



TUGAS AKHIR - TE 141599

**PERANCANGAN KONTROLER *NEURAL NETWORK PI* UNTUK
SISTEM PENGATURAN *LEVEL* DENGAN METODE
DECOUPLING PADA *PLANT COUPLED TANK*.**

Kresna Kartika Surya Putra
NRP. 2212 100 066

Dosen Pembimbing
Ir. Rusdhianto Effendi A.K, M.T.

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2017



FINAL PROJECT - TE 141599

***DESIGNING NEURAL NETWORK PI CONTROLLER TO
CONTROL LEVEL WITH DECOUPLING SYSTEM IN PLANT
COUPLED TANKS***

Kresna Kartika Surya Putra
NRP. 2212 100 066

Supervisor
Ir. Rusdhianto Effendi A.K, M.T.

*DEPARTEMENT OF ELECTRICAL ENGINEERING
Faculty of Industrial Technology
Sepuluh Nopember Insitute of Technology
Surabaya 2017*

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan tugas akhir saya dengan judul **“Perancangan Kontroler Neural Network PI Untuk Sistem Pengaturan Level Dengan Metode Decoupling Pada Plant Coupled Tank.”** adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diijinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka.

Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, Januari 2017

Kresna Kartika Surya Putra
NRP. 2212 100 066

Halaman ini sengaja dikosongkan

**PERANCANGAN KONTROLER NEURAL NETWORK PI UNTUK
SISTEM PENGATURAN LEVEL DENGAN METODE
DECOUPLING PADA PLANT COUPLED TANK**

TUGAS AKHIR

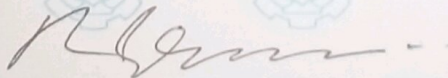
**Diajukan Guna Memenuhi Sebagian Persyaratan Untuk
Memperoleh Gelar Sarjana Teknik Elektro
Pada**

**Bidang Studi Teknik Sistem Pengaturan
Jurusan Teknik Elektro
Institut Teknologi Sepuluh Nopember**

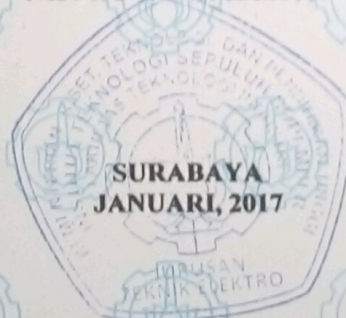
LEMBAR PENGESAHAN

Menyetujui

Dosen Pembimbing,



Ir. Rusdhianto Effendie A.K. M.T
NIP. 195704241985021001



Halaman ini sengaja dikosongkan

Perancangan Kontroler *Neural Network PI* Untuk Sistem Pengaturan *Level* Dengan Metode *Decoupling* Pada *Plant Coupled Tank*

Kresna Kartika Surya Putra
2212 100 066

Dosen Pembimbing : Ir. Rusdhianto Effendi A.K, M.T.
NIP : 195704241985021001

ABSTRAK

Sistem pengendalian pada industri merupakan faktor yang sangat penting dalam proses produksi. Coupled tank merupakan salah satu sistem yang memiliki peran vital . *Sistem nonlinear dari pemodelan coupled tank* mempengaruhi kestabilan sistem. Selain itu konfigurasi sistem *Multi Input Multi Output* membuat variabel yang diatur mengalami kesulitan untuk mengikuti sinyal referensi. Dekopling digunakan untuk mengubah dimana *Coupled tank* yang bersifat *MIMO (Multi Input Multi Output)* menjadi *SISO (Single Input Single Output) neural network* digunakan untuk mencari nilai K_p dan K_i yang optimal berdasarkan hasil simulasi dengan model linier didapatkan nilai K_p 13,0031 dan nilai K_i 0,6013 untuk tangki 1 dan untuk H_2 didapatkan nilai K_p 13,072 dan K_i sebesar 1,0121, kemudian hasil simulasi dengan model nonlinier didapatkan rata rata *error steady state* sebesar 0,025% untuk tangki 1 dan untuk tangki 2 sebesar 0,033% , kemudian menghasilkan nilai K_p optimal sebesar 12,983 dan K_i sebesar 0,527 untuk *level* tangki 1, kemudian untuk *level* tangki 2 mendapat nilai K_p optimal sebesar 13,071 dan K_i optimal sebesar 1,18

