

TUGAS AKHIR - EE184801

**PERANCANGAN SISTEM KONTROL LEVEL
KETINGGIAN AIR MENGGUNAKAN PID MODEL
REFERENCE ADAPTIVE CONTROL (MRAC) PADA
PLANT PCT-100**

MUHAMMAD USAMA SAHAL

NRP 07111940000176

Dosen Pembimbing

Ir. Ali Fatoni, M.T.

NIP 196206031989031002

Mohamad Abdul Hady, S.T, M.T

NIP 198904132015041002

Program Studi Sarjana Teknik Elektro

Departemen Teknik Elektro

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya

2023



TUGAS AKHIR - EE184801

**PERANCANGAN SISTEM KONTROL LEVEL
KETINGGIAN AIR MENGGUNAKAN PID MODEL
REFERENCE ADAPTIVE CONTROL (MRAC) PADA
PLANT PCT-100**

MUHAMMAD USAMA SAHAL

NRP 07111940000176

Dosen Pembimbing

Ir. Ali Fatoni, M.T.

NIP 196206031989031002

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

NIP 198904132015041002

Program Studi Sarjana Teknik Elektro

Departemen Teknik Elektro

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya

2023



FINAL PROJECT - EE184801

DESIGN OF A WATER LEVEL CONTROL SYSTEM USING PID MODEL REFERENCE ADAPTIVE CONTROL (MRAC) ON PLANT PCT-100

MUHAMMAD USAMA SAHAL

NRP 07111940000176

Advisor

Ir. Ali Fatoni

NIP 196206031989031002

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

NIP 198904132015041002

Study Program Bachelor of Electrical Engineering

Department of Electrical Engineering

Faculty of Intelligent Electrical and Informatics Technology

Institut Teknologi Sepuluh Nopember

Surabaya

2023

LEMBAR PENGESAHAN

PERANCANGAN SISTEM KONTROL LEVEL KETINGGIAN AIR MENGGUNAKAN PID MODEL REFERENCE ADAPTIVE CONTROL (MRAC) PADA PLANT 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: MUHAMMAD USAMA SAHAL

NRP. 07111940000176

Disetujui oleh Tim Penguji Tugas Akhir:

1. Ir. Ali Fatoni, M.T.

Pembimbing



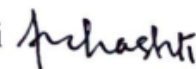
2. Mohamad Abdul Hady, S.T., M.T.

Ko-pembimbing



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

Penguji



4. Eka Iskandar, S.T., M.T.

Penguji



5. Yurid Eka Nugraha, S.T., M.Eng., Ph.D.

Penguji



SURABAYA

Juli, 2023

Halaman ini sengaja dikosongkan

APROVAL SHEET

DESIGN OF A WATER LEVEL CONTROL SYSTEM USING PID MODEL REFERENCE ADAPTIVE CONTROL (MRAC) ON PLANT PCT-100

FINAL PROJECT


Submitted to fulfill one of the requirements
for obtaining a Bachelor Engineering degree at
Undergraduate Program of Electrical Engineering
Department of Electrical Engineering
Faculty of Intelligent Electrical and Informatics Technology
Sepuluh Nopember Institute of Technology

Oleh: MUHAMMAD USAMA SAHAL


NRP. 07111940000176

Disetujui oleh Tim Penguji Tugas Akhir:


1. Ir. Ali Fatoni, M.T.

Advisor 


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

Co-Advisor 

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

Examiner 

4. Eka Iskandar, S.T., M.T.

Examiner 

5. Yurid Eka Nugraha, S.T., M.Eng., Ph.D.

Examiner 

SURABAYA

Juli, 2023

Halaman ini sengaja dikosongkan

PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini:

Nama mahasiswa / NRP : Muhammad Usama Sahal / 07111940000176
Program studi : Teknik Sistem Pengaturan
Dosen Pembimbing / NIP : Ir. Ali Fatoni, M.T. / 196206031989031002
Mohamad Abdul Hady, S.T., M.T. /
198904132015041002

dengan ini menyatakan bahwa Tugas Akhir dengan judul "*Perancangan Sistem Kontrol Level Ketinggian Air Menggunakan PID Model Reference Adaptive Control (MRAC) pada Plant 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, 18 Juli 2023

Mengetahui
Dosen Pembimbing 1




Ir. Ali Fatoni, M.T.
NIP. 196206031989031002

Mengetahui
Dosen Pembimbing 2



Mohamad Abdul Hady, S.T., M.T.
NIP 198904132015041002

Mahasiswa



Muhammad Usama Sahal
NRP. 07111940000176

Halaman ini sengaja dikosongkan

ABSTRAK

PERANCANGAN SISTEM KONTROL LEVEL KETINGGIAN AIR MENGGUNAKAN PID MODEL REFERENCE ADAPTIVE CONTROL (MRAC) PADA PLANT PCT-100

Nama Mahasiswa / NRP : Muhammad Usama Sahal / 07111940000176
Departemen : Teknik Elektro FTEIC - ITS
Dosen Pembimbing : 1. Ir. Ali Fatoni, M.T.
2. Mohamad Abdul Hady, S.T., M.T.

Abstrak

PCT-100 merupakan miniatur sistem kontrol proses pada dunia industri dan merupakan produk dari vendor *Bytronic*. Perangkat lunak yang mendukung PCT-100 mempunyai fasilitas untuk melakukan kontrol terhadap empat buah variabel kontrol diantaranya debit, level, suhu, dan tekanan. Permasalahan kontrol level pada *plant PCT-100* dengan variasi kondisi beban, dapat diselesaikan dengan menggunakan sistem kontrol adaptif. Pada penelitian sebelumnya yang telah dilakukan kontrol level pada *plant PCT-100* dengan menerapkan metode *selftuning PID*, menggunakan pendekatan interaksi adaptif. Namun, Pada tugas akhir ini, dilakukan perancangan *PID Model Reference Adaptive Control (MRAC)* untuk kontrol level yang merupakan skema dari sistem kontrol adaptif. Berdasarkan hasil simulasi dengan parameter PCT-100 yang ditentukan didapatkan untuk pengaturan level nilai parameter *adaptive rate* yaitu nilai $\gamma_p = 1 \times 10^3$, $\gamma_i = 1 \times 10$, dan $\gamma_d = 1 \times 10^2$ untuk pengujian tracking set point menghasilkan nilai RMSE sebesar 0.0685. Pada pengujian dengan beban tanpa noise menghasilkan nilai RMSE 0.0681, kemudian dengan white noise dan random uniform nilai RMSE 0.0810 dan 0.0823. Hasil simulasi PID-MRAC lebih baik dibandingkan dengan kontroler PID karena kontroler PID-MRAC memiliki kemampuan adaptasi nilai parameter K_p, K_i , dan K_d sehingga semakin banyak error maka akan semakin membuat performansi PID-MRAC semakin bagus dibandingkan simulasi lain.

Kata kunci: PCT-100, PID, Sistem Kontrol Adaptif, MRAC.

Halaman ini sengaja dikosongkan

ABSTRACT

DESIGN OF A WATER LEVEL CONTROL SYSTEM USING PID MODEL REFERENCE ADAPTIVE CONTROL (MRAC) ON PLANT PCT-100

Student Name / NRP : Muhammad Usama Sahal/07111940000176
Department : Electrical Engineering FTEIC - ITS
Advisor : 1. Ir. Ali Fatoni, M.T.
2. Mohamad Abdul Hady, S.T., M.T.

Abstract

PCT-100 is a miniature process control system in the industrial world and is a product from the vendor Bytronic. Software that supports the PCT-100 has the facility to control four control variables including discharge, level, temperature, and pressure. The problem of level control at the PCT-100 plant with variations in load conditions can be solved by using an adaptive control system. In previous research, level control was carried out at the PCT-100 plant by applying the PID self-tuning method, using an adaptive interaction approach. However, in this final project, the design of the PID Reference Model Adaptive Control (MRAC) is carried out for level control which is a scheme of an adaptive control system. Based on the simulation results with the specified PCT-100 parameter, it is obtained for setting the level the adaptive rate parameter values are $\gamma_p = 1 \times 10^3$, $\gamma_i = 1 \times 10$, dan $\gamma_d = 1 \times 10^2$ for testing the tracking set point produces an RMSE value of 0.0685. In testing with a load without noise it produces an RMSE value of 0.0681, then with white noise and random uniform RMSE values of 0.0810 and 0.0823. The results of the PID-MRAC simulation are better than the PID controller because the PID-MRAC controller has the ability to adapt the parameter values K_p , K_i , and K_d so that the more errors, the better the performance of the PID-MRAC compared to other simulations.

Keywords: PCT-100, PID, Adaptive Control System, MRAC.

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Allah SWT atas ridhonya penulis dapat menyelesaikan penyusunan skripsi ini. Adapun judul Tugas Akhir yang penulis ajukan adalah ***“Perancangan Sistem Kontrol Level Ketinggian Air Menggunakan PID Model Reference Adaptive Control (MRAC) pada Plant PCT-100”*** Skripsi ini diajukan untuk memenuhi syarat kelulusan mata kuliah Tugas Akhir di Fakultas Teknologi Elektro dan Informatika Cerdas Institut Teknologi Sepuluh Nopember. Tidak dapat disangkal bahwa butuh usaha yang keras dalam penyelesaian pengerjaan skripsi ini. Namun, karya ini tidak akan selesai tanpa orang-orang tercinta di sekeliling penulis yang mendukung dan membantu. Terima kasih penulis sampaikan kepada:

1. Kedua orang tua beserta kakak adik penulis yang sudah menyemangati penulis dalam pengerjaan tugas akhir ini, serta memberikan doa kepada penulis.
2. Bapak Ir. Ali Fatoni, M.T. dan Bapak Mohamad Abdul Hady, S.T., M.T. selaku Dosen Pembimbing Tugas Akhir yang telah membimbing dalam menyelesaikan Tugas Akhir.
3. Ibu Dr. Trihastuti Agustinah, S.T., M.T. selaku dosen wali penulis selama berkuliah di Departemen Teknik Elektro, FTEIC, ITS yang telah membantu proses dalam proses perkuliahan.
4. Bapak dan Ibu Dosen bidang studi Sistem Pengaturan yang telah memberikan materi perkuliahan yang berguna untuk penulis Menyusun tugas akhir ini.
5. Bapak dan Ibu Dosen Departemen Teknik Elektro yang telah memberikan materi perkuliahan selama kurang lebih 8 semester, serta memberikan saran di dalam ujian Tugas Akhir.
6. Teman-teman grup Hidup Sehat, Muhammad Faris Zuhairi, Hakhi Gya Yektianto, pasangan penulis Waseda Himawari yang memberikan bantuan informasi maupun dukungan untuk menyelesaikan tugas akhir ini.
7. Keluarga besar asisten Laboratorium Sistem dan Sibernetika serta asisten Laboratorium Kontrol dan Otomasi yang telah membantu dan memberi dukungan serta memfasilitasi penelitian tugas akhir ini.
8. Semua pihak yang telah membantu dan tidak dapat disebutkan satu persatu.

Dalam menyelesaikan tugas akhir ini, penulis sadar bahwa hasil yang telah dikerjakan masih belum sempurna dan masih banyak hal yang bisa diperbaiki. Saran, kritik, dan masukan dari seluruh pihak sangat diharapkan penulis untuk kemungkinan mengembangkan penelitian ini lebih lanjut. Besar harapan penulis bahwa buku Tugas Akhir ini dapat memberikan informasi dan manfaat bagi pembaca pada umumnya dan mahasiswa Fakultas Teknologi Elektro dan Informatika Cerdas pada khususnya.

Surabaya, Juli 2023

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

LEMBAR PENGESAHAN	i
APPROVAL SHEET	iii
PERNYATAAN ORISINALITAS	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvi
DAFTAR SIMBOL	xvii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	1
1.3 Batasan Masalah	2
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 <i>Plant PCT-100</i>	3
2.2.1.1 Process Rig	4
2.2.1.2 Control Module	5
2.2.2 <i>Pemodelan Plant PCT-100</i>	6
2.2.2.1 Pemodelan Level Transmitter	8
2.2.2.2 Pemodelan Tangki [10]	9
2.2.3 Identifikasi Sistem [11]	10
2.2.4 Kontroler PID MRAC	11
2.2.4.1 MIT Rule	12
2.2.4.2 Kontroler PID	12
2.2.5 Root Mean Square Error (RMSE)	13
BAB 3 METODOLOGI	15
3.1 Metode Penelitian	15
3.2 Bahan dan peralatan yang digunakan	16
3.2.1 <i>Process Control Technology 100</i>	16
3.2.2 <i>Data Acquisition Menggunakan STM32</i>	16

3.2.3	ADAM 5000 L TCP/IP.....	17
3.2.4	Aplikasi Matlab	17
3.2.5	Komputer atau Laptop	17
3.3	Urutan pelaksanaan penelitian	18
3.3.1	Pemodelan Sistem Pengaturan Level.....	18
3.3.1.1	Pemodelan Level Transmitter.....	18
3.3.1.2	Pemodelan Tangki.....	19
3.3.2	Pemodelan Sistem Pengaturan Flow	20
3.3.3	Perancangan Kontroler PID <i>MRAC</i>	22
3.3.3.1	Penentuan Model Referensi.....	22
3.3.3.2	Perancangan Mekanisme Adaptasi MIT Rule	23
3.3.3.3	Normalisasi Parameter	24
BAB 4	Hasil dan Pembahasan	27
4.1	Hasil Penelitian.....	27
4.1.1	Pengujian <i>Tracking Set Point</i>	27
4.1.1.1	Tanpa Kontroler	27
4.1.1.2	Kontroler PID	27
4.1.1.3	PID <i>MRAC</i>	28
4.1.2	Pengujian Perubahan Beban Tanpa <i>Noise</i>	30
4.1.2.1	Kontroler PID	30
4.1.2.2	PID <i>MRAC</i>	30
4.1.3	Pengujian Perubahan Beban Dengan <i>Noise</i>	31
4.1.3.1	Kontroler PID	32
4.1.3.2	PID <i>MRAC</i>	32
4.2	Pembahasan	33
4.2.1	Tanpa Kontroler.....	33
4.2.2	Kontroler PID	33
4.2.3	Kontroler PID <i>MRAC</i>	33
BAB 5	Kesimpulan dan Saran	35
5.1	Kesimpulan.....	35
5.2	Saran	35
	DAFTAR PUSTAKA	37
	LAMPIRAN	39
I.	Pemodelan Flow Meter.....	39
II.	Percobaan Menentukan Adaptive Rate.....	41
III.	Simulink PID <i>MRAC</i>	43
	BIODATA PENULIS	45

DAFTAR GAMBAR

Gambar 2. 1 Process Rig PCT-100	4
Gambar 2. 2 Control Module	6
Gambar 2. 3 Sistem Level Plant PCT-100	7
Gambar 2. 4 Diagram Blok Pemodelan Sistem Pengaturan Level	7
Gambar 2. 5 Posisi Floating Ball Pada Proses Tangki	8
Gambar 2. 6 Vítečková's Curve	11
Gambar 2. 7 Blok Diagram PID MRAC Konvensional [13]	12
Gambar 3. 1 Flowchart Diagram Alur Penelitian	15
Gambar 3. 2 Process Kontrol Technology 100	16
Gambar 3. 3 Mikrokontroler STM32F407VG	16
Gambar 3. 4 ADAM 5000L/TCP	17
Gambar 3. 5 Respon Tegangan Terhadap Ketinggian Air	18
Gambar 3. 6 Parameter Fisis Tiap Kondisi	19
Gambar 3. 8 Kurva Respon Flow	20
Gambar 3. 9 Nilai t_{33} dan t_{70} Menggunakan vítečková Terhadap Kurva Respon Flow	21
Gambar 3. 10 Kurva Respon Flow Dibanding Kurva Model	22
Gambar 3. 11 Mekanisme Pegaturan Level Pada Plant PCT-100 Dengan PID MRAC	22
Gambar 4. 1 Respon Perubahan Set Point Level Tanpa Kontroler	27
Gambar 4. 2 Respon Perubahan Set Point Level Dengan Kontroler PID	28
Gambar 4. 3 Respon Tracking Set Point Level Dengan PID MRAC	29
Gambar 4. 4 Respon Perubahan Beban Level Dengan PID	30
Gambar 4. 5 Respon Beban 1 Level Dengan PID MRAC	31
Gambar 4. 6 Sinyal <i>White Noise</i>	31
Gambar 4. 7 Sinyal <i>Random Uniform</i>	31
Gambar 4. 8 Respon PID Dengan Beban dan <i>White Noise</i>	32
Gambar 4. 9 Respon PID Dengan Beban dan <i>Random Uniform</i>	32
Gambar 4. 10 Respon PID MRAC Dengan Beban dan <i>White Noise</i>	32
Gambar 4. 11 Respon PID MRAC Dengan Beban dan <i>Random Uniform</i>	32

