tabel 4.2 Nilai ISE kontroler PID ZN

ISE level	ISE Flow
637.6	12.74





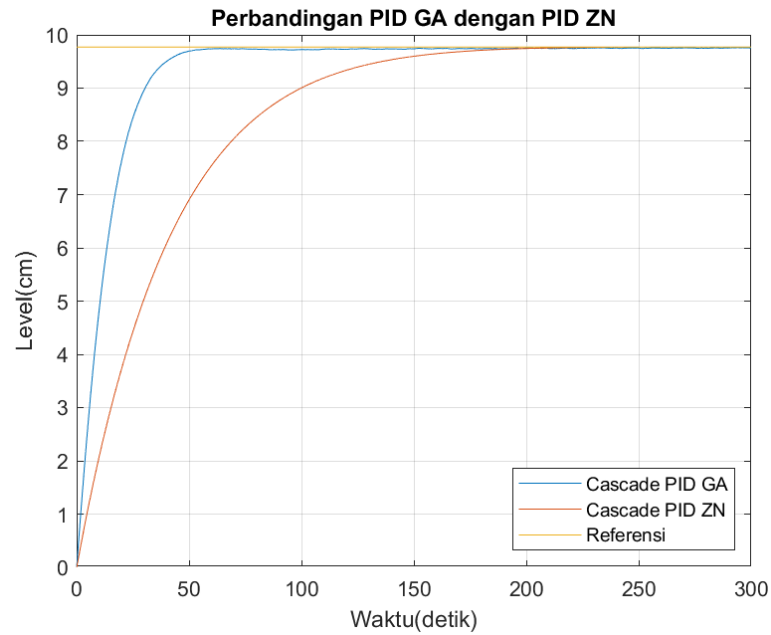
Gambar 4.3 Step Respon Sistem Kontroler PID Ziegler Nichols

berdasarkan hasil simulasi pada gambar 4.3 terlihat bahwa sistem tidak *overshoot* , memiliki steady state level sebesar 9.77, dan sistem memiliki *settling time* (5%) di 117 detik. Hal tersebut sudah memenuhi spesifikasi yang diinginkan berdasarkan subbab 3.3.5. selain itu juga diperoleh nilai indeks performa ISE *level* sebesar 637.6 dan ISE *flow* sebesar 12.74 yang nantinya akan dibandingkan dengan hasil pengujian kontroler PID GA.

#### 4.1.4 Hasil Pengujian Respon Sistem dengan Kontroler PID GA

Kontroler PID-GA merupakan modifikasi dari kontroler PID konvensional dengan menambahkan mekanisme tuning parameter menggunakan GA. Parameter yang diperoleh dari mekanisme tersebut diharapkan merupakan parameter yang optimal berdasarkan indeks performansi yang digunakan yaitu ISE. Pengujian sistem dengan kontroler PID-GA dilakukan untuk menganalisis performa sistem setelah ditambahkan mekanisme *tuning*.

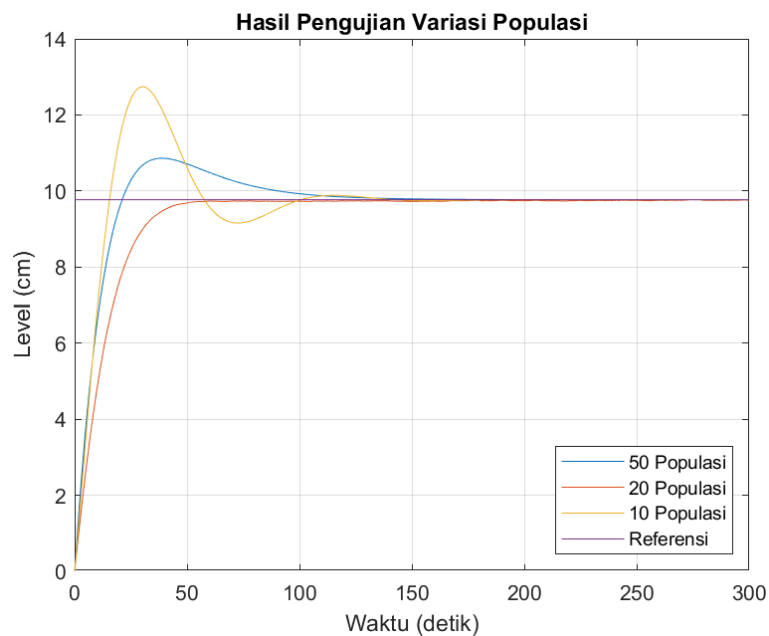
Pada pengujian respon sistem dengan kontroler PID GA juga digunakan konfigurasi *cascade* pada sistem. Adapun konfigurasi awal untuk *tuning* PID GA adalah menggunakan populasi 20 generasi 20 dan probabilitas mutasi 0.25. pada konfigurasi tersebut terlihat bahwa hasil simulasi pada gambar 4.4 sistem memiliki *overshoot* sebesar 0.5% dan memiliki *settling time* sebesar 35 detik adapun untuk konfigurasi tersebut memiliki nilai ISE *flow* sebesar 205.8 dan ISE *level* sebesar 223. Pada pengujian respon sistem dengan kontroler PID GA digunakan beberapa variasi antara lain variasi generasi, populasi, dan juga probabilitas mutasi pada mekanisme tuning PID GA.



*Gambar 4.4 Step Respon Sistem Kontroler PID GA*

#### **4.1.4.1 Variasi Jumlah Populasi**

Pada percobaan ini dilakukan variasi pada jumlah populasi. Yang semula 20 menjadi 10 dan 50 populasi. Berikut merupakan grafik respon sistem yang disajikan pada gambar 4.4. Selain itu juga dapat dihitung tiap performa untuk masing-masing variasi seperti pada tabel 4.3.