Kata Kunci : *Coupled tank*, dekopling, PI, *Neural Network*

Halaman ini sengaja dikosongkan

Designing Neural Network PI Controller To Control Level With Decoupling System In Plant Coupled Tank

Kresna Kartika Surya Putra
2212 100 066

Supervisor : Ir. Rusdhianto Effendi A.K, M.T.
ID Number : 195704241985021001

ABSTRACT

The control system is the important factor in industrial process production. Coupled tank one of the vital subsystem in industry. Nonlinear systems modeling contained in coupled tanks affect the stability of the system. In addition the system configuration Multi Input Multi Output create a variable that is set a challenge to keep the reference signal. Decoupling is used to set the MIMO configuration and control PI kontroler is used to set the level in the coupled tanks. Neural Network is used to best get the optimal parameter for K_p and K_i , from the simulation we get the parameter of K_p and K_i with the value of $K_p=13.0031$ and $K_i=0,6013$ for tank 1 then $K_p=13.072$ and $K_i=1.0121$ for tank 2. Simulation result for nonlinear model still has error steady state with value 0.0255% for tank 1 and for tank 2 the value is 0.033%, then the optimal parameter $K_p=12.983$ and $K_i=0.527$ for tank 1 then the optimal parameter $K_p=13.071$ and $K_i=1.18$ for tank 2

Keywords : Coupled Tank, Decoupling MIMO, PI Controler, Neural Network.

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Alhamdulillah Robbil 'Alamin, terucap syukur kehadiran Allah atas limpahan rahmat dan karuninya serta sholawat serta salam terhadap Nabi besar Muhammad SAW sehingga penulis dapat menyelesaikan Tugas Akhir ini yang berjudul **“Perancangan Kontroler Neural Network PI untuk Sistem Pengaturan Level dengan Metode Decoupling pada Plant coupled tanks”** guna memenuhi persyaratan untuk mencapai Gelar Sarjana Teknik pada Bidang Studi Teknik Sistem Pengaturan, Jurusan Teknik Elektro Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS) Surabaya.

Dalam kesempatan ini penulis ingin mengucapkan terimakasih kepada pihak-pihak yang banyak berjasa terutama dalam penyusunan Tugas Akhir ini, yaitu:

1. Segenap keluarga besar terutama ibu Dra Rahayu Sulistyaningrum, ayah Drs Heri Suryoko, dan kakak tercinta Wendy Surya Pratama yang selalu memberi dukungan, semangat, dan doa untuk keberhasilan penulis.
2. Bapak Ir. Rusdhianto Effendi AK MT.
3. Seluruh dosen bidang studi Teknik Sistem Pengaturan dan Jurusan Teknik Elektro atas pendidikannya dan ilmunya.
4. Seluruh rekan asisten B 405 dan atas dukungan, semangat, bantuan, kebersamaan, dan kerjasamanya selama ini.
5. Terima kasih untuk Rheza Qashmal atas bantuan dan ilmunya
6. Teruntuk Rizal, mas Anas, Anggarjuna, Adi bantuannya selama pengerjaan Tugas akhir ini
7. Seluruh rekan E-52 atas dukungan, kebersamaan dan kerjasamanya selama ini.
8. Serta pihak-pihak yang tidak dapat disebutkan satu persatu atas dukungannya terhadap penulis.

Laporan Tugas Akhir ini masih jauh dari kata sempurna, oleh karena itu penulis mengharapkan kritik dan saran dari pembaca. Semoga buku Tugas Akhir ini dapat memberikan manfaat bagi pembaca sebagai acuan penelitian selanjutnya.