DAFTAR TABEL

Tabel 2. 1 Komponen Penyusun <i>Process Rig</i>	5
Tabel 2. 2 Parameter Model Plant PCT-100	8
Tabel 2. 3 Karakteristik Kontroler PID	13
Tabel 3. 1 Spesifikasi ADAM 5000L/TCP	17
Tabel 3. 2 Spesifikasi Laptop yang Digunakan.....	18
Tabel 4. 2 Hasil Pengukuran Uji Perubahan Set Point Dengan PID	28
Tabel 4. 3 Variasi Parameter Kontroler PID MRAC	29
Tabel 4. 4 Hasil pengukuran Uji Berubahan Set Point Dengan PID MRAC	29
Tabel 4. 5 Hasil Pengukuran Uji Perubahan Beban Dengan PID	30
Tabel 4. 6 Hasil Pengukuran Uji Perubahan Beban Dengan PID MRAC.....	31
Tabel 4. 7 Hasil Pengukuran Uji Perubahan Beban Menggunakan PID Dengan Noise	32
Tabel 4. 8 Hasil Pengukuran Uji Perubahan Beban Menggunakan PID MRAC Dengan Noise....	33
Tabel 4. 9 Hasil Pengukuran Uji Perubahan Beban Dengan PID	33
Tabel 4. 10 Hasil Pengukuran Uji Dengan PID MRAC.....	34

DAFTAR SIMBOL

A	:	Luas alas tangki (cm^2)
a	:	Diameter Pipa (cm)
C	:	Kapasitansi tangki (cm^2)
e	:	Error
H	:	Ketinggian air dalam tangki (cm)
h_0	:	Ketinggian air saat kondisi di titik operasi (cm)
K_d	:	Konstanta derivatif pada kontroler
K_i	:	Konstanta integral pada kontroler
K_p	:	Konstanta proporsional pada kontroler
L	:	Panjang Pipa (cm)
Q	:	Debit air dalam tangki (cm^3)
Q_{in}	:	Debit air yang masuk ke tangki (cm^3/s)
Q_{out}	:	Debit air yang keluar dari tangki (cm^3/s)
Q_0	:	Laju aliran keluar saat kondisi di titik operasi (cm^3/s)
R	:	Resistansi control valve (s/cm^2)
T_d	:	Time derivatif
T_i	:	Time integral
y	:	Keluaran model plant (cm)
y_m	:	Keluaranmodel referensi (cm)
θ	:	Vektor parameter kontroler [K_p K_i K_d]
ζ	:	Rasio Redaman
α	:	Penguatan model referensi
ω_n	:	Frekuensi alami (deg/s)
γ	:	Vektor adaptive rate [γ_p γ_i γ_d]
γ_d	:	Adaptive rate kontroler derivatif
γ_i	:	Adaptive rate kontroler integral
γ_p	:	Adaptive rate kontroler proporsional

Halaman ini sengaja dikosongkan

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Level merupakan salah satu variabel kontrol yang hampir ada di setiap industri. Sistem pengaturan level mempunyai karakteristik *timevarying* dan *nonlinear* pada industri skala besar [1]. Berbagai metode kontrol sudah diterapkan dalam sistem pengaturan level seperti kontroler PID dan *fuzzy*-PID keduanya belum mampu memberikan hasil yang presisi [1]. Kombinasi PID *neural network* membutuhkan waktu yang lama untuk proses *learning* [2]. Kombinasi PID dengan metode adaptif memberikan hasil yang baik ketika digunakan dalam sistem dengan *plant* dinamik [1][3]. Metode adaptif yang sudah sering diterapkan seperti *Self-Tuning PID-Genetic Algorithm* masih menggunakan sistem yang cukup kompleks seperti adanya identifikasi sistem sebelum melakukan proses adaptasi [4][5].

Sampai saat ini, kontroler PID masih mampu menghasilkan performansi sistem sesuai dengan keinginan ketika *di-tuning* dengan benar. Kata *di-tuning* dengan benar menjadi sorotan utama dalam perkembangannya karena performansi PID sangat bergantung pada proses *tuning* yang dilakukan. Terdapat dua metode *tuning* yaitu *offline tuning* dan *online tuning*. *Offline tuning* dilakukan secara manual oleh campur tangan manusia, sedangkan *tuning online* dilakukan secara otomatis sesuai dengan metode yang digunakan. Pada penelitian ini diterapkan metode *online tuning* pada kontroler PID. Menurut Feng Lin dan Robert D. Brandt [6], terdapat dua alasan mengapa menggunakan *online tuning*. Pertama, tujuan dan kebutuhan dari kontroler PID sering berubah selama penerapan dalam kondisi yang berbeda. Sering kali performansi sistem yang diinginkan adalah respon yang cepat, tetapi juga dapat mengurangi kesalahan keadaan tunak. Pada kenyataannya respon yang cepat dan mengurangi kesalahan keadaan tunak membutuhkan parameter yang berbeda jika menggunakan kontroler PID. Jika tetap diinginkan respon dengan spesifikasi cepat dan tanpa kesalahan, maka pada kondisi awal disesuaikan untuk mendapatkan respon yang cepat kemudian dilakukan *tuning* ulang untuk mengurangi kesalahan keadaan tunak.

Alasan kedua adalah dikarenakan dalam kenyataannya di dunia industri *plant* yang dikontrol mengalami perubahan parameter pada kondisi-kondisi tertentu. Untuk mengatasi perubahan parameter *plant* diperlukan metode yang dapat beradaptasi langsung dengan perubahan tersebut. Pada penelitian yang telah dilakukan sebelumnya banyak digunakan metode kontrol adaptif dengan berbagai metode untuk mengatasi perubahan *plant* tersebut. Metode yang digunakan baik melalui pendekatan model, identifikasi parameter estimasi, dan pendekatan kecerdasan buatan seperti *fuzzy* dan jaringan saraf tiruan. Dalam penelitian ini ditawarkan metode kontrol adaptif PID dengan pendekatan *Model Reference Adaptive Control* [6]. Keistimewaan metode kontrol adaptif adalah penerapannya yang tidak membutuhkan banyak informasi seperti parameter estimasi dari *plant*.

1.2 Rumusan Masalah

Pada penelitian ini, permasalahan kontrol level proses dengan variasi kondisi beban, dengan menggunakan kontrol level adaptif, yakni PID *Model Reference Adaptive Control* (MRAC).

1.3 Batasan Masalah

Batasan masalah dipergunakan agar penelitian lebih fokus pada topik, maka peneliti membatasi masalah sebagai berikut:

1. Penurunan model matematis dan identifikasi sistem berdasarkan model pada *PCT-100*,
2. *Hardware* yang digunakan pengambilan data dengan *STM32* melalui laptop,
3. Aplikasi yang digunakan untuk mensimulasikan model matematis sistem *PCT-100* dan hasil perancangan dari kontroler *PID Model Reference Adaptive Control (MRAC)* adalah Matlab real-time controller,

1.4 Tujuan

Tujuan dari penelitian ini adalah merancang kontroler *PID Model Reference Adaptive Control (MRAC)* untuk mendapatkan respon waktu yang cepat dan minimum *error* dalam mengontrol level.

1.5 Manfaat

Penelitian ini diharapkan dapat memberikan manfaat yaitu :

1. Menghasilkan sebuah rancangan sistem kontrol yang menggunakan kontroler *PID Model Reference Adaptive Control (MRAC)* untuk mengatur level pada sistem *Process Control Training-100 (PCT-100)*,
2. Dapat dijadikan referensi dalam mengaplikasikan sistem kontrol pada sistem proses industri.
3. Dipublikasikan pada *international journal* atau *international conference*.

BAB 2 TINJAUAN PUSTAKA

2.1 Hasil Penelitian Terdahulu

Penelitian terdahulu digunakan untuk mengetahui metode dan hasil penelitian yang sudah dilakukan. Penelitian terdahulu diperlukan sebagai landasan dan acuan peneliti terhadap perbandingan dan hubungan antar variabel yang diuji.

Penelitian *plant PCT-100* oleh Rachmad Dwi Raharjo (2014) dilakukan dengan metode PID *gain scheduling* terhadap level proses. Identifikasi sistem dilakukan dengan metode *Harriot* dan divalidasi menggunakan *ISE/Integral Square Error*. Hasil yang diperoleh yakni kontroler 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 [7].

Penelitian level proses kemudian disempurnakan oleh penelitian M. Zakki Ghufon (2016) yang menggunakan *self-tuning* PID dengan pendekatan adaptif. Kontroler adaptif dapat memperbaiki kelemahan metode *gain scheduling*, dimana dapat bekerja pada variasi tak terprediksi. Nilai gamma memengaruhi kecepatan adaptasi kontroler *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 [8].

Penelitian *Plant PCT-100* kemudian dikembangkan oleh Khoiruz Fauzi (2022) dilakukan pengaturan *cascade flow* dan *level* menggunakan PID *Particle Swarm Optimization*. Identifikasi sistem dilakukan dengan metode *Harriot* dan divalidasi menggunakan *ISE/Integral Square Error* dan *ITAE/Integral Time Absolute Error*. Hasil yang diperoleh dari kontroler PID PSO semakin banyak jumlah partikel dapat meningkatkan performansi dinamik. Dengan meningkatnya nilai *w*/bobot inersia, dengan PSO dapat menghasilkan minim *rise time*, *settling time*, *overshoot*, dan *steady state error* kecil.[9]

2.2 Dasar Teori

2.2.1 Plant PCT-100

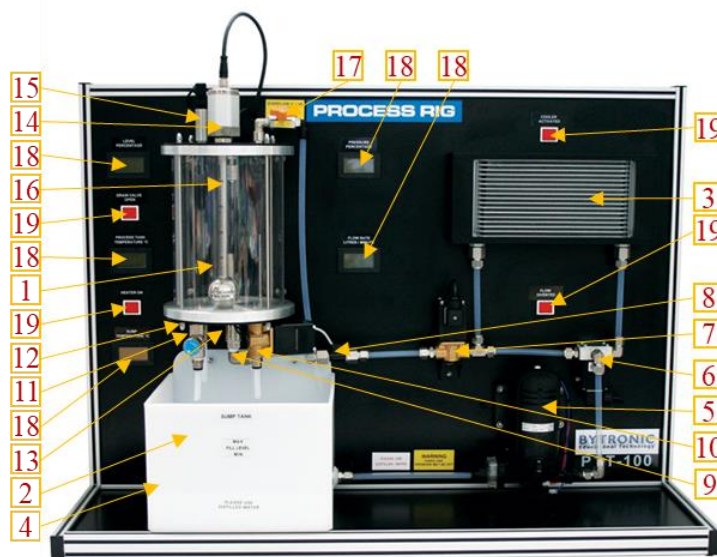
PCT-100 merupakan modul *training process control* yang dilengkapi dengan beberapa sensor dan aktuator untuk aplikasi dari suatu proses. Sensor yang terdapat pada *PCT-100* yaitu sensor *level*, *temperature*, *flow* dan *pressure*, sedangkan aktuator berupa *pump*, *heating*, *valve* dan *cooling*. Semua sensor dan aktuator tersebut saling terintegrasi, tetapi tidak harus diaktifkan keseluruhan, namun dapat diaktifkan sesuai dengan kebutuhan. Sistem ini memiliki sistem yang non linier pada proses level yang mengalami perubahan nilai *set point* secara langsung mengakibatkan hasil keluaran menjadi tidak sesuai dengan yang diinginkan.

PCT-100, juga dikenal sebagai *Process Control Technology-100*, memiliki beberapa tujuan yang ingin dicapai. Pertama, alat ini menghadirkan manfaat besar dalam menerapkan pengaturan berbasis mikroprosesor untuk proses otomatisasi. Kemudian, *PCT-100* menyediakan simulasi skala kecil yang menggambarkan masalah yang sering dihadapi dalam dunia industri. Dengan demikian, pengguna dapat mengaplikasikan berbagai teknik kontrol yang berbeda. Selanjutnya, alat ini menunjukkan keefektifan dan kemudahan penggunaan kontroler yang

populer, seperti PID (*Proportional-Integral-Derivative*). Melalui pengamatan terhadap sinyal digital dan analog, pengguna dapat meningkatkan pemahaman mereka tentang metode yang sedang digunakan. Terakhir, PCT-100 menyediakan fasilitas yang cocok bagi para operator untuk melatih keterampilan dalam mengidentifikasi dan menyelesaikan masalah. PCT-100 terdiri dari dua komponen utama, yaitu *rig* proses dan modul kontrol. Rig proses merepresentasikan sistem fisik yang dikendalikan, sedangkan modul kontrol bertanggung jawab dalam mengatur dan mengendalikan proses tersebut. Kedua komponen ini terhubung melalui *port* serial untuk memfasilitasi pertukaran data yang diperlukan dalam pengendalian. Penjelasan lebih detail mengenai masing-masing komponen disajikan pada subbab selanjutnya.

2.2.1.1 *Process Rig*

Process rig adalah komponen utama dalam *PCT-100* yang merepresentasikan sistem fisik yang ingin dikendalikan seperti pada Gambar 2.1. Rig ini mensimulasikan proses industri dalam skala kecil dan terdiri dari perangkat keras seperti pompa, *valve*, sensor, dan aktuator. Pengguna dapat mengatur, memantau, dan mengendalikan *rig* menggunakan modul kontrol yang terhubung. Dengan *process rig*, pengguna dapat melakukan eksperimen dan mengembangkan pemahaman tentang pengendalian proses serta menerapkan teknik kontrol yang tepat.



Gambar 2. 1 *Process Rig PCT-100*

Pada Gambar 2.1 terdapat penjelasan terhadap angka yang ditunjukkan pada Tabel 2.1.

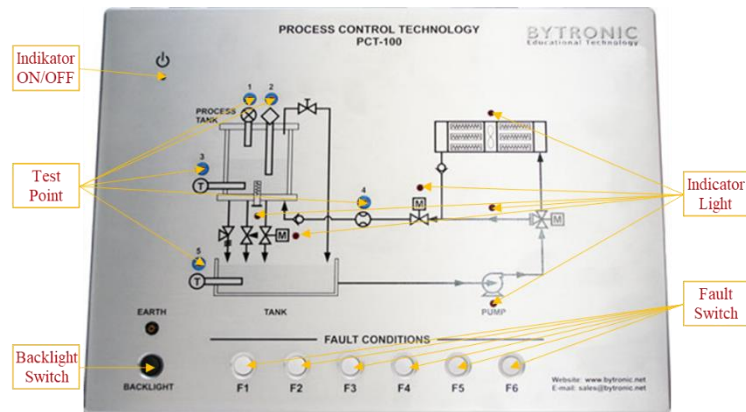
Tabel 2. 1 Komponen Penyusun *Process Rig*

No.	Keterangan
1	Tangki Proses
2	Tangki Penampung Air
3	Pendingin
4	Sensor Suhu Tangki Penampung
5	Pompa
6	<i>3/2 Diverter Valve</i>
7	<i>2/2 Proportional Control Valve</i>
8	Sensor Flow
9	<i>One Way Check Valve</i>
10	<i>2/2 Proportional Gain Valve</i>
11	<i>Needle Valve</i>
12	<i>Pressure Relief Valve</i>
13	Pemanas
14	Sensor Level
15	<i>Pressure Transducer</i>
16	<i>Float Switch</i>
17	<i>Overflow/Vent Valve</i>
18	<i>Digital LCD Display</i>
19	Lampu Indikator

Tangki penampung air digunakan untuk menampung air hasil distilasi sebelum dipompa ke seluruh sistem dengan laju aliran sekitar 3.2 liter/menit. Air dapat langsung mengalir ke *process tank* atau diarahkan ke *cooler*. *Flow* meter digunakan untuk mengukur aliran air yang masuk ke *process tank*, sementara level sensor digunakan untuk mengukur ketinggian permukaan air. Di dalam *process tank*, tekanan dan suhu air dapat diukur. Untuk mengakhiri siklus proses, air dalam tangki ini dapat dikeluarkan melalui manual atau *proportional drain valve*. PCT-100 juga dilengkapi dengan digital *display* yang menampilkan informasi suhu, aliran, tekanan, dan level air saat ini. Selain itu, terdapat indikator *on/off* untuk *cooler*, *heater*, *diverter*, dan *drain valve*. Dengan adanya fitur-fitur ini, pengguna dapat dengan mudah memantau dan mengendalikan suhu, aliran, tekanan, dan level air dalam sistem secara efisien.