*Gambar 4.5 Respon Sistem kontroler PID GA dengan Variasi Populasi*

*Tabel 4.3 Nilai ISE kontroler PID GA dengan Variasi Populasi*

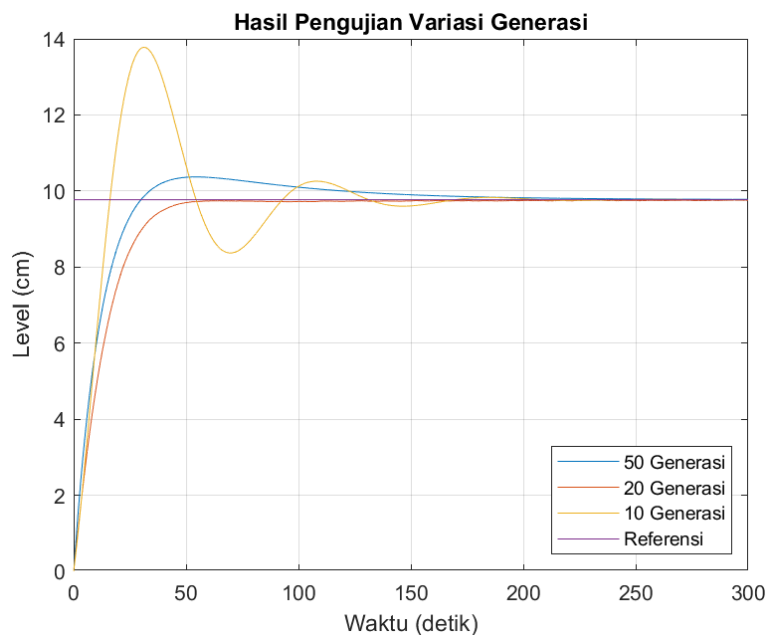
Variasi	ISE level	ISE flow
10	148.7	206.0
20	205.8	222.9

50	84.7	157.0
----	------	-------

Pada pengujian variasi jumlah populasi terlihat pada gambar 4.5. Variasi 10 populasi memiliki indeks performa ISE sebesar 148.7 untuk ISE flow dan 206 untuk ISE level. Variasi 50 populasi memiliki indeks performa ISE sebesar 84.7 untuk ISE flow dan 157 untuk ISE level. Berdasarkan hal tersebut dapat disimpulkan bahwa semakin banyak jumlah populasi semakin baik juga indeks performa pada sistem. Hal tersebut dikarenakan setiap perhitungan generasi akan terpilih populasi terbaik tiap generasi. Semakin banyak jumlah populasi maka akan semakin banyak variasi nilai *fitness* untuk tiap perhitungan populasi. Sehingga semakin cepat untuk menemukan nilai *fitness* yang optimal secara keseluruhan. Akan tetapi semakin banyak generasi semakin lama komputasi untuk mencari hasil optimal.

#### 4.1.4.2 Variasi Jumlah Generasi

Pada percobaan ini dilakukan variasi pada jumlah generasi. Yang semula 20 menjadi 10 dan 50 generasi. Berikut merupakan grafik respon sistem yang disajikan pada gambar 4.5. Selain itu juga dapat dihitung tiap performa untuk masing-masing variasi seperti pada tabel 4.4.



Gambar 4.6 Respon Sistem kontroler PID GA dengan Variasi Generasi

Tabel 4.4 Nilai ISE kontroler PID GA dengan Variasi Generasi

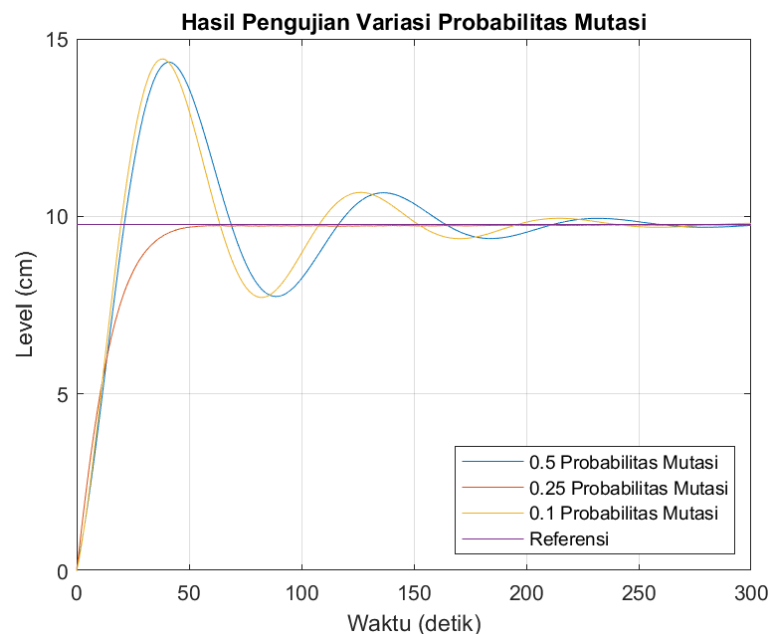
Variasi	ISE level	ISE flow
10	285.9	241.6
20	205.8	222.9
50	119.7	170.7

Pada pengujian variasi jumlah generasi terlihat pada gambar 4.6. Variasi 10 generasi memiliki indeks perform ISE sebesar 285.9 untuk ISE flow dan 241.6 untuk ISE level. Variasi 50 generasi memiliki indeks perform ISE sebesar 119.7 untuk ISE flow dan 170.7 untuk ISE level. Berdasarkan hal tersebut dapat disimpulkan bahwa semakin banyak jumlah generasi semakin baik juga indeks performa pada sistem. Hal

tersebut dikarenakan setiap perhitungan generasi akan terpilih populasi terbaik tiap generasi. Populasi terbaik tersebut nantinya akan menjadi induk untuk menghasilkan populasi yang lebih baik bagi generasi berikutnya. Sehingga semakin banyak generasi akan membuat nilai *fitness value* semakin baik. Akan tetapi semakin banyak generasi semakin lama komputasi untuk mencari hasil optimal.