Surabaya, Januari 2017

Penulis

Halaman ini sengaja dikosongkan

DAFTAR ISI

PERNYATAAN KEASLIAN TUGAS AKHIR.....	iii
LEMBAR PENGESAHAN	v
ABSTRAK.....	vii
<i>ABSTRACT</i>	ix
KATA PENGANTAR	xi
DAFTAR ISI.....	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL.....	xvii
BAB 1 PENDAHULUAN	
1.2 Perumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Sistematika Penulisan	3
1.6 Relevansi.....	3
BAB 2 TINJAUAN PUSTAKA	
2.1 <i>Coupled Tank Plant</i>	5
2.1.1 Karakteristik <i>Plant</i>	7
2.2 Pemodelan Sistem	9
2.3 <i>Decoupling</i>	11
2.4 Kontroler PID	14
2.4.1 Kontroler <i>Proportional Integral</i> (PI)	15
2.5 Jaringan Syaraf Tiruan (<i>Artificial Neural Network</i>)	17
2.5.1 Konsep Dasar Pemodelan <i>NeuralNetwork</i>	18
2.5.2 Learning pada <i>Neural Network</i>	19
BAB 3 PERANCANGAN SISTEM	
3.1 Pemodelan Sistem.....	21
3.2 Model Matematika <i>Coupled Tank</i>	21
3.3 <i>Transfer Function Coupled Tank</i>	25
3.4 Desain <i>Decoupling</i>	26
3.5 Perancangan Kontroler PI <i>Neural Network</i>	30
BAB 4 PENGUJIAN DAN ANALISIS	
4.1 Simulasi Validasi Model Linier	35
4.2 Simulasi Model Linier tanpa Kontroler	36
4.2.1 Simulasi Model Linier tanpa Dekopling	36
4.2.2 Simulasi Model Linier dengan Dekopling	38
4.3 Simulasi Model Linier dengan Kontroler <i>Neural Network</i> PI.	39

4.3.1	Simulasi Model Linier dengan Kontroler tanpa Dekopling	39
4.3.2	Simulasi Model Linier dengan Kontroler dan Dekopling	40
4.3.3	Simulasi Model Linier dengan Dekopling <i>Setpoint</i> yang Berubah Ubah	42
4.4	Simulasi Sistem Coupled Tank Model Nonlinier Kontroler <i>Neural Network PI</i>	43
4.4.1	Simulasi Model Non linier tanpa Beban	43
4.4.2	Simulasi Model Nonlinier dengan Beban Statis	44
4.4.3	Simulasi Model Nonlinier dengan Beban Random	46
BAB 5 PENUTUP		
5.1	Kesimpulan	49
5.2	Saran	49
DAFTAR PUSTAKA		51
LAMPIRAN		53

DAFTAR GAMBAR

Gambar 2.1 Sistem <i>Coupled Tank</i>	6
Gambar 2.2 Diagram Blok Sistem Pengaturan Proses	7
Gambar 2.3 Diagram Blok Sistem Pengaturan Konfigurasi <i>Cascade</i>	8
Gambar 2.4 Sistem Dekopling pada <i>Coupled Tank</i>	12
Gambar 2.5 Blok Diagram Kontroler PID	14
Gambar 2.6 Blok Diagram Kontroler PI	15
Gambar 2.7 Sel Syaraf Manusia.....	18
Gambar 2.8 Model Neuron	18
Gambar 3.1 Respon Nonlinier untuk Pemodelan Linier G_{11}	23
Gambar 3.2 Respon Nonlinier untuk Pemodelan Linier G_{12}	24
Gambar 3.3 Respon Nonlinier untuk Pemodelan Linier G_{22}	24
Gambar 3.4 Respon Nonlinier untuk Pemodelan linier G_{21}	25
Gambar 3.5 Hasil pemodelan Linier dengan Pendekatan Respon <i>Steady State Close Loop</i> Model Nonlinier.....	27
Gambar 3.6 Hasil dari <i>Decoupling</i> untuk <i>Plant</i>	28
Gambar 3.7 Hasil Respon setelah <i>Decoupling</i> pada Tangki 1	29
Gambar 3.8 Hasil Respon setelah <i>Decoupling</i> pada Tangki 2	29
Gambar 3.9 Struktur <i>Neural Network</i> untuk <i>Autotuning</i> PI.....	31
Gambar 4.1. Respon antara Model Nonlinear dan Linear untuk Tangki 1	35
Gambar 4.2. Respon antara Model Nonlinear dan Linear untuk Tangki 2	36
Gambar 4.3 Respon Linier tanpa Dekopling Tangki 1.....	37
Gambar 4.4 Respon Linier tanpa Dekopling Tangki 2.....	37
Gambar 4.5 Respon Linier dengan Dekopling Tangki 1.....	38
Gambar 4.6 Respon Linier dengan Dekopling Tangki 2.....	38
Gambar 4.7 Hasil Respon Tangki 1 dengan Perubahan <i>Setpoint</i> pada Tangki 2	39
Gambar 4.8 Hasil Respon Tangki 2 dengan Perubahan <i>Setpoint</i> pada Tangki 1	40
Gambar 4.9 Hasil Respon Tangki 1 dengan Perubahan <i>Setpoint</i> pada Tangki 2	40
Gambar 4.10 Hasil Respon Tangki 2 dengan Perubahan <i>Setpoint</i> pada Tangki 1	41
Gambar 4.11 Respon Tangki 1 dengan <i>Setpoint</i> yang Berubah Ubah ...	42
Gambar 4.12 Respon Tangki 2 dengan <i>Setpoint</i> yang Berubah Ubah....	42