2.2.1.2 Control Module

Control module PCT-100 adalah komponen sentral yang mengendalikan proses *rig* seperti pada Gambar 2.2. Dengan menggunakan mikroprosesor dan perangkat lunak, *control module* memungkinkan pengguna untuk mengatur, memonitor, dan mengendalikan parameter kunci seperti suhu, level, aliran, dan tekanan. Antarmuka yang intuitif, seperti layar sentuh atau tombol kontrol, mempermudah pengguna dalam mengoperasikan dan memantau proses *rig* dengan mudah.



Gambar 2. 2 Control Module

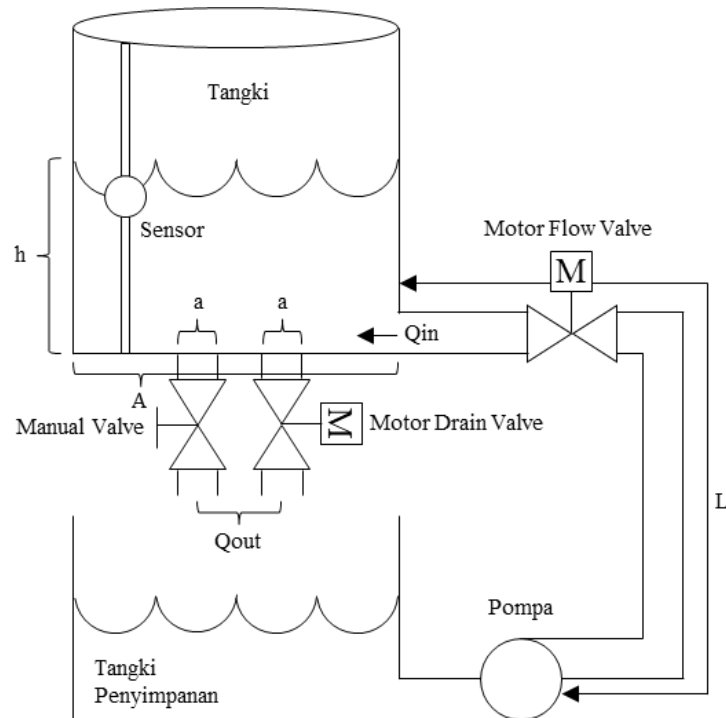
Control module tidak hanya berfungsi sebagai pusat pengendalian, tetapi juga menyediakan kemampuan kontrol otomatis yang efektif, termasuk metode kontrol PID. Dengan kontrol PID, pengguna dapat mengoptimalkan kinerja sistem dengan lebih baik, menghasilkan respons yang cepat dan akurat terhadap perubahan dalam parameter yang diatur.

Data yang dihasilkan oleh sensor-sensor pada *rig* ditampilkan secara *real-time* pada *control module*, memungkinkan pengguna untuk memantau kondisi proses secara langsung. Informasi yang tersedia melalui *control module* membantu pengguna dalam mengambil keputusan yang tepat dan mengatasi masalah yang mungkin muncul selama proses.

Dengan *control module PCT-100*, pengguna dapat mengendalikan dan memantau proses *rig* dengan akurasi dan efisiensi yang tinggi, memastikan sistem berjalan dengan optimal dan sesuai dengan kebutuhan aplikasi yang diinginkan.

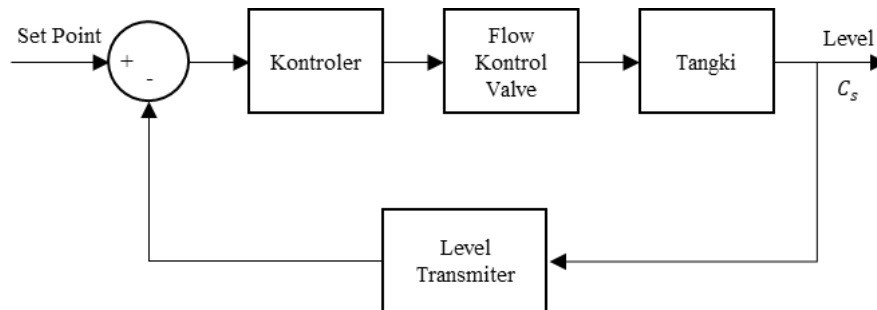
2.2.2 Pemodelan Plant PCT-100

Plant yang digunakan pada Gambar 2.3 merepresentasikan dua proses utama dalam sistem pengaturan level yaitu proses pengisian dan pembuangan air. Proses pembuangan air dalam tangki dilakukan oleh *manual drain valve* dan *motor drain valve* sedangkan proses pengisian air dalam tangki dilakukan oleh pompa dan *motor flow valve*.



Gambar 2. 3 Sistem Level Plant PCT-100

Proses pemodelan *plant* pada sistem pengaturan level digambarkan seperti diagram alir pada Gambar 2.4. Proses pemodelan dimulai dari pemodelan motor *flow valve*, *flow meter*, tangki, *level sensor*, dan *drain valve*.



Gambar 2. 4 Diagram Blok Pemodelan Sistem Pengaturan Level

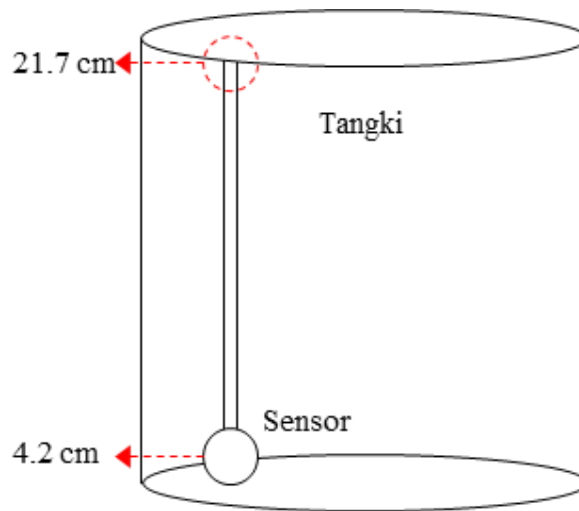
Semua parameter pemodelan *plant* pada Tabel 2.2 beserta keterangan dan satuan yang digunakan nilainya diambil dari *data sheet* dan pengukuran langsung pada *plant PCT-100*.

Tabel 2. 2 Parameter Model *Plant PCT-100*

Simbol	Keterangan	Satuan
Q_{in}	Debit air yang masuk ke tangki	cm^3/s
Q_{out}	Debit air yang keluar dari tangki	cm^3/s
Q	Debit air dalam tangki	cm^3/s
H	Ketinggian air dalam tangki	cm
A	Luas alas tangki	cm^2
a	Diameter Pipa	cm
h_0	Ketinggian air saat kondisi di titik operasi	cm
Q_0	Laju aliran keluar saat kondisi di titik operasi	cm^3/s
R	Resistansi <i>control valve</i>	s/cm^2
C	Kapasitansi tangki	cm^2
L	Panjang Pipa	cm
θ	Waktu tunda	detik

2.2.2.1 Pemodelan Level Transmitter

Level transmitter merupakan *instrument* pengukur ketinggian permukaan air. Terdapat *floating ball* yang akan bergerak naik seiring kenaikan air dalam tangki. Hasil kenaikan *floating ball* menghasilkan output tegangan sensor. Level sensor memiliki rentang kerja dengan batas awal bergerak 4.2cm (*floating ball* bergerak) dan batas akhir bergerak 21.7cm (*floating ball* berhenti bergerak) seperti pada Gambar 2.6



Gambar 2. 5 Posisi *Floating Ball* Pada Proses Tangki

Output berupa tegangan level sensor (V_{LT}) terhadap ketinggian air (h), kemudian dilakukan regresi linier untuk mendapatkan persamaan garis lurus.

$$V_{LT}(h) = m h + c \quad (2.1)$$

$$m = \frac{V_{LT\ 4} - V_{LT\ 1}}{h_4 - h_1} \quad (2.2)$$

$$c = V_{LT\ 4} - m h_4 \quad (2.3)$$

2.2.2.2 Pemodelan Tangki [10]

Sistem pengaturan level tangki reaksi umumnya terdiri dari pengumpanan dan pembuangan air. Kapasitansi tangki adalah besarnya perubahan volume cairan yang diperlukan untuk menghasilkan perubahan ketinggian satu satuan. Jika luas penampang tangki konstan, kapasitansi juga konstan. Kapasitansi tangki mempengaruhi respon sistem terhadap perubahan pengumpanan dan pembuangan air. Semakin besar kapasitansi tangki, semakin besar perubahan volume air yang dapat ditampung oleh tangki untuk setiap perubahan ketinggian yang diberikan.

$$C = \frac{\text{perubahan volume cairan}(cm^3)}{\text{perubahan ketinggian cairan}(cm)} \quad (2.4)$$

$$C = \frac{\Delta V_{tangki}}{\Delta h_{tangki}} = A_{tangki} \quad (2.5)$$

Pemodelan tangki dimulai dari hukum kesetimbangan massa pada Persamaan (2.6) dimana debit air didalam merupakan selisih antara debit air yang masuk ke dalam tangki dan debit air yang keluar dari tangki.

$$Q_{in} - Q_{out} = Q \quad (2.6)$$

$$Q = Av = A \frac{dh}{dt} \quad (2.7)$$

Substitusikan Persamaan (2.5) dan (2.7) pada Persamaan (2.6) sehingga hubungan antara debit air yang masuk dan yang keluar dituliskan pada Persamaan (2.8).

$$Q_{in} - Q_{out} = C \frac{dh}{dt} \quad (2.8)$$

Control valve diasumsikan seperti resistansi atau hambatan pada saat proses pembuangan air. Hal tersebut dikarenakan debit air yang keluar bergantung pada bukaan dari control valve, sehingga dapat dirumuskan pada Persamaan (2.9).

$$R = \frac{\text{perubahan ketinggian air}(cm)}{\text{perubahan debit air keluar}(cm^3/s)} = \frac{h}{Q_{out}} \quad (2.9)$$

$$Q_{out} = \frac{h}{R} \quad (2.10)$$

Substitusi hubungan antara debit air yang keluar dengan bukaan *valve* pada Persamaan (2.10) ke Persamaan (2.8), sehingga rumus kesetimbangan massa menjadi Persamaan (2.11).

$$Q_{in} - \frac{h}{R} = C \frac{dh}{dt} \quad (2.11)$$

$$RQ_{in} - h = RC \frac{dh}{dt} \quad (2.12)$$

Dengan menggunakan *transformasi Laplace* pada Persamaan (2.12) maka diperoleh fungsi alih sistem pada Persamaan (2.15).

$$\frac{H(s)}{Q_{in}(s)} = \frac{R}{RCs + 1} \quad (2.13)$$

$$\frac{Q_{out}(s)}{Q_{in}(s)} = \frac{1}{RCs + 1} \quad (2.14)$$

$$Q_{out}(s) = \frac{1}{R} H(s) \quad (2.15)$$

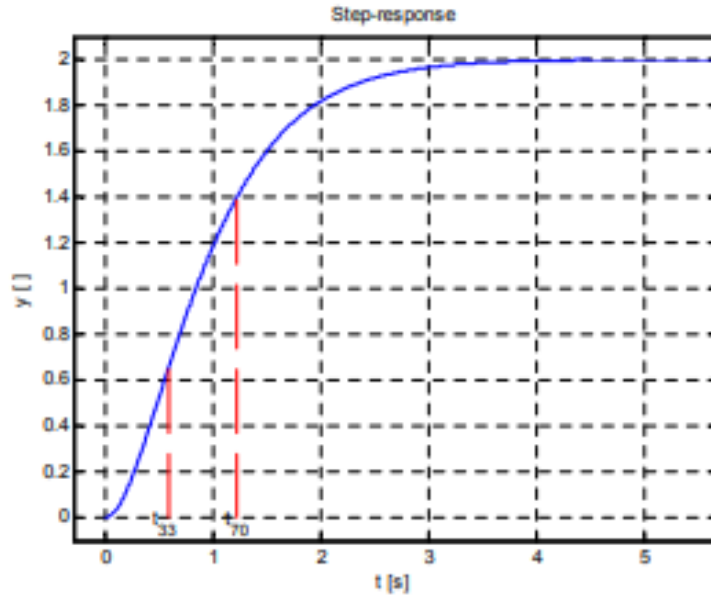
2.2.3 Identifikasi Sistem [11]

Identifikasi sistem merupakan proses pengumpulan informasi dan analisis untuk memahami bagaimana suatu sistem bekerja, termasuk komponen-komponennya, interaksi antar komponen, dan fungsi-fungsi yang dilakukan.

Pada penelitian ini digunakan identifikasi dengan metode *vítečková* untuk menghasilkan model *flow*. *Vítečková* merupakan metode yang digunakan untuk melakukan aproksimasi sistem. Dalam metode ini, sistem dapat diaproksimasi dengan model orde pertama dengan kemungkinan inklusi tundaan waktu. Fungsi transfer model yang dihasilkan dapat dituliskan pada persamaan (2.16).

$$G_{V1}(s) = \frac{K}{(\tau_{V1}s + 1)} e^{-T_{av1}s} \quad (2.16)$$

$$K = \frac{Y_{ss}}{U} \quad (2.17)$$



Gambar 2. 6 Vítěčková's Curve

Dimana T_{dV1} merupakan time delay dan τ_{V1} merupakan konstanta waktu, dapat dituliskan pada persamaan (2.18) dan (2.19).

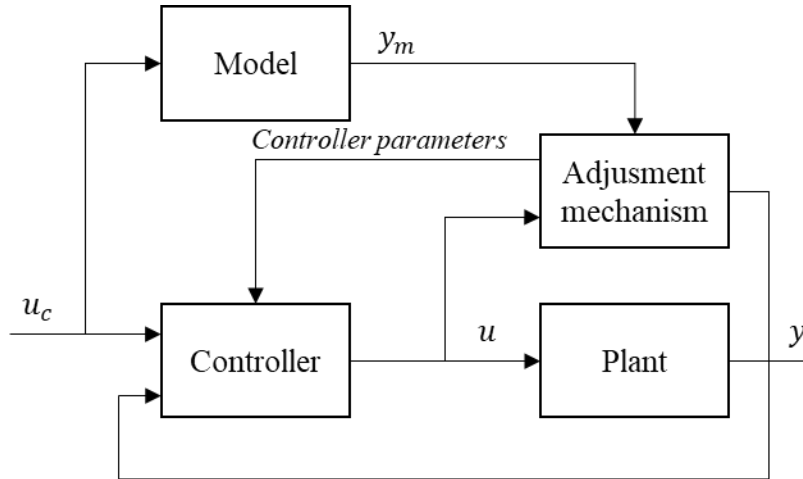
$$T_{dV1} = 1.498t_{33} - 0.498t_{70} \quad (2.18)$$

$$\tau_{V1} = 1.245(t_{70} - t_{33}) \quad (2.19)$$

Dimana t_{33} dan t_{70} merupakan waktu yang didapat dari step respon mencapai 33% dan 70% dari nilai stady state.

2.2.4 Kontroler PID MRAC

Sistem kontrol adaptif memiliki kontroler dimana parameter-parameternya dapat diatur (*adjustable*) dan memiliki mekanisme pengaturan terhadap parameter-parameternya dengan susunan blok diagram pada Gambar 2.8. Skema *Model Reference Adaptive Controller (MRAC)* merupakan salah satu pengaturan adaptif dimana performansi keluaran sistem (*plant*) mengikuti performansi keluaran model referensinya. Kontroler MRAC bisa ditingkatkan lagi akurasi dan juga robustness nya dengan cara menggabungkan MRAC dengan kontroler *Proportional-Integral-Derivative* (PID). Jika dibandingkan dengan kontroler PID konvensional, kontroler PID MRAC dapat melakukan penyetelan parameter PID secara otomatis sehingga sistem akan stabil dalam hampir segala kondisi [13].



Gambar 2. 7 Blok Diagram PID MRAC Konvensional [13]

Pada perancangan kontroler PID-MRAC, parameter kontroler pada PID disetel secara otomatis oleh kontroler MRAC [12].