#### 4.1.4.3 Variasi Probabilitas Mutasi

Pada percobaan ini dilakukan variasi pada jumlah probabilitas mutasi. Yang semula 0.25 menjadi 0.1 dan 0.5 populasi. Berikut merupakan grafik respon sistem yang disajikan pada gambar 4.6. Selain itu juga dapat dihitung tiap performa untuk masing-masing variasi seperti pada tabel 4.5.



Gambar 4.7 Respon Sistem kontroler PID GA dengan Variasi Probabilitas Mutasi

Tabel 4.5 Nilai ISE kontroler PID GA dengan Variasi Probabilitas Mutasi

Variasi	ISE level	ISE flow
0.1	419.1	86.8
0.25	205.8	222.9
0.5	442.5	1180.0

Pada pengujian variasi probabilitas mutasi terlihat pada gambar 4.7. Variasi 0.1 probabilitas mutasi memiliki indeks performa ISE sebesar untuk ISE *flow* 86.8 dan 419.1 untuk ISE *level*. Variasi 50 probabilitas mutasi memiliki indeks perform ISE sebesar 1180 untuk ISE flow dan 442.5 untuk ISE level. Berdasarkan hal tersebut dapat disimpulkan bahwa diperlukan probabilitas mutasi yang sesuai agar mencapai hasil yang terbaik. Semakin besar probabilitas mutasi maka semakin acak data untuk generasi berikutnya karena populasi terbaik generasi sebelumnya semakin besar kemungkinan untuk melakukan mutasi sehingga membuat populasi berikutnya tidak mewakili populasi terbaik. Akan tetapi semakin kecil probabilitas mutasi akan membuat populasi berikutnya menjadi kurang beragam. Sehingga membuat nilai

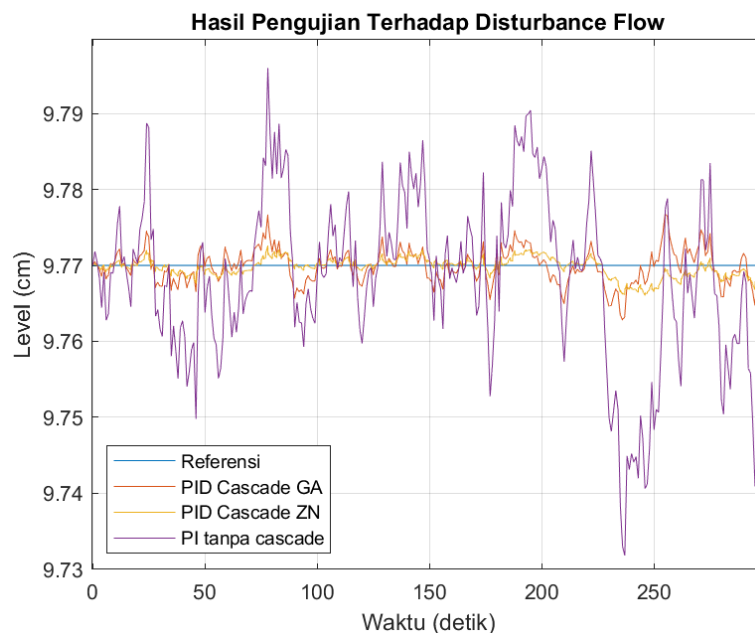
populasi berikutnya hanya berada disekitar nilai dari populasi terbaik generasi sebelumnya.

Berdasarkan berbagai variasi yang telah dilakukan dipilih konfigurasi populasi 20 generasi 20 dan probabilitas mutasi 0.25 untuk sebagai parameter kontroler PID. Hal tersebut dikarenakan meskipun konfigurasi tersebut tidak memiliki indeks performa terbaik, akan tetapi konfigurasi tersebut merupakan yang paling mendekati untuk menghasilkan spesifikasi yang diinginkan yaitu 5% *overshoot*, 0% *error steady state* dan *settling time* < 120 detik.

#### 4.1.5 Hasil Pengujian Pada Disturbance Flow

Pada percobaan ini dilakukan pengujian disturbance. Pengujian disturbance dilakukan untuk mempresentasikan ketika terjadi kesalahan pembacaan pada flow transmitter sehingga berakibat gangguan pada bukaan *control valve* yang mengakibatkan ketinggian level tangki menjadi terganggu. Adapun pengujian digunakan sinyal random gaussian dengan mean 0 dan varians 2 pada pembacaan *flow transmitter* disaat *level* air sudah sesuai referensi ketinggian yang diinginkan yaitu 9.77 cm.

Pengujian dilakukan dengan membandingkan pengujian *disturbance* pada controller PI tanpa *cascade*, pada controller PID *cascade* Ziegler nichols, dan pada controller PID *cascade* GA. Hasil pengujian tersebut diperoleh seperti pada gambar 4.8.