Gambar 4.13	Respon Simulasi Nonlinier pada Tangki 1 dengan <i>Decoupling</i>	43
Gambar 4.14	Hasil Simulasi Nonlinier pada H ₂ dengan <i>Decoupling</i>	44
Gambar 4.15	Respon Tangki 1 dengan Pembebanan Statis 0,08	45
Gambar 4.16	Hasil Respon H ₂ dengan Pembebanan Statis 0,08	45
Gambar 4.17	Hasil Respon Tangki 1 dengan Pembebanan <i>Random Number</i>	46
Gambar 4.18	Hasil Respon Tangki 2 dengan Pembebanan <i>Random Number</i>	47

DAFTAR TABEL

Tabel 2.1 Paramater Pemodelan <i>Plant</i>	8
Tabel 3.1 Paramater Pemodelan <i>Plant</i> Nonlinier	22

Halaman ini sengaja dikosongkan

BAB 1

PENDAHULUAN

Pada Bab ini berisikan penjelasan awal mengenai penelitian yang akan dilakukan. Penjelasan tersebut meliputi latar belakang masalah, permasalahan yang diangkat, batasan masalah yang digunakan, tujuan dari penelitian ini, sistematika penelitian dan penyusunan laporan dan juga relevansi.

1.1 Latar Belakang

Seiring berkembangnya teknologi instrumentasi pada industri, serta tuntutan terhadap keberhasilan sistem melakukan proses kerjanya agar produksi tetap tercapai perlu dilakukan pengendalian, yang tujuannya adalah agar sistem dapat menjaga kestabilannya dalam suatu proses yang sedang berlangsung secara efektif dan efisien. Ada banyak pengendalian yang harus dikendalikan dalam suatu proses di industri, diantaranya *pressure, flow, temperature, concentration* dan *level*. Pengendalian yang dilakukan diharapkan dapat mempertahankan *life time unit*, memudahkan kerja manusia, mengurangi biaya perawatan, mendapatkan kualitas produksi yang baik, keselamatan dan keamanan lingkungan proses.

Pada industri proses seperti industri *petro-chemical*, pembuatan kertas, dan industri pengolahan air yang melibatkan cairan, suatu cairan akan dipompa dan dialirkan dari satu tangki ke tangki yang lain untuk diolah. Ada beberapa jenis tangki yang banyak digunakan pada industri proses. Ada jenis tangki yang tidak terhubung dengan tangki yang lain (*single tank*) dan ada yang tersusun dari beberapa tangki yang saling berhubungan (*coupled tanks*). [1]

Coupled tank digolongkan pada *plant* dengan sistem *Multi Input Multi Output* (MIMO) atau proses *Multivariable*. Pengaturan pada *plant* MIMO lebih sulit dibanding *plant* dengan sistem *Single Input Single Output* (SISO). Karena adanya interaksi silang antara variabel masukan dan keluaran, sehingga metode kontrol yang diterapkan pada sistem SISO bisa tidak dapat digunakan secara baik pada sistem MIMO. [2]