2.2.4.1 MIT Rule

MIT Rule merupakan salah satu pendekatan yang dapat digunakan dalam perancangan MRAC. Hal utama dari perancangan *MRAC* terletak pada model referensi, model sistem, dan juga kontroler. Keluaran dari model referensi dinotasikan dengan (y_m) sementara untuk keluaran model *plant* dinotasikan dengan (y). Selisih dari kedua nilai tersebut akan menghasilkan sebuah tracking error (e).

$$e = y - y_m \quad (2.20)$$

$$J(\theta) = \frac{1}{2} e^2 \quad (2.21)$$

J merupakan *cost function* dari kontroler dan θ merupakan vektor dari parameter kontroler. Nilai dari θ dapat berubah terhadap waktu, dengan arah negatif gradien dari e^2 [13]. Hal ini dikarenakan pada MRAC, kontroler dirancang agar keluaran dari model sistem memiliki nilai yang mendekati keluaran dari model referensi, sehingga nilai dari e harus dibuat sekecil mungkin [13].

$$\frac{d\theta}{dt} = -\gamma \frac{dJ}{d\theta} = -\gamma \frac{dJ}{de} \cdot \frac{de}{d\theta} = -\gamma e \cdot \frac{de}{d\theta} \quad (2.22)$$

Dimana:

γ : *adaptive rate*

$\frac{de}{d\theta}$: *sensitivity derivative*

2.2.4.2 Kontroler PID

Kontroler *Proportional – Integral – Derivative* atau PID merupakan pendekatan yang paling sering digunakan pada proses industri. Hal ini dikarenakan struktur yang sederhana

sehingga mudah diaplikasikan dan juga dapat memberikan hasil yang memuaskan. Kontroler PID dapat ditulis dengan persamaan [12].

$$G_{PID}(s) = K_p + \frac{K_i}{s} + K_d \cdot s = K_p \left(1 + \frac{1}{T_i \cdot s} + T_d \cdot s \right) \quad (2.23)$$

di mana K_p adalah *gain proportional*, K_i adalah *gain integral*, K_d adalah *gain derivative*, T_i adalah konstanta *integral time*, dan T_d adalah konstanta *derivative time*. Persamaan (2.24) dapat ditulis dalam domain waktu sebagai [12].

$$u(t) = K_p e(t) + K_i \int e(t) d\tau + K_d \frac{de(t)}{dt} \quad (2.24)$$

Dalam aplikasi kontroler secara kaskade, diperlukan penyesuaian di mana pada kontroler *derivative* tidak menggunakan nilai *error* sistem, melainkan menggunakan variabel keluaran sistem $y(t)$ [12].

$$u(t) = K_p e(t) + K_i \int e(\tau) d\tau + K_d \frac{dy(t)}{dt} \quad (2.25)$$

Untuk mendapatkan hasil yang memuaskan, kontroler PID harus disetel terlebih dahulu. Terdapat beragam metode penyetelan PID, seperti *Ziegler–Nichols* atau *Cohen–Coon*. Karakteristik dari masing-masing nilai kontroler dapat dilihat pada Tabel 2.3.

Tabel 2. 3 Karakteristik Kontroler PID

<i>Gain Kontroler</i>	<i>Rise Time</i>	<i>Settling Time</i>	<i>Overshoot</i>	<i>Error Steady State</i>
K_p	Berkurang	Sedikit Perubahan	Berkurang	Berkurang
K_i	Berkurang	Bertambah	Berkurang	Berkurang
K_d	Sedikit Perubahan	Berkurang	Berkurang	Sedikit Perubahan

Nilai karakteristik kontroler PID di atas hanya bisa dapat digunakan sebagai acuan namun belum tentu tepat jika masih terdapat kesalahan dalam tuning.

2.2.5 Root Mean Square Error (RMSE)

Root Mean Square Error (RMSE) merupakan besarnya *error* (galat) antara nilai referensi/model dengan nilai aktual dari sebuah respon. Semakin kecil nilai *RMSE* maka semakin baik respon tersebut terhadap nilai referensinya.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n e^2} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y - y_m)^2} \quad (2.26)$$

Dimana:

Y_m : Nilai referensi

Y : Nilai aktual

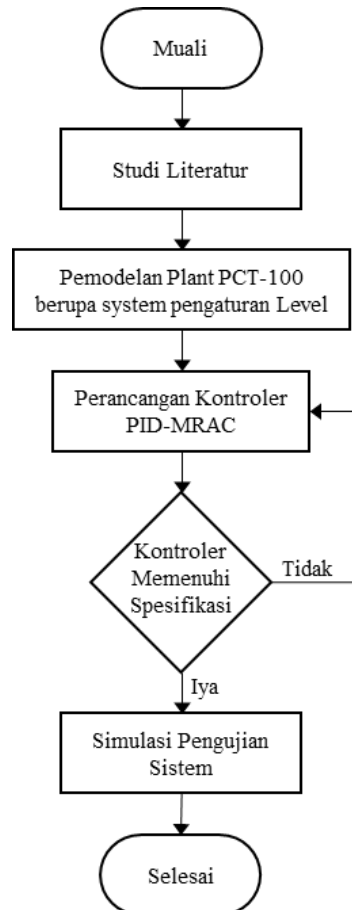
n : Jumlah sampel data

Penggunaan RMSE dinilai memiliki sensitivitas yang lebih baik terhadap keberadaan data *outlier* dan kemungkinan kemunculannya dibandingkan dengan *Mean Absolute Error* (MAE) [14].

BAB 3 METODOLOGI

3.1 Metode Penelitian

Bab ini akan menjelaskan mengenai pemodelan sistem dan perancangan kontroler pada *PCT-100*, mulai dari *Indetifikasi system* untuk *flow*, *modelling* dengan matematis untuk pengaturan level, hingga perancangan kontroler PID-MRAC yang digunakan dalam Tugas Akhir ini. Pada Gambar 3.1 merupakan diagram alur kerja yang dilakukan dalam perancangan kontroler PID-MRAC pada *PCT-100*.



Gambar 3. 1 Flowchart Diagram Alur Penelitian

Berdasarkan Gambar 3.1 terdapat beberapa tahap yang perlu dikerjakan sesuai diagram alur. Tahap pertama adalah studi literatur, pada tahap ini dicari beberapa referensi untuk mendapatkan pemahaman terkait dengan model *flow*, model *level*, dan perancangan kontroler PID-MRAC. Tahap kedua pemodelan *flow*, dilakukan metode identifikasi sistem dalam simulasi dan dilakukan dengan memberikan sinyal uji pada flow plant *PCT 100* sehingga diketahui karakteristik dari dari flow plant tersebut. Tahap ketiga pemodelan level, dilakukan *modelling* dengan matematis berdasarkan parameter level tangki pada *PCT 100*. Tahap keempat adalah perancangan kontroler PID-MRAC untuk nantinya akan digunakan pada sistem. Tahap terakhir adalah melakukan pengujian kontroler PID-MRAC pada model sistem.

3.2 Bahan dan peralatan yang digunakan

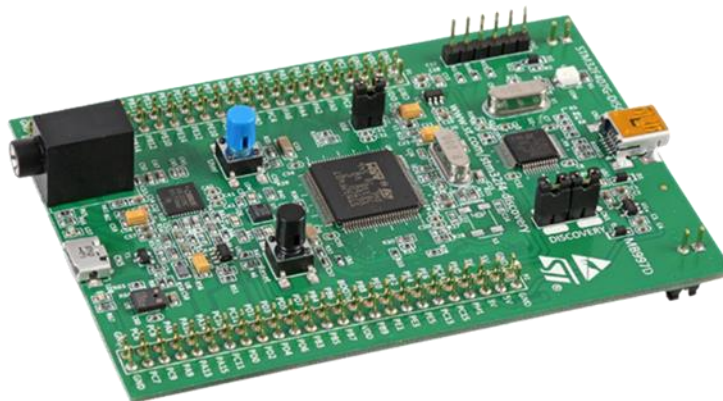
3.2.1 *Process Control Technology 100*



Gambar 3. 2 *Process Kontrol Technology 100*

PCT-100 merupakan modul *training process control* yang dilengkapi dengan beberapa sensor dan aktuator untuk aplikasi dari suatu proses. Sensor yang terdapat pada PCT-100 yaitu sensor *level*, *temperature*, *flow* dan *pressure*, sedangkan aktuator berupa *pump*, *heating*, *valve* dan *cooling*. Semua sensor dan aktuator tersebut saling terintegrasi, tetapi tidak harus diaktifkan keseluruhan, namun dapat diaktifkan sesuai dengan kebutuhan.

3.2.2 *Data Acquisition Menggunakan STM32*



Gambar 3. 3 Mikrokontroler STM32F407VG

Data acquisition merupakan proses pengumpulan dan perekaman data dari berbagai sumber atau sensor fisik dalam suatu sistem. Ini melibatkan penggunaan perangkat keras dan konversi sinyal analog menjadi digital. Data yang diakuisisi dapat dianalisis dan digunakan untuk pengendalian sistem. Pada penelitian ini digunakan *STM32F407VG* sebagai mikrokontroler seperti pada Gambar 3.3 untuk pengambilan data *real time system plant* dan implementasi.

3.2.3 ADAM 5000 L TCP/IP



Gambar 3. 4 ADAM 5000L/TCP

ADAM 5000L/TCP merupakan sistem I/O berbasis *Ethernet*. Sistem ini dirancang untuk menyediakan konfigurasi jarak jauh dan pengelolaan yang efisien dalam arsitektur *eAutomation*. Sistem ini mendukung komunikasi *Ethernet* dan dapat mencakup jarak komunikasi hingga 100 m tanpa memerlukan *repeater*. Hal ini memungkinkan penempatan modul I/O yang fleksibel dalam infrastruktur jaringan.

Pada Penelitian ini ADAM 5000L digunakan sebagai *Input* modul dengan konfigurasi dengan STM32F407VG . Fungsi ADAM untuk memberi sinyal *Input* berupa tegangan pada pompa PCT 100. Untuk aktuator diberi *input* tegangan dari STM32F407VG. Spesifikasi dari ADAM 5000L dapat dilihat pada Tabel 3.1.

Tabel 3. 1 Spesifikasi ADAM 5000L/TCP

Konsumsi Daya	4W, 24 VDC (tidak termasuk modul I/O)
<i>Power Input</i>	10-30 VDC
Konektor	1 x DB9-M/DB9-F/ <i>screw</i> terminal untuk RS-485 (komunikasi), 1 x DB9-F untuk RS-232 (penggunaan internal), 1 x <i>Screw</i> terminal untuk <i>power input</i> , 1 x RJ-45 untuk LAN

3.2.4 Aplikasi Matlab

Matlab merupakan aplikasi bahasa pemrograman dan lingkungan komputasi numerikal yang digunakan untuk mengelola data, menghitung, dan mengembangkan algoritma. *Matlab* dikembangkan oleh *MathWorks* dan menyediakan fungsi-fungsi matematis, visualisasi data, dan pemodelan integratif yang kuat untuk menganalisis dan memvisualisasikan data serta membangun aplikasi yang didasarkan pada hasil analisis tersebut. *Matlab* juga dapat digunakan untuk membangun sistem kontrol, memproses sinyal, dan menganalisis data secara *real-time*. *Simulink* adalah perangkat lunak untuk pemodelan sistem dinamis dalam bentuk diagram blok. *Simulink* dapat dilakukan pemilihan dan menghubungkan blok-blok yang mewakili fungsi-fungsi sistem untuk membangun model. *Simulink* juga mendukung simulasi, analisis, dan integrasi dengan kode implementasi.

3.2.5 Komputer atau Laptop

Pada Penelitian ini digunakan Laptop pribadi dengan spesifikasi terdapat pada Tabel 3.2.

Tabel 3. 2 Spesifikasi Laptop yang Digunakan

<i>Processor</i>	Intel(R) Core(TM) i7-3632QM CPU @ 2.20GHz 2.20 GHz
<i>RAM</i>	8 GB
<i>Storage</i>	500 GB

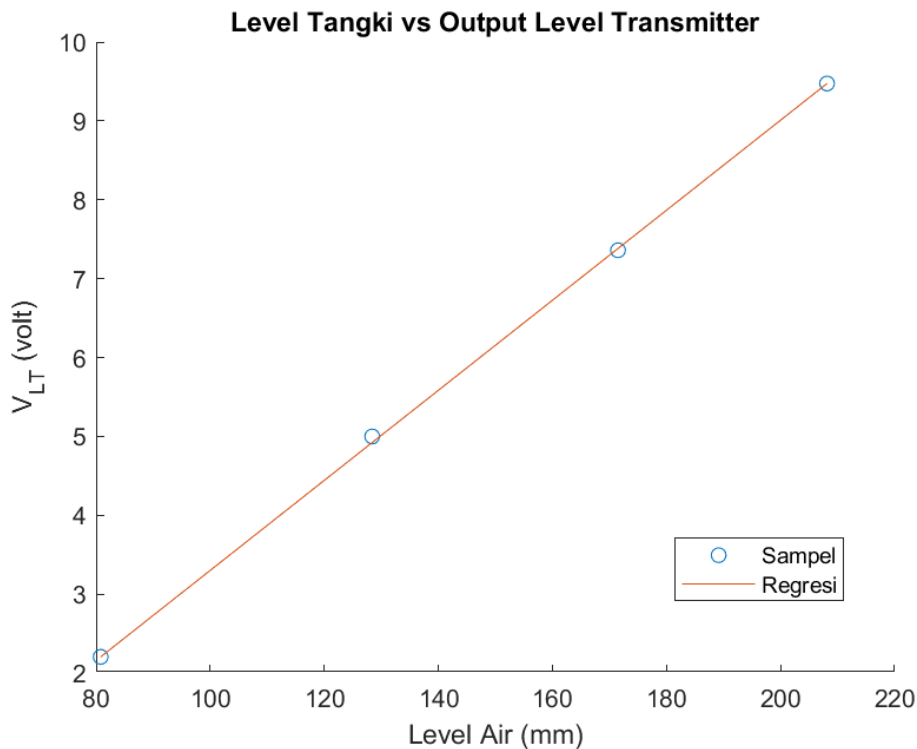
3.3 Urutan pelaksanaan penelitian

3.3.1 Pemodelan Sistem Pengaturan Level

Prinsip kerja dari pengaturan level air pada suatu sistem adalah untuk menjaga tingkat air dalam tangki atau wadah pada nilai yang diinginkan. Pada dasarnya, pengaturan level air melibatkan penggunaan sensor level yang mengukur tingkat air aktual dalam tangki. Sensor ini akan mengirimkan informasi mengenai tingkat air tersebut kepada sistem pengendali. Sistem pengendali akan membandingkan tingkat air aktual dengan nilai yang diinginkan atau *set point*. Jika ada perbedaan antara keduanya, sistem pengendali akan menghasilkan sinyal pengendali yang sesuai.

3.3.1.1 Pemodelan Level Transmitter

Pemodelan *level transmitter* dilakukan pengambilan 4 sampel ketinggian air dan menghasilkan tegangan dari level transmitter, kemudian dilakukan regresi linier untuk mendapatkan persamaan garis lurus seperti pada Gambar 3.6 .



Gambar 3. 5 Respon Tegangan Terhadap Ketinggian Air

Perhitungan Gradien garis lurus digunakan sampel awal/terendah dan akhir/tertinggi yaitu $h_1 = 8.08 \text{ cm}$ menghasilkan tegangan $V_{LT1} = 2.2046$ dan $h_4 = 20.815$ menghasilkan tegangan $V_{LT4} = 9.4739$. Kemudian digunakan persamaan (2.2), (2.3).

$$m = \frac{9.4739 - 2.2046}{20.815 - 8.08} = 0.5708 \text{ Volt/cm} \quad (3.1)$$

$$c = 9.4739 - (0.5708 \times 20.815) = -2.4073 \text{ Volt} \quad (3.2)$$

Kemudian hasil dari persamaan (3.1) dan (3.2) di substitusi kedalam persamaan (2.1).

$$V_{LT}(h) = 0.5708 h - 2.4073 \quad (3.3)$$

Fungsi level sensor untuk jangkauan dari dasar hingga dinding atas tangki dirumuskan oleh persamaan

$$V_{LT}(h) = \begin{cases} 0 & ; h < 4.2 \text{ cm} \\ 0.5708 h - 2.4073 & ; 4.2 \text{ cm} \leq h \leq 21.7 \text{ cm} \\ 10 & ; h \geq 21.7 \text{ cm} \end{cases} \quad (3.4)$$

$$h = \frac{V_{LT}(h) + 2.4073}{0.5708} \quad (3.5)$$

3.3.1.2 Pemodelan Tangki

Pada Pemdolan tangki terdapat 3 kondisi yaitu pada saat *manual drain valve full open*, *motor drain valve full open*, dan kedua *drain valve full open*. Data didapatkan melalui pengamatan secara fisis. Untuk memodelkan tangki dilakukan dengan pemodelan matematis.