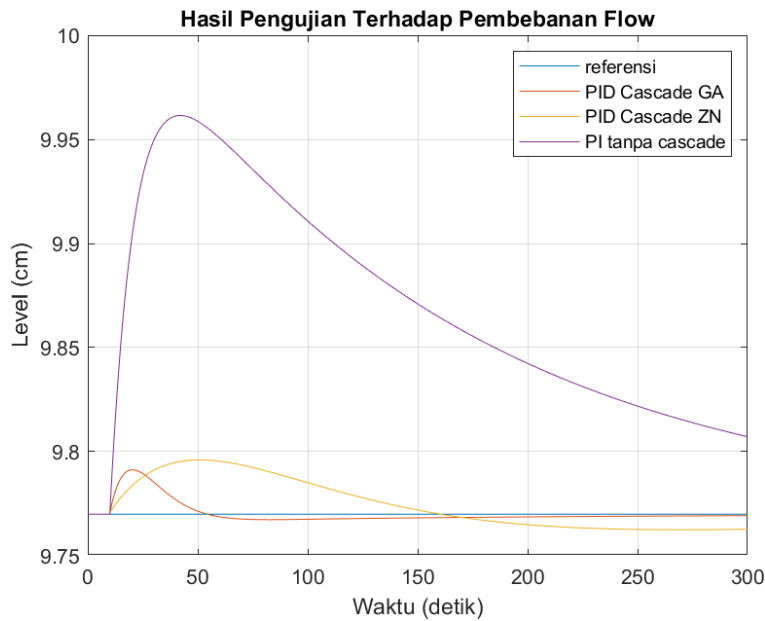
Gambar 4.8 Respon Sistem terhadap Disturbance pada Flow

Berdasarkan gambar 4.8 terlihat bahwa sistem dengan konfigurasi cascade lebih baik dalam menangani disturbance pada pembacaan *flow transmitter*. Hal itu *disturbance* akan tangani terlebih dahulu oleh *inner loop* pada konfigurasi *cascade*. Sehingga tidak langsung berefek terhadap ketinggian tangki. Sedangkan pada konfigurasi tanpa *cascade*, *disturbance* akan ditangani oleh kontroler yang mengatur level tangki. Hal tersebut membuat disturbance akan berdampak lebih signifikan terhadap level pada tangki. Adapun untuk perbandingan kontroler PID ZN dengan PID GA. PID ZN memiliki *error steady state* yang lebih kecil daripada PID GA.

#### 4.1.6 Hasil Pengujian pada Pembebanan Flow

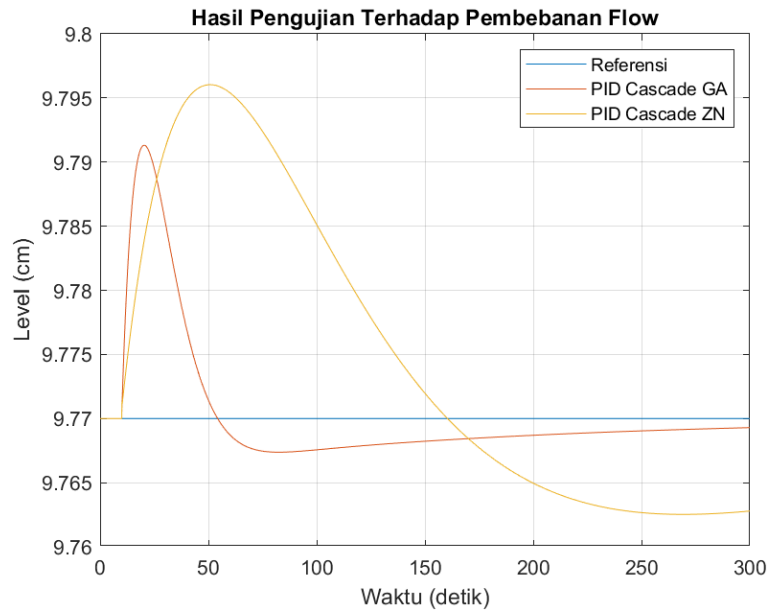
Pada percobaan ini dilakukan pengujian pembebanan. Pengujian pembebanan dilakukan untuk mempresentasikan ketika terjadi perubahan debit *supply* air yang masuk kedalam *control valve*. Adapun pengujian digunakan sinyal step sebesar 5 cm<sup>3</sup>/s yang dimulai pada detik ke 10 disaat *level* air sudah sesuai referensi ketinggian yang diinginkan yaitu 9.77 cm.

Pengujian dilakukan dengan membandingkan pengujian disturbace pada controller PI tanpa *cascade*, pada kontroler PID Ziegler nichols, dan pada PID GA. Hasil pengujian tersebut diperoleh seperti pada gambar 4.9



Gambar 4.9 Respon Sistem terhadap Pembebanan pada Flow

Pada pengujian pengujian pembebanan dibandingkan hasil antara sistem konfigurasi tanpa *cascade* kontroler yaitu hanya menggunakan satu buah PID konvensional dengan sistem konfigurasi *cascade* yang dituning menggunakan PID Ziegler Nichols dan GA. Pengujian pembebanan dilakukan untuk mempresentasikan ketika terjadi perubahan debit *supply* air yang masuk kedalam *control valve*. Dari gambar 4.9 terlihat bahwa sistem dengan konfigurasi *cascade* lebih baik dalam menangani perubahan beban pada debit aliran *input*. Hal itu dikarenakan debit *input* yang masuk kedalam tangki akan dikontrol terlebih dahulu oleh *inner loop* pada konfigurasi *cascade*. Sehingga perubahan debit *input* tidak langsung berefek terhadap ketinggian tangki. Sedangkan pada konfigurasi tanpa *cascade* perubahan debit *input* akan langsung masuk kedalam tangki. Perubahan debit *input* yang mendadak akan menyebabkan perubahan pada *level* tangki. Perubahan *level* tersebut yang nantinya akan di respon oleh kontroler untuk mengatur ulang debit air yang masuk.