Pada pengaturan *level* air di *coupled tanks*, *level* pada tangki kedua akan dipengaruhi oleh aliran cairan dari tangki pertama. Pemindahan cairan dari satu tangki ke tangki yang lain menyebabkan berubahnya *level* cairan dalam tangki. Dalam pengaturan *level*, pemindahan cairan biasa disebut sebagai pembebanan pada *level*. Perubahan beban ini dapat mempengaruhi dari kinerja kontroler. Kontroler yang umum digunakan

pada industri adalah kontroler PI karena kesederhanaan strukturnya dan keandalanya [3]. Pada setiap perangkat kontrol di industri seperti *programmable logic* kontroler (PLC) tersedia kontroler PID dengan tambahan fitur *autotuning* untuk memudahkan operator menjalankan sistem kontrol umpan balik dengan PID.

Tuning parameter kontroler PI dilakukan dengan menyesuaikan nilai *proportional gain* (K_p), *integral time* (T_i), agar tercapai spesifikasi yang diinginkan. Pada prakteknya di dunia industri *tuning* parameter PI sering dilakukan dengan cara trial and error.[3]

Prosedur ini akan memakan banyak waktu dan membutuhkan seorang operator yang handal. Akan tetapi apabila plant sering mengalami perubahan beban maka kontroler PI perlu dilakukan *tuning* parameter ulang agar tetap memenuhi spesifikasi kontrol yang diharapkan. Sehingga perlu diterapkan metode *autotuning* parameter kontroler PI yang dapat meminimumkan akan adanya *error* bila *tuning* dilakukan dengan prosedur trial and error dan tetap menjaga spesifikasi kontrol yang diharapkan.

Metode *autotuning* yang akan digunakan adalah penggunaan algoritma cerdas *Neural Network*. Algoritma *Neural Network* ini digunakan untuk mempelajari perubahan beban pada plant dan akan melakukan *tuning* parameter kontroler PI secara otomatis selama proses berlangsung sehingga dapat memenuhi spesifikasi kontrol yang diinginkan.

1.2 Perumusan Masalah

Plant Coupled Tank menggunakan sistem *Multi Input Multi Output* (MIMO). Sehingga variabel yang diatur pada *Coupled tank* mengalami kesulitan mengikuti sinyal referensi berupa kondisi titik yang diinginkan. Untuk mengatasi masalah tersebut, dirancang metode dekopling untuk mengatasi konfigurasi plant MIMO serta kontroler *Neural Network PI* untuk mengatur *level* pada tangki 1 dan tangki 2.

1.3 Batasan Masalah

Pada tugas akhir ini yang akan dibahas adalah mengenai pengaturan *level* dengan sistem MIMO. Implementasi kontroler pada simulasi pada *software Matlab*.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk merancang desain pengaturan *level* pada *Coupled Tank Process* dan mengimplementasikan kontroler *Neural Network - PI* pada *Coupled Tank Process*.

1.5 Sistematika Penulisan

Penulisan Tugas Akhir ini disusun dalam lima bab yang masing-masing membahas permasalahan yang berhubungan dengan Tugas Akhir yang telah dibuat dengan sistematika penulisan sebagai berikut:

- BAB 1 : PENDAHULUAN
Berisi tentang latar belakang, permasalahan, pembatasan masalah, tujuan, metodologi, sistematika penulisan dan relevansi pembahasan Tugas Akhir ini.
- BAB 2 : TINJAUAN PUSTAKA
Menjelaskan tentang teori-teori dasar yang melandasi pembuatan Tugas Akhir ini yang meliputi: gambaran umum mengenai pemodelan sistem, sistem pengaturan proses, *Coupled Tank Process*, dan *Neural Network*.
- BAB 3 : PERANCANGAN SISTEM
Dijelaskan secara detail perancangan sistem yang dibuat, pemodelan *plant*, yaitu *Coupled Tank Process* dan perancangan kontroler *Neural Network - PI*
- BAB 4 : SIMULASI DAN ANALISA
Berisi data hasil simulasi disertai dengan analisis mengenai hasil yang didapatkan dari pengujian sistem yang dilakukan.
- BAB 5 : KESIMPULAN DAN SARAN
Berisi tentang kesimpulan dan saran-saran untuk pengembangan Tugas Akhir ini.