Gambar 3. 6 Parameter Fisis Tiap Kondisi

Parameter	Nilai		
	<i>Manual Drain Valve</i>	<i>Motor Drain Valve</i>	<i>Kedua Drain Valve</i>
R_{tangki}	9cm	9cm	9cm
h	8.425cm	8.66cm	7.675cm
Q_{out}	15cm ³ /s	20cm ³ /s	33.33cm ³ /s

Setelah menemukan nilai parameter digunakan persamaan (2.5) dan (2.9) untuk memperoleh model fungsi alih dari sistem pengaturan level.

$$C = A_{tangki} = \pi R_{tangki}^2 = 3.14 (9)^2 = 254.47 \text{ cm}^2 \quad (3.6)$$

Manual Drain Valve
(Kondisi Normal)

$$R = \frac{8.425}{15} = 0.5616 \quad (3.7)$$

$$C = A_{tangki} = \pi R_{tangki}^2 = 3.14 (9)^2 = 254.47 \text{ cm}^2 \quad (3.8)$$

Motor Drain Valve

$$R = \frac{8.66}{20} = 0.433 \quad (3.9)$$

(3.10)

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

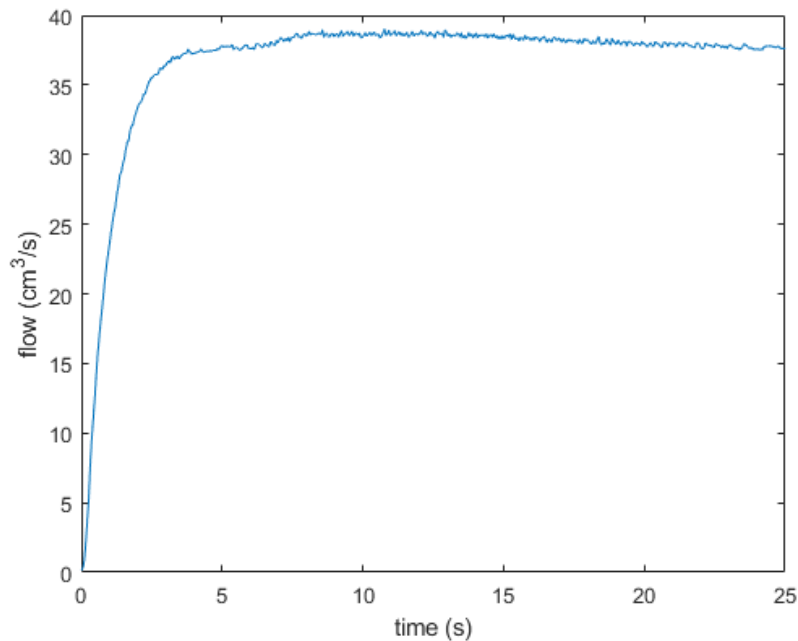
Kedua Drain Valve

(3.11)

$$R = \frac{7.675}{33.33} = 0.2302$$

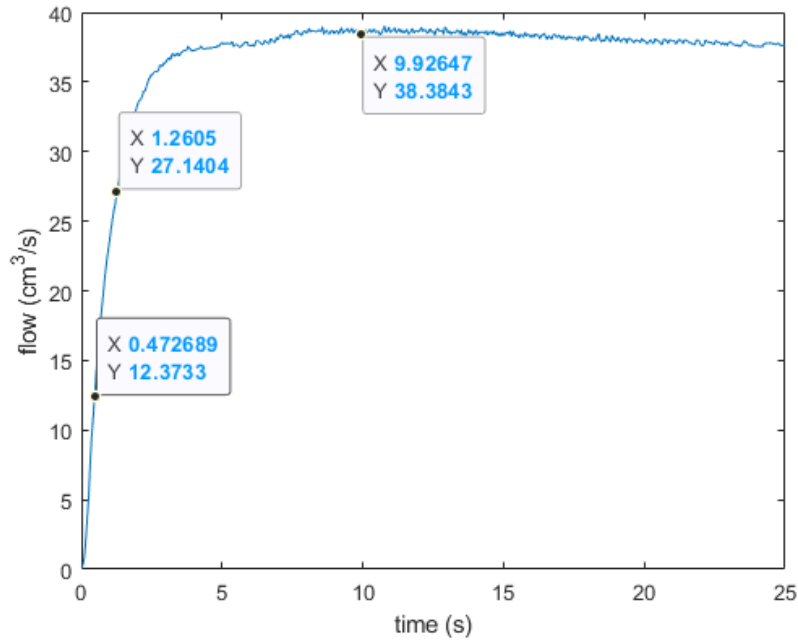
3.3.2 Pemodelan Sistem Pengaturan Flow

Pemodelan *flow* dilakukan identifikasi sistem dengan pengambilan data menggunakan STM32F407VG sebagai *data acquisition*. Data diambil dengan diberi *input* tegangan pompa 10V dan *control valve* 15V. *Control valve* memiliki rentang kerja 12-22V, kemudian dihasilkan grafik respon *flow* pada Gambar 3.7.



Gambar 3. 7 Kurva Respon *Flow*

Dari Gambar 3.7 digunakan metode *vítečková* untuk melakukan aproksimasi *system* dengan pendekatan orde pertama. Kurva respon *flow* kemudian di cari nilai t_{33} dan t_{70} dimana waktu yang didapat dari step respon mencapai 33% dan 70% dari nilai stady state.



Gambar 3.8 Nilai t_{33} dan t_{70} Menggunakan *vítečková* Terhadap Kurva Respon Flow

Dari Gambar 3.8 diperoleh nilai t_{33} dan t_{70} , kemudian digunakan persamaan (2.17), (2.18), dan (2.19) untuk mendapatkan nilai K , T_{dv1} , dan τ_{v1} .

$$K = \frac{38.38}{15} = 2.56 \quad (3.12)$$

$$T_{dv1} = 1.498(0.48) - 0.498(1.27) = 0.0865 \quad (3.13)$$

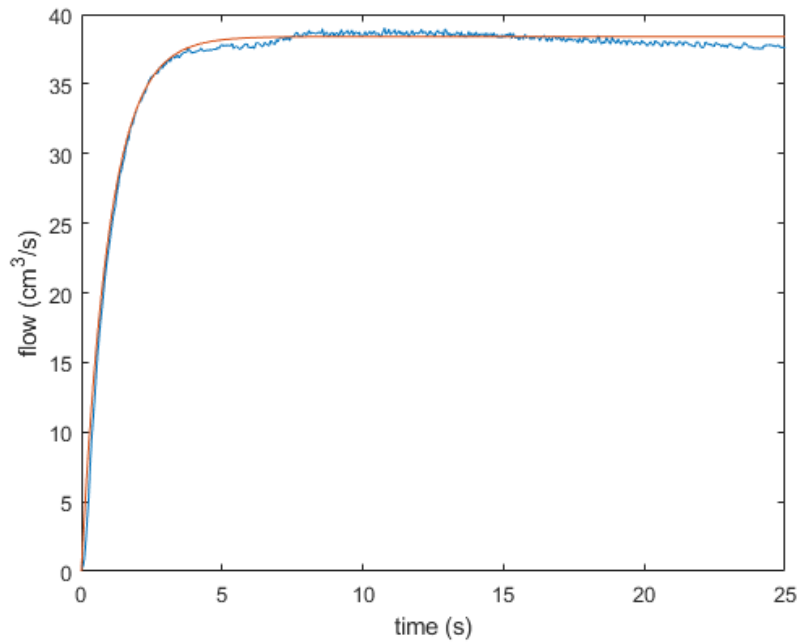
$$\tau_{v1} = 1.245(1.27 - 0.48) = 0.98 \quad (3.14)$$

Kemudian hasil dari persamaan (3.12), (3.13), dan (3.14) di substitusi kedalam persamaan (2.13) dan dihasilkan *transfer fuction flow* pada persamaan (3.15) dan dilakukan perbandingan kurva respon *flow* dengan kurva model seperti pada Gambar 3.9.

$$G_{v1}(s) = \frac{2.56}{0.98s + 1} e^{-0.0865s} \quad (3.15)$$

Karena nilai *time delay* sangat kecil, maka nilai *time delay* bisa diabaikan. Sehingga persamaan (3.18) menjadi.

$$G_{v1}(s) = \frac{2.56}{0.98s + 1} \quad (3.16)$$

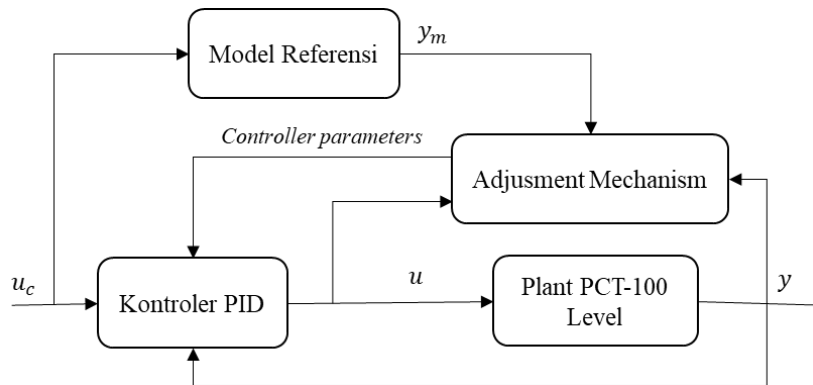


Gambar 3. 9 Kurva Respon Flow Dibanding Kurva Model

Dari Gambar 3.9 dilakukan validasi *error* dengan *root mean square error*, kemudian dihasilkan sebesar $rmse = 2.18$.

3.3.3 Perancangan Kontroler PID MRAC

Mekanisme pengaturan *level* pada *plant PCT-100* dengan kontroler PID MRAC secara garis besar dapat dilihat pada Gambar 3.9.



Gambar 3. 10 Mekanisme Pegaturan Level Pada Plant PCT-100 Dengan PID MRAC

Kontroler PID-MRAC bertugas untuk mengatur agar performansi keluaran sistem mengikuti keluaran model referensinya. Terdapat beberapa sub sistem dari perancangan PID MRAC ini yaitu model sistem, model referensi, mekanisme adaptasi, dan kontroler PID.

3.3.3.1 Penentuan Model Referensi

Dengan mengasumsikan model *plant* berupa orde pertama

$$G(p) = \frac{b}{p + a} \quad (3.17)$$

Dimana nilai a diasumsikan positif dan b adalah nonnegative. Dalam domain waktu dengan menggunakan persamaan operator diferensial $p = \frac{d}{dt}$, persamaan (2.25) dapat dituliskan.

$$u(t) = K_p(u_c(t) - y(t)) + \frac{K_i}{p}(u_c(t) - y(t)) - K_d p y(t) \quad (3.18)$$

Maka didapatkan hubungan antara sinyal masukan dengan model *plant* seperti pada persamaan (3.18).

$$y(t) = G(p)u(t) \quad (3.19)$$

$$y(t) = \frac{\frac{bK_p}{(1+bK_d)} + \frac{bK_i}{(1+bK_d)}}{p^2 + \frac{(a+bK_p)}{(1+bK_d)}p + \frac{bK_i}{(1+bK_d)}} u_c(t) \quad (3.20)$$

Dari persamaan (3.19) dapat dipilih bentuk transfer *function* untuk model referensi kontroler PID *MRAC* dalam bentuk sistem orde dua seperti persamaan [12].

$$G_m(s) = \frac{Y_m(s)}{U_m(s)} = \frac{\alpha s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (3.21)$$

Dimana:

α : gain model referensi

ω_n : frekuensi natural

ζ : rasio redaman

3.3.3.2 Perancangan Mekanisme Adaptasi *MIT Rule*

Mekanisme adaptasi *MIT rule* bekerja pada sistem *plant* yang *closed loop* dan akan ada kontroler yang memiliki parameter θ berubah-ubah dimana nilai $\theta = [K_p \ K_i \ K_d]$ berdasarkan mekanisme *MIT rule* pada persamaan (2.22) didapatkan.

$$\theta = -\frac{\gamma}{p} \cdot e \cdot \frac{de}{d\theta} \quad (3.22)$$

Dari persamaan (3.22) untuk masing-masing nilai parameter K_p , K_i , dan K_d dapat rumuskan [12].

$$K_p = \frac{\left(-\frac{\gamma_p'}{p}\right) \cdot \frac{b}{1+bK_d} \cdot e \cdot p \cdot (u_c - y)}{p^2 + \frac{(a+bK_p)}{(1+bK_d)}p + \frac{bK_i}{(1+bK_d)}} \quad (3.23)$$

$$K_i = \frac{\left(-\frac{\gamma_i'}{p}\right) \cdot \frac{b}{1+bK_d} \cdot e \cdot (u_c - y)}{p^2 + \frac{(a+bK_p)}{(1+bK_d)}p + \frac{bK_i}{(1+bK_d)}} \quad (3.24)$$

$$K_p = \frac{\left(-\frac{\gamma_d'}{p}\right) \cdot \frac{b}{1+bK_d} \cdot e \cdot p^2 y}{p^2 + \frac{(a+bK_p)}{(1+bK_d)}p + \frac{bK_i}{(1+bK_d)}} \quad (3.25)$$

Jika diamati dari persamaan (3.23), (3.24), dan (3.25) dan dibandingkan dengan model referensi pada persamaan (3.21) didapatkan pendekatan berupa.

$$\begin{cases} 2\zeta\omega \approx \frac{(a+bK_p)}{1+bK_d} \\ \omega_n^2 \approx \frac{bK_i}{1+bK_d} \end{cases} \quad (3.26)$$

dengan menyubstitusi persamaan (3.26) ke persamaan (3.23), (3.24), dan (3.25) serta menyubstitusikan $\gamma'' = \frac{b\gamma'}{1+bK_d}$ didapatkan.

$$K_p = \frac{\left(-\frac{\gamma_p''}{p}\right) \cdot e \cdot p \cdot (u_c - y)}{p^2 + 2\zeta\omega_n p + \omega_n^2} \quad (3.27)$$

$$K_i = \frac{\left(-\frac{\gamma_i''}{p}\right) \cdot e \cdot (u_c - y)}{p^2 + 2\zeta\omega_n p + \omega_n^2} \quad (3.28)$$

$$K_p = \frac{\left(-\frac{\gamma_d''}{p}\right) \cdot e \cdot p^2 y}{p^2 + 2\zeta\omega_n p + \omega_n^2} \quad (3.29)$$

3.3.3.3 Normalisasi Parameter

Untuk menyesuaikan parameter kontroler θ dengan desain referensi yang diinginkan, parameter θ harus dinormalisasi untuk memenuhi spesifikasi *bandwidth* sistem. Asumsikan sistem *closed-loop* didefinisikan sebagai.

$$G_{mb}(s) = \frac{\left(\alpha \left(\frac{s}{\omega_b}\right) + \omega_n^2\right)}{\left(\frac{s}{\omega_b}\right)^2 + 2\zeta\omega_n \left(\frac{s}{\omega_b}\right) + \omega_n^2} \quad (3.30)$$

Dimana ω_b merupakan frekuensi *bandwidth* sistem. Jika ω_n sebesar 1 cm³/s, maka persamaan diatas dapat ditulis.

$$G_{mb}(s) = \frac{(\alpha\omega_b + \omega_b^2)}{s^2 + 2\zeta\omega_b s + \omega_b^2} \quad (3.31)$$

Sehingga perhitungan parameter – parameter PID dapat ditulis sebagai.

$$K_p = \frac{\left(-\omega_b^2 \frac{\gamma_p''}{p}\right) \cdot e \cdot p \cdot (u_c - y)}{p^2 + 2\zeta\omega_b p + \omega_b^2} \quad (3.32)$$

$$K_i = \frac{\left(-\omega_b^3 \frac{\gamma_i''}{p}\right) \cdot e \cdot (u_c - y)}{p^2 + 2\zeta\omega_b p + \omega_b^2} \quad (3.33)$$

$$K_p = \frac{\left(-\omega_b \frac{\gamma_d''}{p}\right) \cdot e \cdot p^2 y}{p^2 + 2\zeta\omega_b p + \omega_b^2} \quad (3.34)$$

Dikarenakan ω_b adalah konstanta, maka persamaan (3.32), (3.33), dan (3.34) dapat disederhanakan dan diubah dalam domain frekuensi menjadi.

$$K_p = \left(-\frac{\gamma_p}{s}\right) \cdot e \cdot (u_c - y) \cdot \frac{\omega_b s}{s^2 + 2\zeta\omega_b s + \omega_b^2} \quad (3.35)$$

$$K_i = \left(-\frac{\gamma_i}{s}\right) \cdot e \cdot (u_c - y) \cdot \frac{\omega_b^2}{s^2 + 2\zeta\omega_b s + \omega_b^2} \quad (3.36)$$

$$K_d = \left(-\frac{\gamma_d}{s}\right) \cdot e \cdot y \cdot \frac{s^2}{s^2 + 2\zeta\omega_b s + \omega_b^2} \quad (3.47)$$

Normalisasi parameter PID-MRAC pada persamaan (3.35), (3.36), dan (3.37) dapat mengatur sistem mencapai model referensi yang diinginkan. Normalisasi memastikan ketidakpekaan terhadap variasi kondisi beban dalam sinyal *noise* sehingga membuat *plant* lebih bagus performansinya.