*Gambar 4.10 Perbandingan Respon Pembebanan pada PID ZN dan PID GA*

Adapun untuk perbandingan kontroler PID ZN dengan PID GA, berdasarkan gambar 4.10, PID GA memiliki *perturbation peak* yang lebih kecil dari PID GA, serta PID GA memiliki *error recovery* dan *recovery time* yang lebih baik dari PID ZN.

*(Halaman ini sengaja dikosongkan)*



## BAB 5 Kesimpulan dan Saran

### 5.1 Kesimpulan

Kontroler cascade PID-GA mampu memenuhi spesifikasi level air yang diinginkan, yaitu 5% overshoot, 0% *error stedy state* dan *settling time* (5%) < 120 detik. Selain itu kontroler *cascade* PID-GA memiliki respon yang lebih baik dari kontroler cascade PID ZN dimana PID ZN memiliki spesifikasi respon overshoot 0%, *settling time* (5%) pada 117 detik dan 0 *error stedy state* sedangkan PID GA memiliki Spesifikasi respon overshoot 0.5%, *settling time* (5%) pada 35 detik dan 0 *error stedy state*.

Kontroler yang menggunakan konfigurasi *cascade* memiliki respon lebih baik dalam menagani pembebanan dan *disturbance flow* pada sistem. Hal itu diakibatkan gangguan tersebut akan dikoreksi terlebih dahulu oleh kontroler *inner loop* sehingga tidak langsung berdampak pada level tangki. Adapun untuk perbandingan kontroler konfigurasi cascade, kontroler hasil *tuning* PID GA lebih baik daripada PID ZN yang dibuktikan dengan *perturbation*, *recovery time* dan *error recovery* yang lebih baik.

Semakin banyak jumlah populasi dan generasi semakin baik nilai performa ISE dari sistem akan tetapi semakin banyak jumlah populasi dan generasi membuat waktu komputasi menjadi semakin lama. Diperlukan nilai probabilitas mutasi yang tepat agar nilai ISE yang diperoleh menjadi optimal. probabilitas mutasi tidak berpengaruh terhadap waktu komputasi.

### 5.2 Saran

Adapun untuk saran yang diberikan terhadap pengerjaan tugas akhir ini adalah:

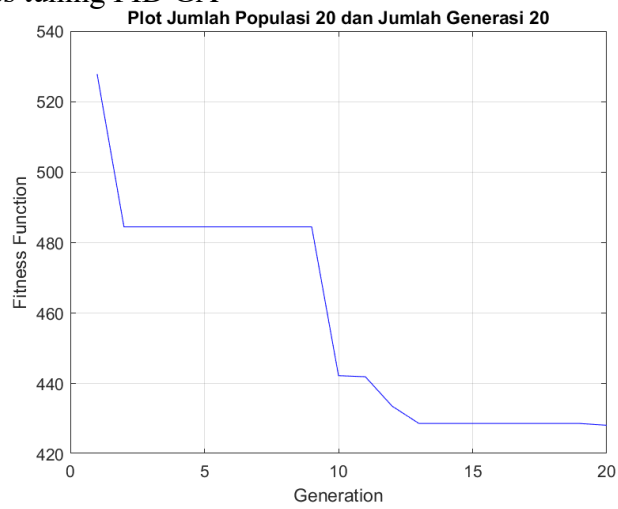
- Pada plant yang digunakan saat ini memiliki kendala modul bawaan untuk *control valve* yang rusak, Sehingga penulis membuat modul sendiri untuk menggerakkan *control valve*. Modul yang telah penulis buat terkadang memiliki masalah ketika digunakan.
- Pada penelitian ini tangki dimodelkan secara linear sehingga model hanya digunakan pada titik kerja tertentu, kedepannya model tangki dapat dimodelkan secara non-linear sehingga model dapat digunakan pada semua rentang kerja.