1.6 Relevansi

Hasil yang diharapkan dari penelitian tugas akhir ini adalah didapatkan desain sistem pengendalian *level* pada *coupled-tank Process* menggunakan kontroler *neural network PI* yang dapat bekerja memprediksi perilaku sistem pada masa depan yang bergantung pada informasi sistem saat ini diharapkan dengan menggunakan kontroler neural network PI, *coupled tank* dapat mengoptimalkan waktu pengerjaan pada sistem dan lebih cepat mengkompensasi gangguan terukur.

Halaman ini sengaja dikosongkan

BAB 2 TINJAUAN PUSTAKA

Pada Bab ini, berisi tentang teori penunjang dari berbagai pustaka atau literatur seperti buku dan *paper* yang mendukung penelitian ini. Teori penunjang pada Bab ini meliputi gambaran umum mengenai pemodelan sistem, sistem pengaturan proses, *Coupled Tank Process*, dan kontroler *Neural Network - PI*.

2.1 *Coupled Tank Plant*

Coupled Tank tersusun dari dua buah tangki yang dihubungkan oleh sebuah pipa atau lubang saluran. *Level* cairan pada tangki pertama ditunjukkan sebagai H_1 dan H_2 adalah *level* tangki kedua. Bila *input* kontrol adalah *flow* Q_1 , maka variabel yang akan dikontrol adalah kedua *Level* H_1 dan H_2 , dengan *disturbance* yang disebabkan oleh variasi *flow* pada *valve* A, *valve* B. Disini akan timbul suatu sistem dengan dua tangki saling berinteraksi.[1]

Sistem *Coupled Tank* dapat dikonfigurasi sebagai sistem SISO atau sebagai Sistem MIMO melalui manipulasi *input* dan *sectional* daerah *valve*.

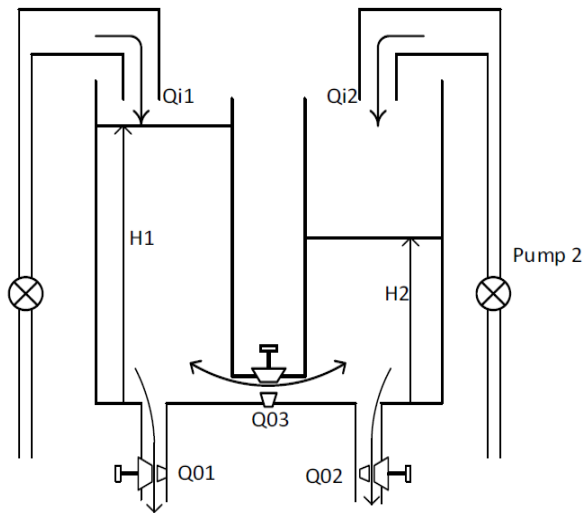
Sistem pengaturan merupakan otak dari *plant level Coupled Tank*. Dengan sistem pengaturan, *plant level* dapat digunakan untuk melakukan sesuatu yang berguna. Pengaturan didapat dari komponen yang mengubah kecepatan aliran air dan volume aliran guna mengatur *level* air yang diinginkan. Komponen pengaturan biasanya diletakkan pada tempat yang berbeda-beda.

Unuk pengaturan *level*, komponen yang dipakai berupa servo *valve*, yang mengatur besar kecilnya volume air yang dikeluarkan. Semakin terbuka kondisi dari servo *valve* maka aliran air yang keluar akan semakin besar, begitu pula sebaliknya. Buka tutupnya pergerakan servo *valve* diatur oleh sinyal kontrol yang dikirimkan oleh kontroler.

Plant Level mempunyai tujuan utama untuk menghasilkan sesuatu yang berguna pada suatu proses industri. Beban memiliki pengertian komponen yang melakukan usaha dan mengkonsumsi daya. Pada dunia industri, *plant level* berguna untuk melakukan kegiatan yang berhubungan dengan ketinggian cairan, kekentalan cairan dan pencampuran cairan. Beban pada *plant level* ini terletak pada aktuator berupa servo *valve*.