Halaman ini sengaja dikosongkan

BAB 4 Hasil dan Pembahasan

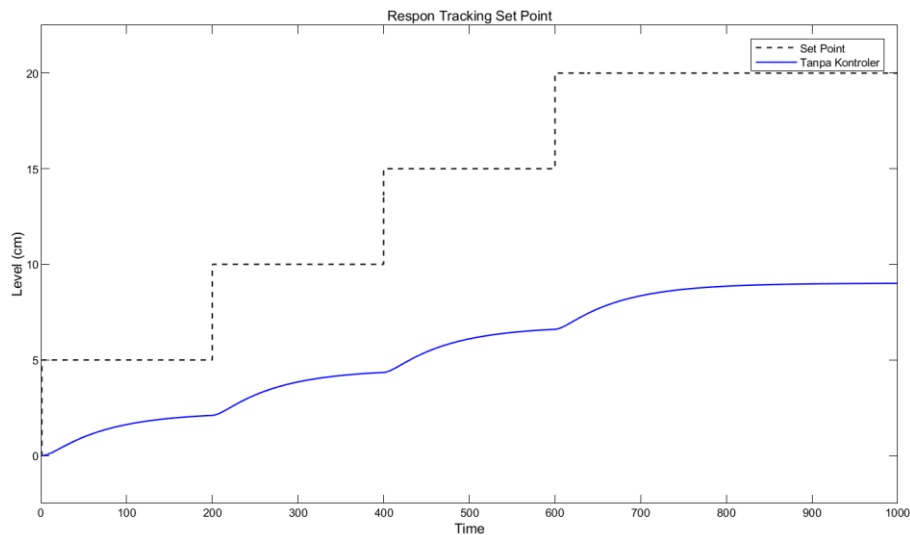
Di dalam bab ini akan menjelaskan mengenai hasil simulasi dan analisis pembahasan data terhadap perancangan kontroler PID-MRAC pengaturan level pada *plant PCT-100* beserta pengujiannya.

4.1 Hasil Penelitian

4.1.1 Pengujian *Tracking Set Point*

4.1.1.1 Tanpa Kontroler

Pada pengujian ini, pengaturan level dirancang tanpa menggunakan kontroler atau hanya dirangkai secara *closed loop* untuk mendapatkan respons sistem. Dengan menggunakan masukan tegangan dan tanpa *noise* didapatkan respons pengaturan level ketinggian air seperti pada Gambar 4.1.

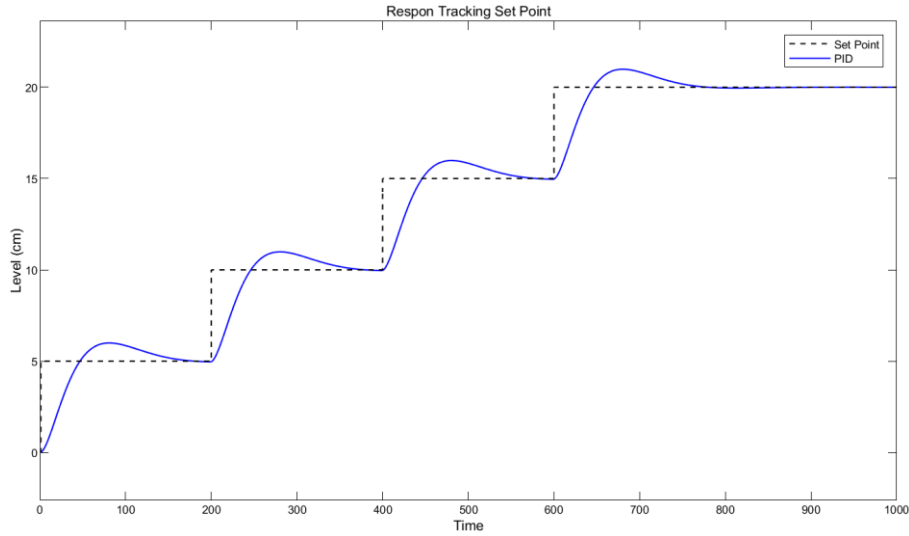


Gambar 4. 1 Respon Perubahan *Set Point Level* Tanpa Kontroler

Berdasarkan respon *close loop*, respon tidak mampu mencapai *set point* masih terdapat error steady state yang besar, yaitu 2.899 cm. Dengan adanya permasalahan tersebut, perlu digunakan kontroler untuk meminimalkan *error steady state* pada sistem.

4.1.1.2 Kontroler *PID*

Pada pengujian ini, digunakan kontroler PID secara *closed loop* untuk mendapatkan respon *system*. Dengan melakukan *tuning* PID ZN kemudian didapatkan nilai $K_p = 5.8$, $K_i = 0.13$, dan $K_d = 7.4$, kemudian diuji pada kondisi 2. Seperti pada Gambar 4.2.



Gambar 4. 2 Respon Perubahan *Set Point Level* Dengan Kontroler PID

Tabel 4. 1 Hasil Pengukuran Uji Perubahan *Set Point* Dengan PID

RiseTime (s)	39.889
Overshoot (%)	0.1988
Error Steady State	0
RMSE	1.5473

4.1.1.3 *PID MRAC*

Pada pengujian ini digunakan kontroler PID-MRAC untuk mengatur level ketinggian air. Model referensi yang digunakan pada pengujian memiliki *settling time* sebesar 3 detik untuk mencegah *overshoot* yang terlalu besar. Untuk mendapatkan kriteria tersebut sesuai dengan persamaan. Persamaan model referensi dalam bentuk *transfer function* sesuai persamaan (3.24) maka ditentukan nilai $\alpha = 1$, $\omega_b = 2$, dan $\zeta = 1$ sehingga didapatkan model referensi.

$$G_m(s) = \frac{\alpha s + \omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{2s + 4}{s^2 + 4s + 4} \quad (3.24)$$

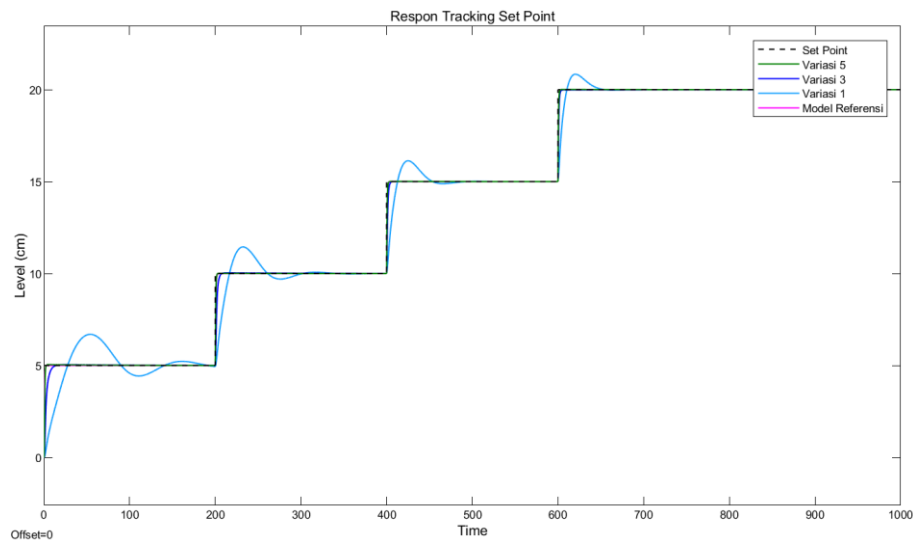
Masukan yang digunakan berupa tegangan telah dibuat sebelumnya pada persmaan (3.3). Pada pengujian menggunakan kontroler PID-MRAC nilai dari inisial $K_p = 1$, $K_d = 0$, dan $K_i = 0$.

Untuk mendapatkan kontroler PID-MRAC yang memiliki performansi yang bagus sesuai dengan *system* pengaturan level maka dibuat beberapa variasi untuk nilai γ_p , γ_i , dan γ_d ditetapkan variasi pada Tabel.

Tabel 4. 2 Variasi Parameter Kontroler PID MRAC

Variasi	Adaptive Rate			Keterangan
	γ_p	γ_i	γ_d	
1	1×10	1×10^{-2}	1×10^2	-
2	1×10	1×10^{-1}	1×10^{-2}	-
3	1×10^2	1×10^{-1}	1×10^2	-
4	1×10^2	1	1×10^{-2}	-
5	1×10^3	1×10	1×10^2	-
6	1×10^3	1×10^2	1×10^{-2}	-
7	10	10	1×10^{-2}	Tidak Stabil
8	10	10	1×10^{-1}	Tidak Stabil
9	10	10	1	Tidak Stabil

Penentuan nilai *adaptive rate* yang kurang tepat akan menyebabkan kegagalan pengujian dalam simulasi menggunakan *Matlab/Simulink* seperti yang terjadi pada variasi 7, 8, dan 9. Hal ini terjadi karena *solver* pada *Matlab/Simulink* menemukan perhitungan yang nilainya tidak terhingga pada waktu (t) tertentu sehingga simulasi dihentikan. Selanjutnya variasi yang tidak mengalami ketidakstabilan saat simulasi akan diujikan variasi 1, 3, dan 5 untuk pengujian pengaturan level dikarenakan variasi 1 dengan 2, 3 dengan 4, 5 dengan 6 menghasilkan selisih *error* yang sangat kecil.



Gambar 4. 3 Respon Tracking Set Point Level Dengan PID MRAC

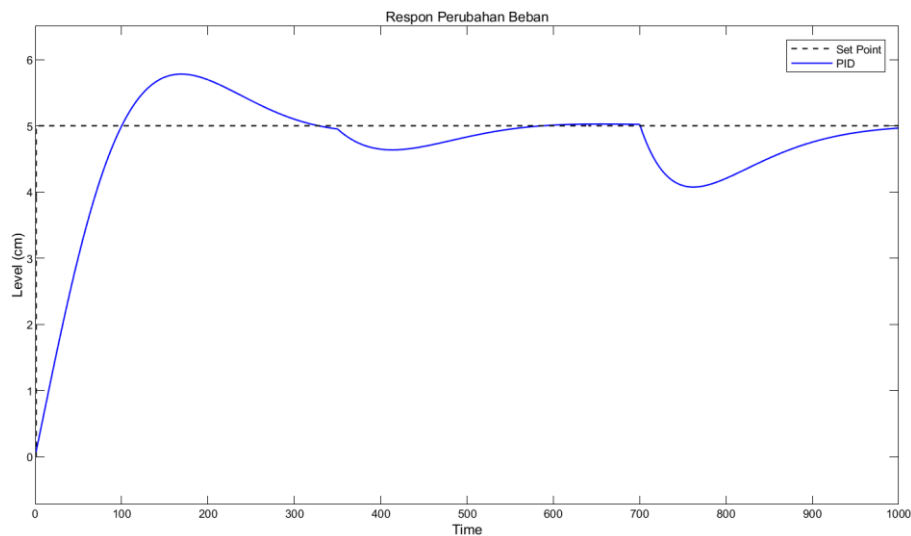
Tabel 4. 3 Hasil pengukuran Uji Berubahan Set Point Dengan PID MRAC

	Variasi 1	Variasi 3	Variasi 5
RiseTime Awal (s)	24.091	6.243	1.527
Overshoot Awal (%)	0.339	0.009	0.0096
Steady State Error Awal	0	0	0
Rise Time Adaptasi (s)	8.380	1.770	1.124
Overshoot Adaptasi (%)	0.042	0	0
Steady State Error Adaptasi	0	0	0
K_p Adaptasi (t=1000s)	2.0628	2.5279	4.0993
K_i Adaptasi (t=1000s)	-2.2387	-1.1241	-1.1846
K_d Adaptasi (t=1000s)	0	2.0093	5.0384
RMSE	0.7612	0.2075	0.0685

4.1.2 Pengujian Perubahan Beban Tanpa Noise

4.1.2.1 Kontroler PID

Pada pengujian ini, digunakan kontroler PID secara *closed loop* untuk mendapatkan respon *system*. Dengan melakukan *tuning* PID ZN pada beban kondisi 1 didapatkan nilai $K_p = 5.8$, $K_i = 0.13$, dan $K_d = 7.4$, kemudian diuji pada beban kondisi 2 dan 3. Seperti pada Gambar 4.2.



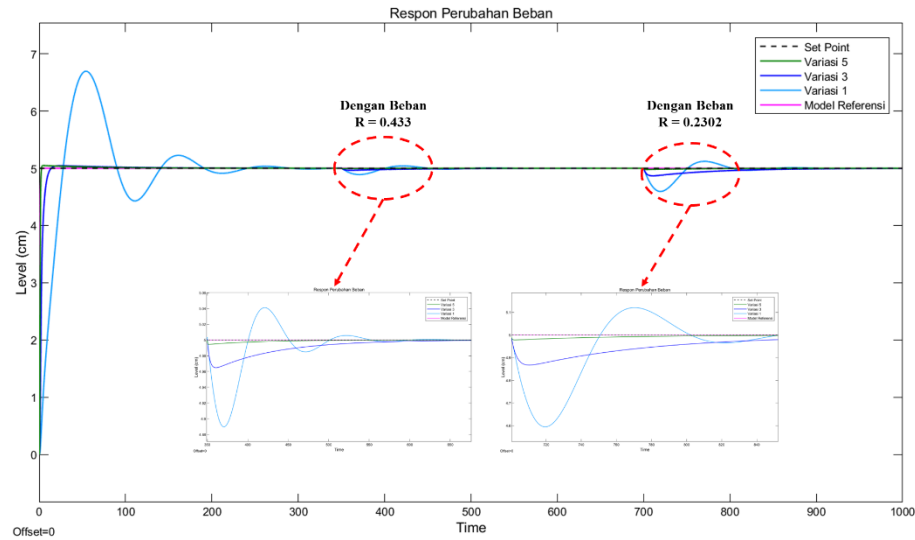
Gambar 4. 4 Respon Perubahan Beban Level Dengan PID

Tabel 4. 4 Hasil Pengukuran Uji Perubahan Beban Dengan PID

<i>RiseTime</i> (s)	83.710
<i>Overshoot</i> (%)	0.1562
<i>Undershoot</i> Beban 1 (%)	-0.0728
<i>Undershoot</i> Beban 2 (%)	-0.1852
<i>Error Steady State</i>	0
<i>RMSE</i>	1.0305

4.1.2.2 PID MRAC

Pada pengujian ini digunakan kontroler PID-MRAC dengan beban bertujuan menguji seberapa cepat kontroler mampu kembali menuju kondisi awal ketika diberikan gangguan. Selain kecepatan, besarnya lonjakan dan penurunan nilai yang terjadi akibat pembebanan juga menjadi kriteria yang dianalisis. Pembebanan dilakukan dengan mengubah nilai R atau bukaan *control valve* yang sudah dimodelkan pada Bab 3. Semakin besar bukaan *control valve* maka semakin besar pula debit air yang keluar tangki, sehingga nilai R semakin kecil. Pengujian dengan beban digunakan *adaptive rate* dengan variasi 1, 3, dan 5 untuk pengujian pengaturan level.