## DAFTAR PUSTAKA

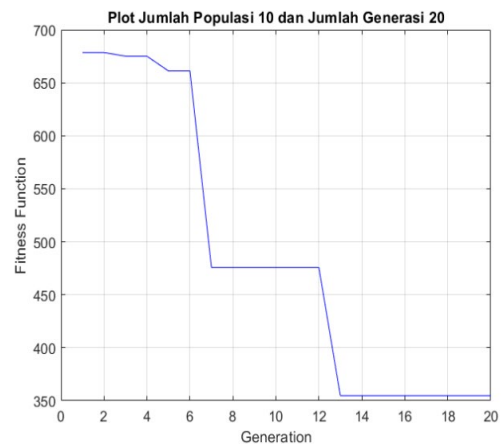
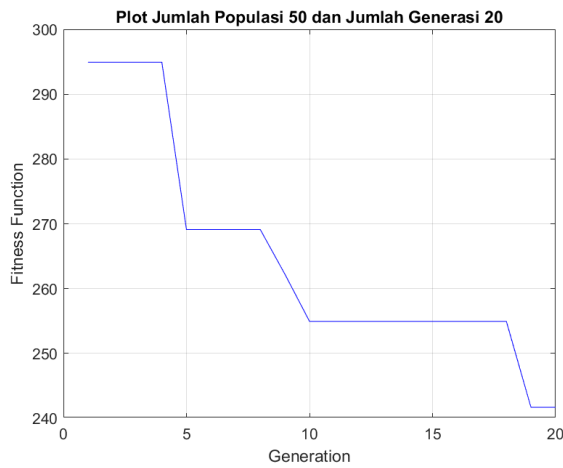
- Bytronics. (2015). *Process Control Technology PCT-100 Datasheet*.  
<http://www.bytronic.net/wp-content/uploads/2015/04/PCT100-PL0214.pdf>
- Devritama, A. (2016). *Analisa Metode Pengaturan Adaptif Untuk Penalaan Otomatis Parameter Kontroler PID Pada Sistem Supervisory Control and Data Acquisition (SCADA)*. Institut Teknologi Sepuluh Nopember.
- Ghufron, M. (2016). *Desain Dan Implementasi Kontroler Self-Tuning PID Dengan Pendekatan Interaksi Adaptif Pada Sistem Pengaturan Level*. Institut Teknologi Sepuluh Nopember.
- Johnson, C. D. (2014). *Process Control Instrumentation Technology*. In *Pearson Prentice Hall, New Jersey: Vol. 8th edition*. <https://lib.hpu.edu.vn/handle/123456789/28701>
- Khuriati R.S., A. (2014). Identifikasi dan Perancangan Pengendali PID Menggunakan Penduga ARX Sistem Pemanas Udara (Halaman 1 s.d. 5). *Jurnal Fisika Indonesia*, 17(51), 1–5.  
<https://doi.org/10.22146/jfi.24424>
- mathworks. (2023). <https://www.mathworks.com/help/control/ug/designing-cascade-control-system-with-pi-controllers.html>
- Melanie, M. (2007). Introduction to genetic algorithms. *Proceedings of GECCO 2007: Genetic and Evolutionary Computation Conference, Companion Material*, 3205–3224.  
<https://doi.org/10.1145/1274000.1274111>
- Ogata, K. (2010). *Modern Control Engineering*(5th Edition). In *Pearson*. New York
- Pritandi, D. A. P. A. (2016). Perancangan Kontroler PID-Fuzzy untuk Sistem Pengaturan Cascade Level dan Flow pada Basic Process Rig 38-100. *Jurnal Teknik ITS*, 5(2).  
<https://doi.org/10.12962/j23373539.v5i2.16179>
- Raharjo, R. D. (2014). Desain Dan Implementasi Kontroler PID Gain Scheduling Untuk Sistem Pengaturan Proses Level Pada Process Control Technology - 100. In *Journal of Chemical Information and Modeling* (Vol. 53, Issue 9).
- Rahmadhani, K. F. (2022). *Particle Swarm Optimization Untuk Sistem Pengaturan Cascade Level Dan Flow Pada Plant PCT-100*. Institut Teknologi Sepuluh Nopember.
- Rizki, R. (2016). *Analisis Perbandingan Supervisory Predictive Control (SPC) Untuk Konfigurasi Cascade Dan Parallel Pada Performansi Sistem Pengaturan Proses Dengan Variabel Waktu Tunda*. Institut Teknologi Sepuluh Nopember.
- STMicroelectronics. (2020). *DS8626, STM32F407xx*. August, 203.
- Tridianto, E., Ariwibowo, T. H., Almasa, S. K., & Prasetya, H. E. G. (2017). Cascaded PID temperature controller for FOPDT model of shell-and-tube heat exchanger based on Matlab/Simulink. *Proceedings IES-ETA 2017 - International Electronics Symposium on Engineering Technology and Applications*, 2017-Decem, 185–191.  
<https://doi.org/10.1109/ELECSYM.2017.8240400>
- Yusuf, M., Fatoni, A., & Hady, M. A. (2017). Desain Kontroler PID-Genetic Algorithm untuk Sistem Pengaturan Level Air Steam Drum pada Pembangkit Listrik Tenaga Uap (PLTU). In *Jurnal Teknik ITS* (Vol. 6, Issue 1). <https://doi.org/10.12962/j23373539.v6i1.21391>

## LAMPIRAN

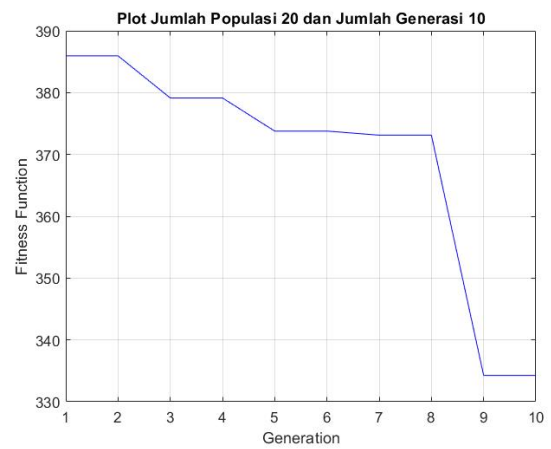
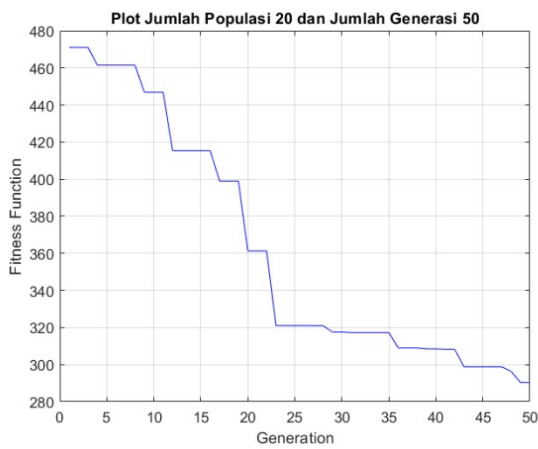
### Lampiran 1 Plot Fitness tuning PID GA