Indikator merupakan elemen penting pada *plant level Coupled Tank* yang berfungsi sebagai evaluasi *plant*. Indikator ini menunjukkan nilai *level* dan aliran yang sedang terjadi pada *plant level*. Nilai ini dijadikan bahan evaluasi oleh kontroler yang selanjutnya digunakan untuk mengatur pergerakan dari servo *valve* sehingga dapat digunakan untuk mengatur *level* air.

Plant Coupled Tank merupakan tangki dengan konfigurasi dua tangki atau lebih yang saling berhubungan dengan sebuah pipa atau saluran air seperti pada Gambar 2.1. Adanya hubungan antara tangki ini membuat *level* cairan pada setiap tangki saling berinteraksi atau berhubungan. Konfigurasi tangki seperti ini banyak digunakan pada industri proses seperti industri *petro-chemical*, pembuatan kertas, dan industri pengolahan air.[1]



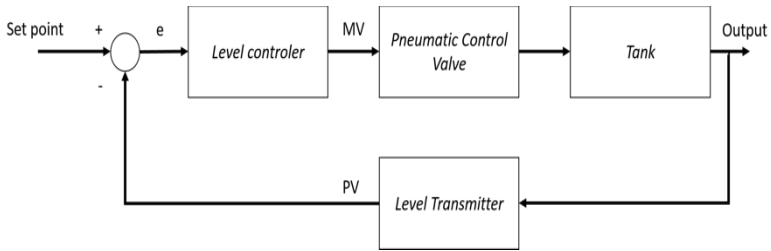
Gambar 2.1 Sistem *Coupled Tank* [1]

Level cairan pada tangki pertama ditunjukkan sebagai H_1 dan H_2 untuk *level* tangki kedua. *Flow* cairan yang masuk pada tangki dilambangkan sebagai Q_i . Untuk *flow* cairan yang keluar dari tangki ditunjukkan sebagai Q_b untuk tangki pertama, Q_c tangki kedua, dan Q_a sebagai *flow* interaksi antara tangki pertama dan tangki kedua. Sistem

Coupled Tanks dapat dikonfigurasi sebagai sistem *Single Input Single Output (SISO)*, *Multi Input Multi Output (MIMO)*, atau *Single Input Multi Output (SIMO)* berdasarkan manipulasi masukan pada pompa dan daerah kerja dari *rotary valve* yang terdapat pada *plant Coupled Tanks*. [1]

2.1.1 Karakteristik *Plant*

Konfigurasi *single loop* merupakan konfigurasi sistem kendali paling sederhana dimana hanya terdapat satu buah *loop* dengan satu kontroler untuk memenuhi objektif kontrol. Diagram blok dari sistem pengaturan dengan konfigurasi *single loop* ini sama seperti pada Gambar 2.2.

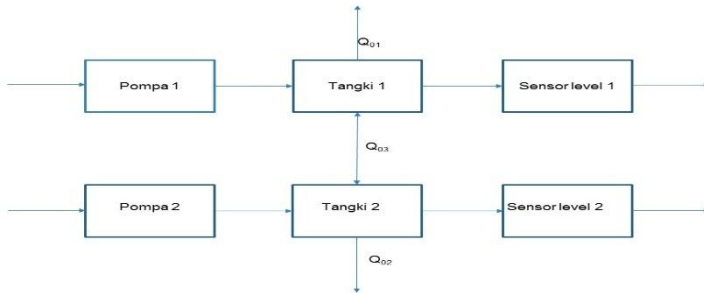


Gambar 2.2 Diagram Blok Sistem Pengaturan Proses [1]

Pengontrolan yang dipakai pada *plant* ini adalah pengontrolan *loop* tertutup dengan *feedback*. Aksi pengontrolan ini keluarannya sangat memengaruhi proses pengontrolan sistem. Keluaran yang dihasilkan sangat tergantung pada keluaran pada aksi pengontrolan sebelumnya. Keluaran yang terjadi selalu diumpan balikkan sebagai pembanding untuk keluaran selanjutnya. Apabila terjadi penyimpangan, maka kontroler akan melakukan penyesuaian sendiri untuk meminimalisir penyimpangan tersebut.