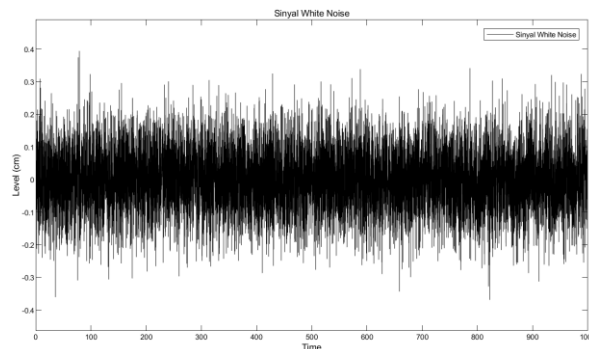
Gambar 4. 5 Respon Perubahan Beban Dengan PID MRAC

Tabel 4. 5 Hasil Pengukuran Uji Perubahan Beban Dengan PID MRAC

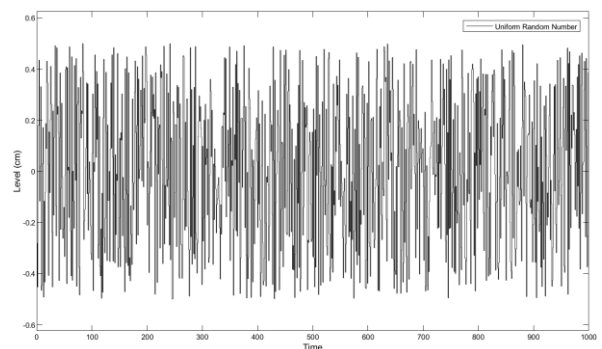
	Variasi 1	Variasi 3	Variasi 5
<i>RiseTime</i> (s)	24.091	6.243	1.527
<i>Overshoot</i> (%)	0.339	0.009	0.0096
<i>Error Steady State</i>	0	0	0
<i>Undershoot</i> Beban 1 (%)	-0.022	-0.007	-0.0012
<i>Undershoot</i> Beban 2 (%)	-0.081	-0.0266	-0.0044
K_p Adaptasi (t=1000s)	1.2319	1.9315	1.1807
K_i Adaptasi (t=1000s)	-4.0307	-6.0631	-3.0966
K_d Adaptasi (t=1000s)	1.8217	2.1914	1.6024
<i>RMSE</i>	0.5355	0.17	0.0681

4.1.3 Pengujian Perubahan Beban Dengan Noise

Pengujian dengan *noise* dilakukan dengan menambahkan sinyal *white noise* dan *random uniform* sepanjang jalannya simulasi agar menyerupai dengan kondisi saat implementasi.



Gambar 4. 6 Sinyal White Noise

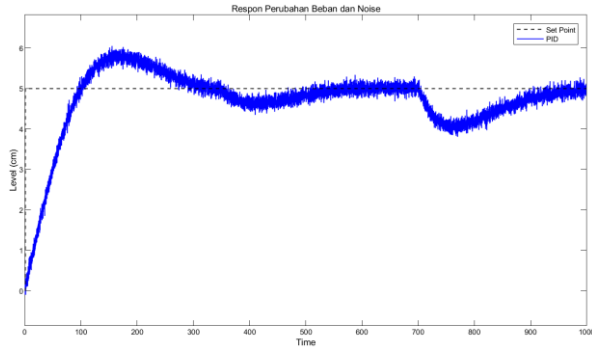


Gambar 4. 7 Sinyal Random Uniform

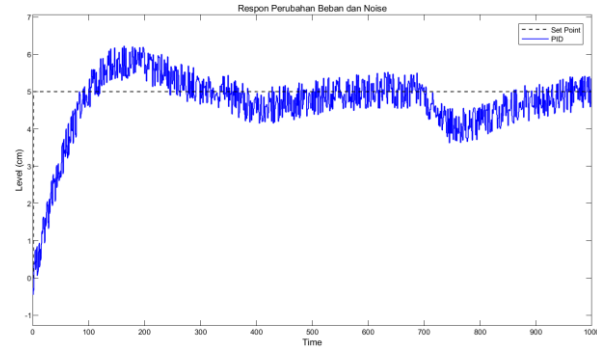
Pada Gambar 4.6 merupakan sinyal *white noise* dengan *noise power* 0.001 dan *sample time* 0.1s, dan Gambar 4.7 merupakan sinyal *random white noise* dengan nilai *min* -0,5 dan *max* 0.5 *sample time* 1s.

4.1.3.1 Kontroler PID

Pada pengujian ini, digunakan kontroler PID secara *closed loop* dengan sinyal *noise* untuk mendapatkan respon *system*. Dengan melakukan *tuning* PID ZN pada beban kondisi 1 didapatkan nilai $K_p = 5.8$, $K_i = 0.13$, dan $K_d = 7.4$, kemudian diuji pada beban kondisi 2 dan 3. Seperti pada Gambar 4.9 dan 4.10.



Gambar 4. 8 Respon PID Dengan Beban dan White Noise



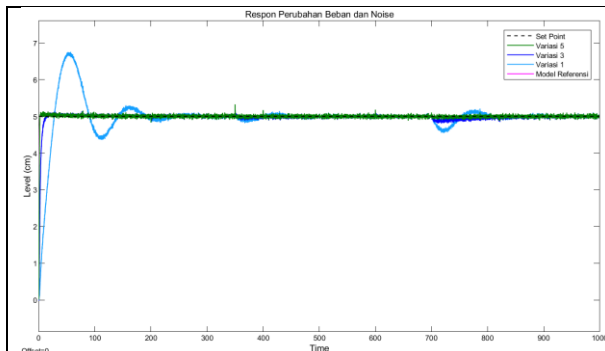
Gambar 4. 9 Respon PID Dengan Beban dan Random Uniform

Tabel 4. 6 Hasil Pengukuran Uji Perubahan Beban Menggunakan PID Dengan Noise

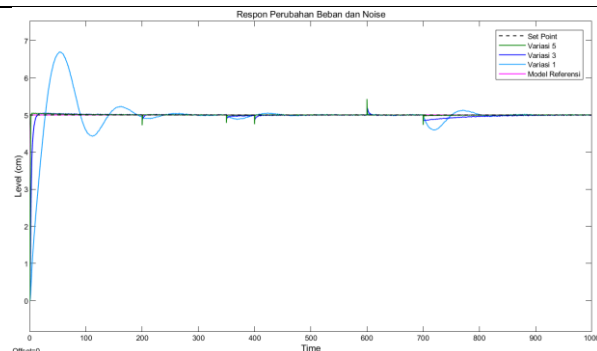
	White Noise	Random Uniform
RiseTime (s)	76.881	74.997
Overshoot (%)	0.2108	0.245
Undershoot Beban 1 (%)	-0.1302	-0.1706
Undershoot Beban 2 (%)	-0.2392	-0.276
Error Steady State	0	0
RMSE	1.0360	1.0726

4.1.3.2 PID MRAC

Pada pengujian ini digunakan kontroler PID-MRAC dengan beban bertujuan menguji seberapa cepat kontroler mampu kembali menuju kondisi awal ketika diberikan gangguan dan diberi *noise* agar menyerupai dengan kondisi saat implementasi. Pengujian dengan beban digunakan *adaptive rate* dengan variasi 1, 3, dan 5 untuk pengujian pengaturan level.



Gambar 4. 10 Respon PID MRAC Dengan Beban dan White Noise



Gambar 4. 11 Respon PID MRAC Dengan Beban dan Random Uniform

Tabel 4. 7 Hasil Pengukuran Uji Perubahan Beban Menggunakan PID MRAC Dengan Noise

	White Noise			Random Uniform		
	Variasi 1	Variasi 3	Variasi 5	Variasi 1	Variasi 3	Variasi 5
<i>RiseTime</i> (s)	24.151	6.034	1.573	24.047	6.217	1.526
<i>Overshoot</i> (%)	0.351	0.0252	0.0272	0.3394	0.0102	0.0106
<i>Error Steady State</i>	0	0	0	0	0	0
<i>Undershoot</i> Beban 1 (%)	-0.31	-0.0011	-0.0017	-0.35	-0.0014	-0.0021
<i>Undershoot</i> Beban 2 (%)	-4.0337	-6.0671	-0.0049	-4.0347	-6.0679	-0.0054
K_p Adaptasi (t=1000s)	5.053	2.8971	5.774	1.0087	1.0498	-3.1413
K_i Adaptasi (t=1000s)	-5.8434	1.6943	2.7746	-1.3151	-2.1415	1.0188
K_d Adaptasi (t=1000s)	2.737	1.614	4.0972	7.8599	9.3366	1.1277
<i>RMSE</i>	0.5359	0.1744	0.0810	0.5335	0.1752	0.0823

4.2 Pembahasan

4.2.1 Tanpa Kontroler

Berdasarkan hasil pengujian pada subbab 4.1.1, tidak adanya kontroler respon tidak mampu mencapai *set point* sehingga terdapat *error steady state* sebesar 2.899 cm. Dengan adanya permasalahan tersebut yang tidak adanya kontroler, sistem tidak dapat menyelesaikan permasalahan sehingga perlu digunakan kontroler untuk meminimalkan *error steady state* pada sistem.

4.2.2 Kontroler PID

Berdasarkan hasil pengujian PID pada subbab 4.1.1, 4.1.2, dan 4.1.3 didapatkan rekap data untuk semua pengujian pengaturan level dengan kontroler PID seperti pada Tabel 4.9.

Tabel 4. 8 Hasil Pengukuran Uji Perubahan Beban Dengan PID

	<i>Tracking Set Point</i>	Dengan Beban		
		Tanpa Noise	Noise	
			White Noise	Random Uniform
<i>RiseTime</i> (s)	39.889	83.710	76.881	74.997
<i>Overshoot</i> (%)	0.1988	0.1562	0.2108	0.245
<i>Undershoot</i> Beban 1 (%)	-	-0.0728	-0.1302	-0.1706
<i>Undershoot</i> Beban 2 (%)	-	-0.1852	-0.2392	-0.276
<i>Error Steady State</i>	0	0	0	0
<i>RMSE</i>	1.5473	1.0305	1.0360	1.0726

Berdasarkan hasil pengujian pada Tabel 4.9 maka kontroler PID mampu mencapai *set point* pada pengujian *tracking set point*, namun dengan hanya menggunakan kontroler PID menghasilkan nilai RMSE 1.5473. Pada Pengujian Beban tanpa noise menghasilkan RMSE 1.0305 dan dengan noise berupa white noise dan random uniform menghasilkan RMSE 1.036 dan 1.0726.

4.2.3 Kontroler PID MRAC

Berdasarkan hasil pengujian PID MRAC pada subbab 4.1.1, 4.1.2, dan 4.1.3 didapatkan rekap data untuk semua pengujian pengaturan level dengan kontroler PID seperti pada Tabel 4.10, 4.11, dan 4.12.

Tabel 4. 9 Hasil Pengukuran Uji Dengan PID MRAC

	Tracking Set Point			Dengan Beban								
				Tanpa Noise			Noise					
							White Noise			Random Uniform		
	Variasi 1	Variasi 3	Variasi 5	Variasi 1	Variasi 3	Variasi 5	Variasi 1	Variasi 3	Variasi 5	Variasi 1	Variasi 3	Variasi 5
<i>RiseTime</i> Awal (s)	24.091	6.243	1.527	24.091	6.243	1.527	24.151	6.034	1.573	24.047	6.217	1.526
<i>Overshoot</i> Awal (%)	0.339	0.009	0.0096	0.339	0.009	0.0096	0.351	0.0252	0.0272	0.3394	0.0102	0.0106
<i>Steady State Error</i> Awal	0	0	0	0	0	0	0	0	0	0	0	0
<i>Rise Time</i> Adaptasi (s)	8.380	1.770	1.124	-	-	-	-	-	-	-	-	-
<i>Overshoot</i> Adaptasi (%)	0.042	0	0	-	-	-	-	-	-	-	-	-
<i>Steady State Error</i> Adaptasi	0	0	0	-	-	-	-	-	-	-	-	-
<i>Undershoot</i> Beban 1 (%)	-	-	-	-0.022	-0.007	-0.0012	-0.31	-0.0011	-0.0017	-0.35	-0.0014	-0.0021
<i>Undershoot</i> Beban 2 (%)	-	-	-	-0.081	-0.027	-0.004	-4.034	-6.067	-0.005	-4.0347	-6.0679	-0.0054
K_p Adaptasi	2.0628	2.5279	4.0993	1.2319	1.9315	1.1807	5.053	2.8971	5.774	1.0087	1.0498	-3.141
K_i Adaptasi	-2.239	-1.124	-1.185	-4.031	-6.063	-3.097	-5.843	1.6943	2.7746	-1.315	-2.142	1.019
K_d Adaptasi	0	2.0093	5.0384	1.8217	2.1914	1.6024	2.737	1.614	4.0972	7.8599	9.3366	1.1277
<i>RMSE</i>	0.7612	0.2075	0.0685	0.5355	0.17	0.0681	0.5359	0.1744	0.0810	0.5335	0.1752	0.0823

Berdasarkan hasil pengujian pada Tabel 4.10 maka kontroler PID MRAC mampu mencapai *set point* dan beradaptasi tiap kenaikan *set point* dan dengan beban baik tanpa maupun dengan *noise*. Nilai *adaptive rate* pada kontroler PID-MRAC untuk pengaturan level yang terbaik adalah pada variasi 5 pada semua kondisi beban. Pada variasi ini nilai γ_p bernilai 1×10^3 , γ_i bernilai 1×10 , dan γ_d bernilai 1×10^2 . Dalam pengujian *tracking set point* nilai RMSE sebesar 0.0685. Pada pengujian dengan beban tanpa *noise* nilai RMSE 0.0681, kemudian dengan *white noise* dan *random uniform* nilai RMSE 0.0810 dan 0.0823. Berdasarkan hasil pengujian didapatkan bahwa kontroler PID-MRAC memiliki performansi yang lebih bagus dibandingkan jika hanya menggunakan rangkaian tanpa kontroler (*closed loop*) ataupun dengan kontroler PID biasa. Kontroler PID-MRAC menghasilkan selisih yang sedikit saat diberi *noise* seperti pada Tabel 4.9. Kemampuan adaptasi nilai parameter K_p, K_i dan K_d yang membuat RMSE pada kontroler PID-MRAC lebih baik dibandingkan dengan kontroler PID dan tanpa kontroler (*closed loop*).

BAB 5 Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil penelitian yang didapat dari simulasi dan analisa data yang telah dilakukan dalam perancangan kontroler PID-MRAC pada PCT-100 ini, maka didapatkan kesimpulan sebagai berikut.

1. Pemodelan sistem pengaturan level dapat berubah dikarenakan adanya variasi kondisi beban yang disebabkan nilai R (resistansi *control valve*) dipengaruhi oleh Q_{out} (debit air yang keluar dari tangki) dan h (ketinggian air dalam tangki) yang berubah pada tiap variasi kondisi beban.
2. Perbandingan pengujian tanpa kontroler respon tidak mampu mencapai *set point*. Kontroler PID tanpa noise menghasilkan respon yang mampu mencapai set point namun tidak dapat beradaptasi di tiap kenaikan *set point* dan menghasilkan nilai RMSE 1.5473. Pada pengujian beban tanpa noise menghasilkan RMSE 1.0305 dan dengan noise berupa white noise dan random uniform menghasilkan RMSE 1.036 dan 1.0726. Kontroler PID MRAC menghasilkan respon yang mampu mencapai *set point* dan dapat beradaptasi di tiap kenaikan step baik dengan dan tanpa *noise* dan menghasilkan *error* yang kecil
3. Perancangan kontroler PID-MRAC pada pengaturan level memiliki performansi yang lebih bagus dibandingkan dengan rangkaian tanpa menggunakan kontroler (*closed loop*) dan kontroler PID. Hal ini disebabkan ketika skenario simulasi mendekati sesungguhnya dengan menggunakan tambahan *noise*, kontroler PID MRAC bekerja lebih baik karena semakin banyak perubahan *error* terjadi akan membuat kontroler PID MRAC semakin beradaptasi menyesuaikan nilai K_p , K_i , dan K_d sehingga menghasilkan selisih *error* yang kecil. Pada pengujian ini dengan nilai awal $K_p = 1$, $K_i = 0$, dan $K_d = 0$ dengan model referensi orde dua dengan parameter $\alpha = 1$, $\omega_b = 2$, dan $\zeta = 1$ yang memiliki settling time (5%) selama 3 detik didapatkan parameter *adaptive rate* terbaik pada variasi 5 dengan nilai $\gamma_p = 1 \times 10^3$, $\gamma_i = 1 \times 10$, dan $\gamma_d = 1 \times 10^2$ untuk pengujian *tracking set point* menghasilkan nilai RMSE sebesar 0.0685. Pada pengujian dengan beban tanpa noise menghasilkan nilai RMSE 0.0681, kemudian dengan white noise dan random uniform nilai RMSE 0.0810 dan 0.0823. Nilai RMSE pada kontroler PID-MRAC ini lebih baik dibandingkan dengan rangkaian tanpa menggunakan kontroler (*closed loop*) dan kontroler PID.