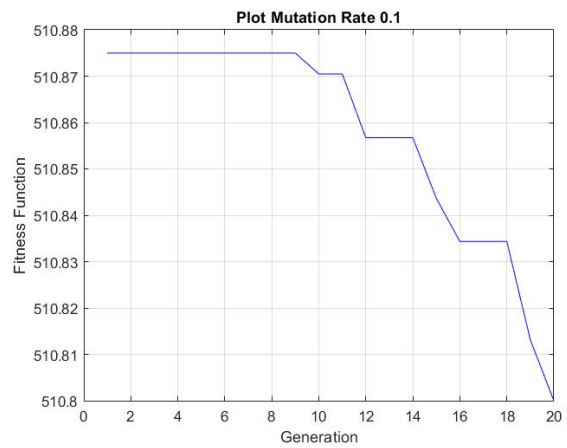
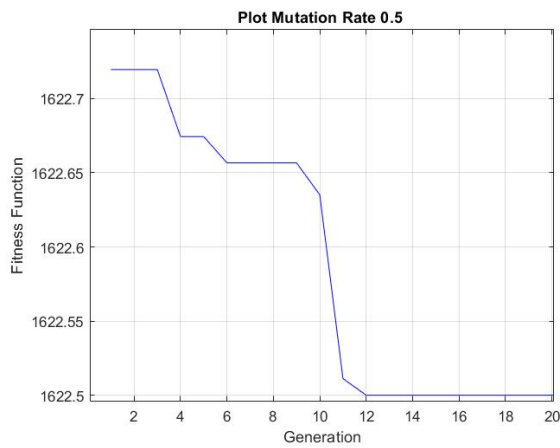
Plot fitness PID GA dengan populasi 20 generasi 20 mutrate 0.25



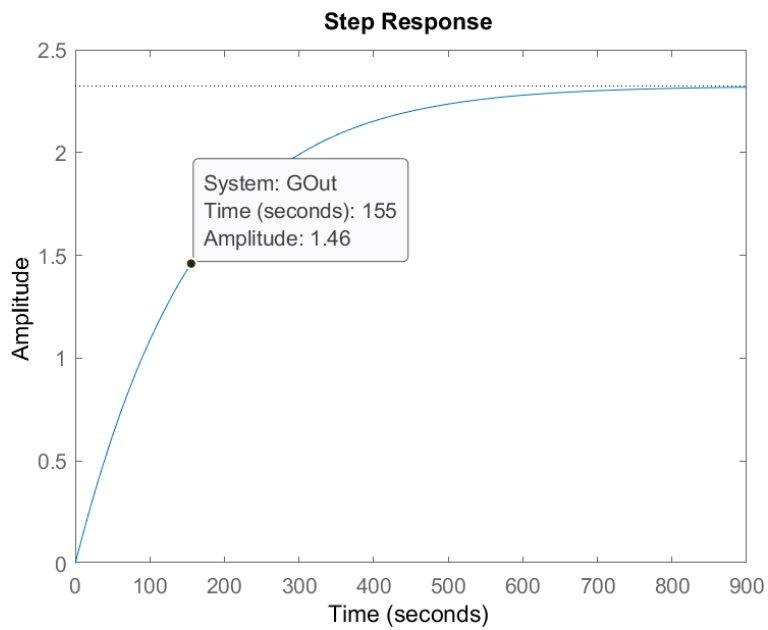
Plot fitness PID GA dengan Variasi Populasi



Plot fitness PID GA dengan Variasi Generasi



Plot fitness PID GA dengan Variasi Mutation Rate



Penentuan time konstant Gout

## Lampiran 2 program Tuning kontroler PID cascade Matlab

```
% Parameter GA
LB = [0 0 0 0 0 0];
UB = [10 1 1 10 1 1];
nVars = 6;
generation = 50;
populationSize = 20;
mutRate = 0.25;
crossoverTimes = 2;
mutationTimes = 2;

% Inisiasi populasi acak
population = zeros(populationSize, nVars);
tempPopulation = zeros(populationSize, nVars);
sPopulation = zeros(populationSize, nVars+1);
F = zeros(populationSize, 1);
for i=1:1:populationSize
    for n=1:1:nVars
        population(i, n) = unifrnd(LB(1,n), UB(1,n));
    end
    F(i) = fitness(population(1,:));
end

% Sorting of the population
sPopulation(:, 1:nVars) = population(:, :);
sPopulation(:, nVars+1) = F;

%% Selection dan Crossover tiap Generation
for ii=1:1:generation
    k=1;

    % Elitism
    tempPopulation(k, 1:nVars) = sPopulation(1,1:nVars);
    k=k+1;

    % Parent Selection
    for j=1:1:crossoverTimes
        y1(j) = geornd(0.1)+1;
        while y1(j) > populationSize
            y1(j) = geornd(0.1)+1;
        end
        y2(j) = geornd(0.1)+1;
        while y2(j) > populationSize
            y2(j) = geornd(0.1)+1;
        end
    end

    % Arithmetic Crossover
    for u=1:1:crossoverTimes
```

```

    parent1 = sPopulation(y1(u), 1:nVars);
    parent2 = sPopulation(y2(u), 1:nVars);
    [children] = arithmeticCrossover(parent1, parent2);
    tempPopulation(k, 1:nVars) = children(1,:);
    tempPopulation(k, 1:nVars) = max(tempPopulation(k, 1:nVars), LB);
    tempPopulation(k, 1:nVars) = min(tempPopulation(k, 1:nVars), UB);

    k=k+1;
    tempPopulation(k, 1:nVars) = children(2,:);
    tempPopulation(k, 1:nVars) = max(tempPopulation(k, 1:nVars), LB);
    tempPopulation(k, 1:nVars) = min(tempPopulation(k, 1:nVars), UB);
    k=k+1;
end

% Mutation
for e=1:1:mutationTimes
    parent = sPopulation(unidrnd(populationSize), 1:nVars);
    [child] = geneMutation(parent, mutRate, LB, UB);

    tempPopulation(k, 1:nVars) = child;
    tempPopulation(k, 1:nVars) = max(tempPopulation(k, 1:nVars), LB);
    tempPopulation(k, 1:nVars) = min(tempPopulation(k, 1:nVars), UB);
    k = k+1;
end
for k=k:1:populationSize
    replicatedChild = sPopulation(randi([1 populationSize]), 1:nVars);
    tempPopulation(k, 1:nVars) = replicatedChild;
    k = k+1;
end

% Menghitung Fitness
for iii=1:1:populationSize
    F(iii) = fitness(tempPopulation(iii,:));
end
sPopulation(:, 1:nVars) = tempPopulation;
sPopulation(:, nVars+1) = F(:,:);
sPopulation = sortrows(sPopulation, nVars+1);
bestF(ii) = sPopulation(1, nVars+1);

%% Individu Terbaik
Xmin = sPopulation(1,1:nVars);
end

%Plot fitness individu terbaik tiap generasi
Fval = bestF(1, generation);
figure(3);
plot(bestF, 'b');
title("Plot Jumlah Populasi "+populationSize+" dan Jumlah Generasi "+
generation)
xlabel('Generation');
ylabel('Fitness Function');
grid on;

```

### Fungsi evaluasi Fitness

```
function cost = fitness(x)
assignin('base','x',x);
sim('GAsim.slx');
cost=iseLevel(length(iseLevel))+iseFlow(length(iseFlow))
end
```

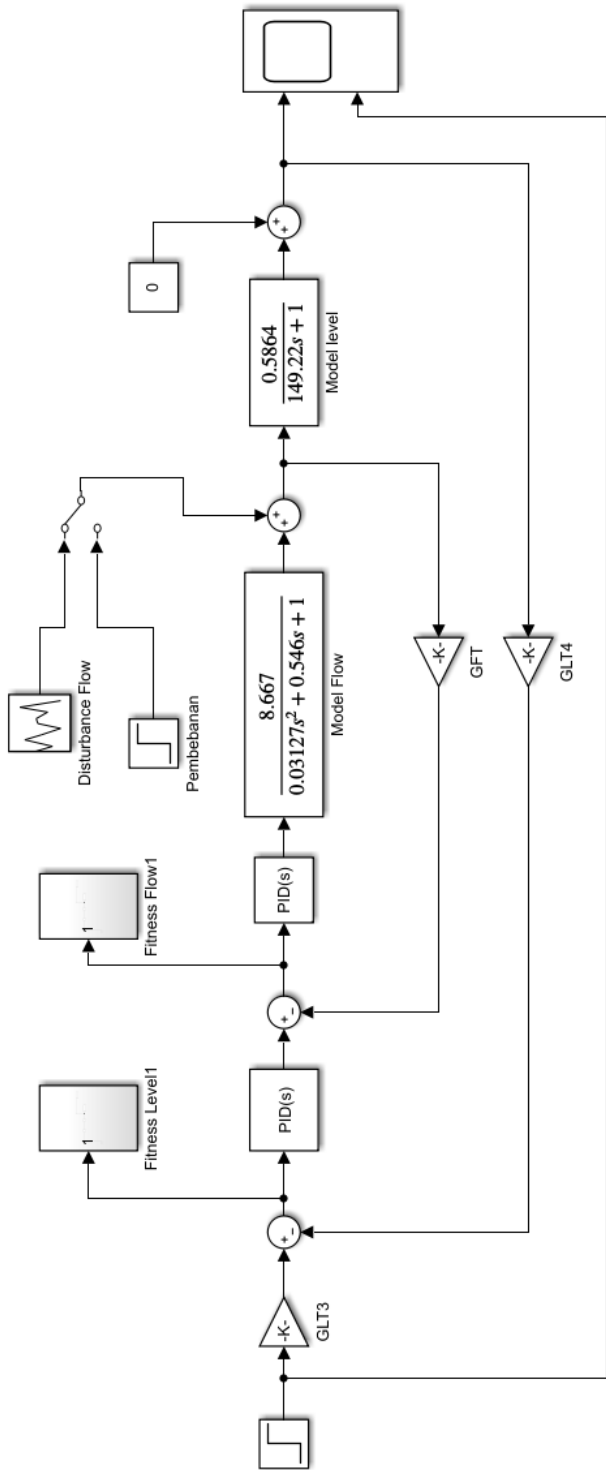
### Fungsi Mutasi

```
function [child] = geneMutation(parent, mutRate, LB, UB)
nVars = numel(parent);
nmu = ceil(mutRate * nVars);
j = randsample(nVars, 1);
child = parent;
child(j) = parent(j) - rand*parent(j);
end
```

### Fungsi crossover

```
function [children] = arithmeticCrossover(parent1, parent2)
alpha = rand;
child1 = alpha*parent1 + (1-alpha)*parent2;
child2 = (1-alpha)*parent1 + alpha*parent2;
children = [child1; child2];
end
```

Lampiran 3 Diagram Blok kontroler PID cascade Matlab





## BIODATA PENULIS



Hakhi Gya Yektianto lahir di Lamongan, 23 Desember 2000, merupakan anak pertama dari dua bersaudara. Penulis telah menempuh pendidikan formal yaitu di, SDN 1 Cerme, SMPN 1 Cerme dan SMAN 1 Gresik. Setelah lulus dari SMA tahun 2019, Penulis mengikuti SNMPTN dan diterima di Departemen Teknik Elektro FTEIC - ITS pada tahun 2019 dan terdaftar dengan NRP 07111940000022.

Di Departemen Teknik Elektro Penulis aktif sebagai Asisten Laboratorium Sistem dan Sibernetika. Selain itu penulis juga mengikuti kegiatan Studi Independen Bangkit 2022 tentang machine learning. Selain itu penulis juga pernah mengikuti PKM.