5.2 Saran

Sebagai bentuk pengembangan lebih lanjut terhadap topik penelitian ini, terdapat beberapa saran yang diberikan antara lain.

1. Masih Banyak metode kontrol yang belum di lakukan pada pengaturan *level PCT-100*, sehingga dapat digunakan metode kontrol lain untuk pengaturan *level PCT-100*
2. Dapat dilakukan penyempurnaan pada rangkain *Data Acquisition STM32* supaya lebih proper saat penggunaan pengambilan data *PCT-100*

DAFTAR PUSTAKA

- [1] Pan, Yongping, dan Qinruo Wang, "Research on a Stable Adaptive Fuzzy Control of Nonlinear Liquid Level System", Sixth Internatioanl Conference on Hybrid Intelligent Systems (HIS), pp. 65, Rio de Janeiro, December 2006.
- [2] Wu, Yifeng, dan Xiaoqi Lin, "Reserach of Boiler Fault Diagnosis Based on Fuzzy Neural Network", Internatioanl Conference on Information Science and Computer Application (ISCA).2013
- [3] Chen, Liang, Cuizhu Wang, Yang Yu, dan Yawei Zhao, "The research on Boiler Drum Water Level Control System Based on Self-Adaptive Fuzzy PID", Control and Decision Conference (CCDC), pp. 1582 – 1584, Xuzhou, Chinese, May 2010.
- [4] Mohamad Abdul Hady, Mohamad Yusuf, Ali Fatoni, Imam Arifin, "Self-Tuning PID-Genetic Algorithm Controller for Steam Drum Boiler Water Level Control System", Journal on Advanced Research in Electrical Engineering, Vol. 5, No. 2, Oct. 2021
- [5] Lin, Feng dan Robert D Brandt, and George Saikalis, "Self-tuning od PIF Controllers by Adaptive Interaction", Proceedings of the American Control Conference, pp. 3676 – 3681 vol.5, Chicago, Illonis, June 2000.
- [6] Ehsani, M., "Adaptive control by MRAC method", Proceodings OCEANS 2003, vol 1, San Diego, September 2003
- [7] 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).
- [8] Ghuftron, M. (2016). DESAIN DAN IMPLEMENTASI KONTROLER SELF-TUNING PID DENGAN PENDEKATAN INTERAKSI ADAPTIF PADA SISTEM PENGATURAN LEVEL. Institut Teknologi Sepuluh Nopember.
- [9] Rahmadhani, K. F. (2016). PERANCANGAN KONTROLER PID DENGAN METODE PARTICLE SWARM OPTIMIZATION UNTUK SISTEM PENGATURAN CASCADE LEVEL DAN FLOW PADA PLANT PCT-100. Institut Teknologi Sepuluh Nopember.
- [10] Ogata, K. (2010). Modern Control Engineering 5th Edition. In Prentice Hall PTR (5th ed.).
- [11] Jakoubek, P. (2009). Experimental Identification of Stabile Nonoscillatory Systems from Step-Responses by Selected Methods. Student Conderence STC 2009.
- [12] Xiong, A., & Fan, Y. (2007). Application of a PID controller using MRAC techniques for control of the DC electromotor drive. Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation, ICMA 2007, 2616–2621.
- [13] Anstrom, K. J., & Wittenmark, B. (2013). Adaptive Control: Second Edition (2nd ed.). Dover Publications, Inc.

- [14] Chai, T., & Draxler, R. R. (2014). Root mean square error (RMSE) or mean absolute error (MAE)? -Arguments against avoiding RMSE in the literature. *Geoscientific Model Development*, 7(3), 1247–1250.

LAMPIRAN

I. Pemodelan Flow Meter

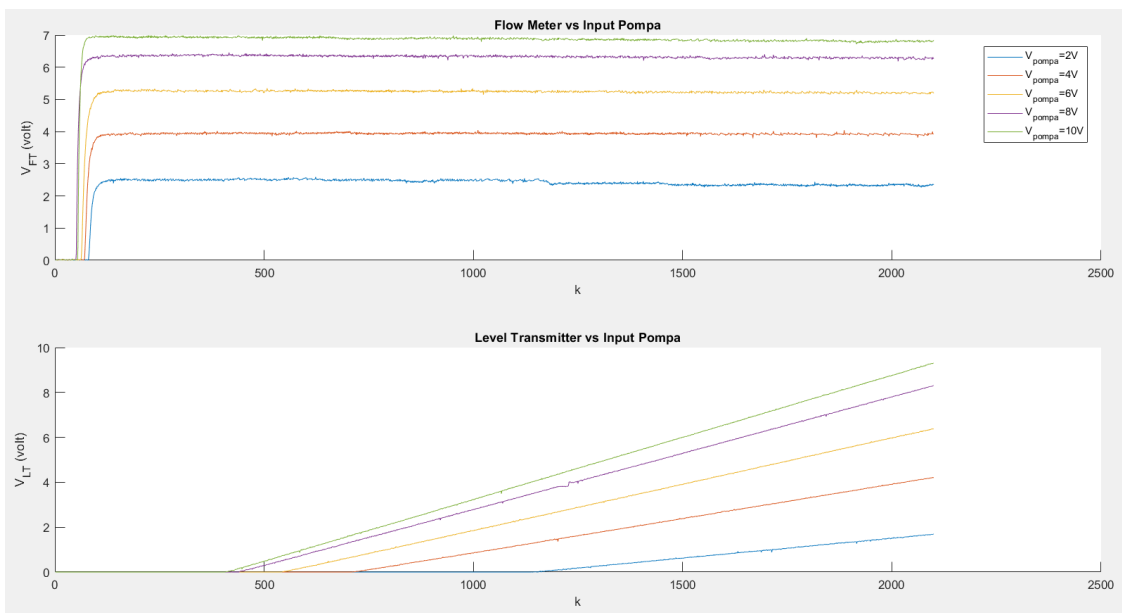
$$Q_{in} = \frac{\Delta V}{\Delta t} = A_{tabung} \frac{\Delta h}{\Delta t}$$

$$Q_{in} = m_{FT} V_{FT}$$

$$m_{FT} V_{FT} = A_{tabung} \frac{\Delta h}{\Delta t}$$

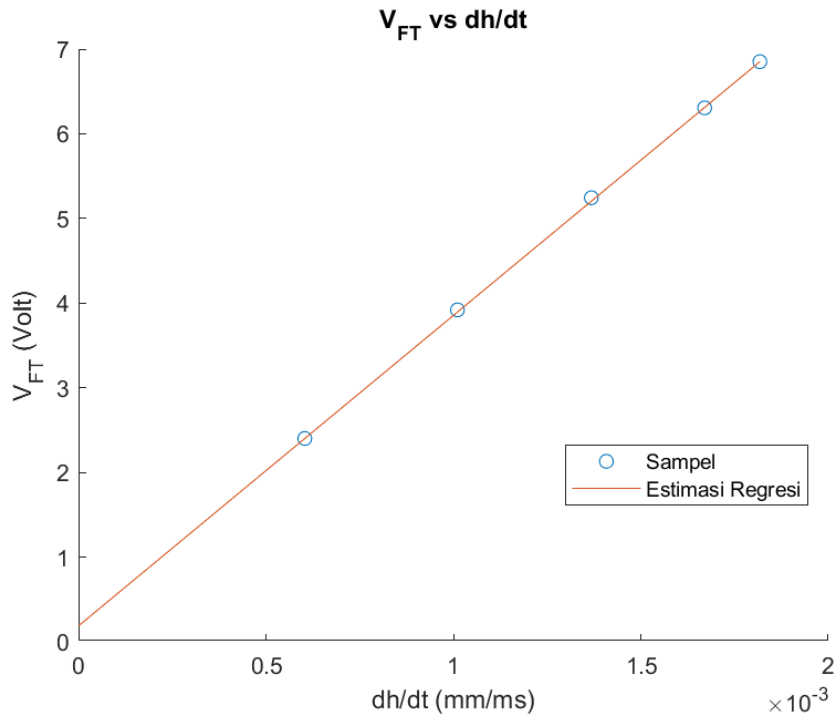
$$V_{FT}(h) = \frac{A_{tabung}}{m_{FT}} \frac{\Delta h}{\Delta t}$$

$$V_{FT}(h) = G_{FT} \frac{\Delta h}{\Delta t}$$



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

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



Gambar Respon tegangan flow meter terhadap laju level tangki.

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

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

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

$$\text{Diketahui } A_{tabung} = \pi r_{tabung}^2 = 8100\pi \text{ mm}^2$$

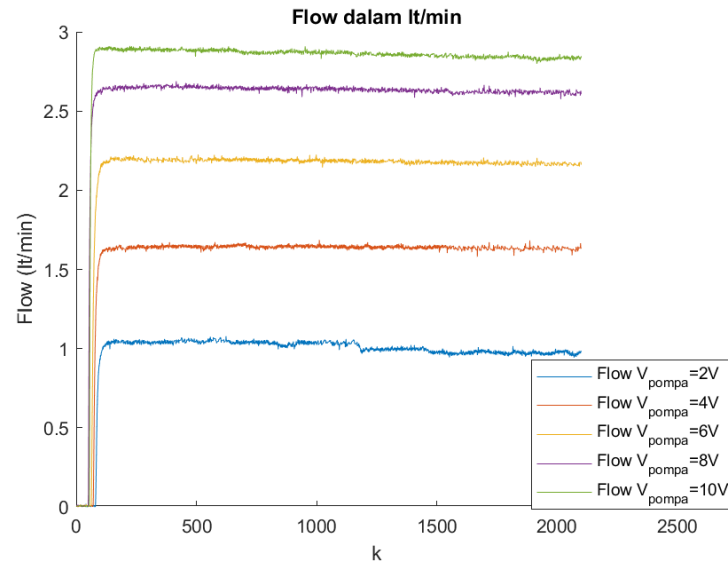
$$m_{FT} = \frac{A_{tabung}}{G_{FT}}$$

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

$$V_{FT} = \frac{Q_{in} \left(\frac{\text{mm}^3}{\text{ms}} \right)}{m_{FT} \left(\frac{\text{mm}^3}{\text{Volt.ms}} \right)}$$

$$Q_{in} = m_{FT} V_{FT} \text{ dalam } \frac{\text{mm}^3}{\text{ms}}$$

$$Q_{in} = m_{FT} V_{FT} (6 \times 10^{-2}) \text{ dalam lt/min}$$



Gambar Respon pembacaan flow meter terhadap variasi tegangan pompa

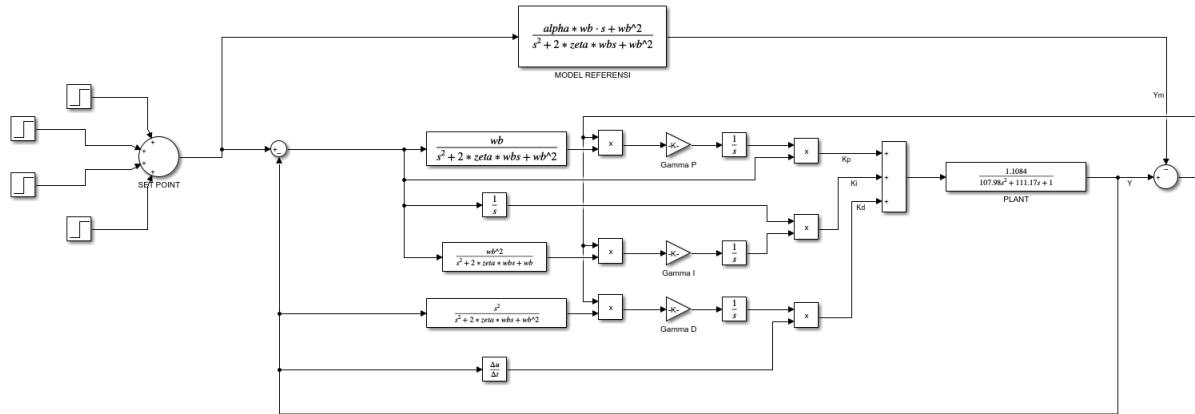
II. Percobaan Menentukan Adaptive Rate

Adaptasi K_p	Adaptasi K_i	Adaptasi K_d	RMSE	
			Tracking Set Point	Dengan Beban
10	0.01	0.01	0.8311	0.5516
		0.1	0.8311	0.5516
		1	0.8305	0.5514
		10	0.8247	0.55
		100	0.7612	0.5355
	0.1	0.01	0.7564	0.5562
		0.1	0.7564	0.5562
		1	0.7546	0.5562
		10	0.7385	0.5525
		100	0.6127	0.5178
	1	0.01	0.6043	0.4891
		0.1	0.604	0.4891
		1	0.6008	0.4882
		10	0.5688	0.4796
		100	0.4753	ts
	10	0.01	0.4812	0.4003
		0.1	0.4808	0.4001
		1	0.477	0.3989
		10	0.4395	0.3865
		100	ts	ts
	100	0.01	0.4037	0.3186
		0.1	0.4034	0.3185
		1	0.3999	0.3174
		10	0.3665	0.3066
		100	0.363	0.3444

100	0.01	0.01	0.2752	0.2084
		0.1	0.2751	0.2084
		1	0.2748	0.2083
		10	0.2727	0.209
		100	0.2371	0.2005
	0.1	0.01	0.2585	0.1829
		0.1	0.2585	0.1829
		1	0.2581	0.1828
		10	0.2552	0.1833
		100	0.2075	0.17
	1	0.01	0.2604	0.1925
		0.1	0.2603	0.1925
		1	0.2598	0.1923
		10	0.2557	0.1918
		100	0.2026	0.1745
	10	0.01	0.2357	0.1937
		0.1	0.2356	0.1937
		1	0.235	0.1936
		10	0.2299	0.1929
		100	0.185	0.1745
	100	0.01	0.1988	0.1675
		0.1	0.1987	0.1674
		1	0.1981	0.1673
		10	0.1934	0.1668
		100	0.163	0.151
1000	0.01	0.01	0.0899	0.0773
		0.1	0.0899	0.0773
		1	0.0899	0.0772
		10	0.0911	0.0782
		100	0.0933	0.0753
	0.1	0.01	0.0842	0.0755
		0.1	0.0842	0.0755
		1	0.0841	0.0754
		10	0.0853	0.0765
		100	0.0864	0.0739
	1	0.01	0.0717	0.0706
		0.1	0.0716	0.0706
		1	0.0715	0.0705
		10	0.0726	0.0717
		100	0.0712	0.0692
	10	0.01	0.0712	0.0702
		0.1	0.0712	0.0702
		1	0.0711	0.0701
		10	0.0718	0.071
		100	0.0685	0.0681
	100	0.01	0.0773	0.0733
		0.1	0.0773	0.0733

		1	0.0772	0.0732
		10	0.0779	0.074
		100	0.0755	0.0711

III. Simulink PID MRAC



BIODATA PENULIS



Muhammad Usama Sahal, lahir di Surabaya, 11 Oktober 2001, merupakan anak kedua dari 4 bersaudara. Penulis telah menempuh pendidikan formal yaitu di TK Sepuluh Nopember Surabaya, SD Luqman Al Hakim Surabaya, SMPN 39 Surabaya dan SMAN 20 Surabaya. Setelah lulus dari SMAN tahun 2019, Penulis mengikuti PROGRAM MANDIRI dan diterima di Departemen Teknik Elektro FTEIC - ITS pada tahun 2019 dan terdaftar dengan NRP 07111940000176.

Di Departemen Teknik Elektro Penulis sempat aktif di beberapa kegiatan Seminar yang diselenggarakan oleh Departemen maupun di luar Departemen, Kepanitiaan ELECTRA 9 2020, Kepanitiaan LKMM Pra TD 2021, Kepanitiaan LKMM TD 2022, dan aktif sebagai STAFF PSDK UKM KARATE-DO ITS 2020. Penulis dapat dihubungi melalui kontak email : muhammadusamasahal@gmail.com.