Sistem kontrol *loop* tertutup juga merupakan sistem kontrol berumpan balik. Sinyal kesalahan penggerak, yang merupakan selisih antara sinyal masukan dan sinyal umpan balik (yang dapat berupa sinyal keluaran atau suatu fungsi sinyal keluaran atau turunannya). Diumpankan ke kontroler untuk memperkecil kesalahan dan membuat agar keluaran sistem mendekati harga yang diinginkan. Dengan kata lain, istilah “*loop* tertutup” berarti menggunakan aksi umpan balik untuk memperkecil kesalahan sistem. [2]

Sistem yang akan dirancang adalah sistem pengaturan *level* dengan konfigurasi MIMO yaitu pengaturan *level* pada tangki pertama dan kedua, dengan ditunjukkan masukan berupa tegangan pada pompa air untuk memberi aliran air masuk ke tangki dan keluaran berupa *level* air pada tangki satu atau dua. Model dari *plant Coupled Tank* seperti pada Gambar 2.3.



Gambar 2.3 Diagram Blok Sistem Pengaturan Konfigurasi *Cascade*

Parameter yang digunakan pada pemodelan sistem pengaturan *level* ini dapat dilihat pada Tabel 2.1. Dalam pemodelan suatu sistem diperlukan juga untuk mengetahui karakteristik dari komponen setiap penyusun sistem. Sebagai contoh pada pompa perlu diketahui berapa tegangan maksimal kerja pompa, cara kerja dari pompa. Dengan diketahui dengan detail mengenai karakteristik dari sistem maka model matematis yang akan dibuat bisa lebih akurat dan juga dalam *real* sistem maka akan diketahui permasalahan yang timbul apabila ada karakteristik dari setiap komponen yang diabaikan.[2]

Tabel 2.1 Paramater Pemodelan *Plant*

Simbol	Keterangan	Satuan
Q_i	Debit air yang masuk ke tangki	cm^3 / s
Q_{01}, Q_{02}	Debit air yang keluar dari tangki 1 dan 2	cm^3 / s

Simbol	Keterangan	Satuan
Q_{o3}	Debit air yang keluar dari tangki 2 menuju tangki 1	cm^3 / s
H_1, H_2	Ketinggian air dalam tangki 1 dan 2	cm
A	Luas alas tangki	cm^2
a	luas pipa	cm^2

2.2 Pemodelan Sistem [2]

Sistem *Coupled Tank* terdiri dari dua tangki vertikal interkoneksi oleh saluran aliran yang menyebabkan ketinggian dari dua tangki berinteraksi. Setiap tangki memiliki pompa independen untuk masuknya cairan. Luas penampang dari bukaan katup dan dasar dari masing-masing tangki dan saluran menghubungkan dua tangki dapat divariasikan dengan *rotary valves*. Sistem *Coupled Tank* dapat dikonfigurasi sebagai SISO atau sebagai sistem MIMO melalui manipulasi pompa *input*.

Berdasarkan hukum kesetimbangan massa, persamaan dinamis setiap tangki adalah:

$$A_1 \frac{dH_1}{dt} = Q_{i1} - Q_{o1} + Q_{o3} \quad (2.1)$$

$$A_2 \frac{dH_2}{dt} = Q_{i2} - Q_{o2} - Q_{o3} \quad (2.2)$$

Dimana, H_1, H_2 masing-masing adalah tinggi dari cairan pada tangki 1 dan 2, A_1 dan A_2 adalah luas penampang tangki 1 dan tangki 2. Q_{o3} adalah debit air yang keluar dari tangki 1 menuju tangki 2. Q_{i1} dan Q_{i2} adalah debit air dari pompa masing-masing menuju tangki 1 dan tangki 2. Q_{o1} dan Q_{o2} adalah debit air keluar masing-masing dari tangki 1 dan tangki 2.

Dari Persamaan Bernoulli untuk cairan non-kental, mampat cairan dalam aliran didapatkan Persamaan (2.3) sampai Persamaan (2.5) sebagai berikut.

$$Q_{o1} = \beta_1 \cdot a \cdot \sqrt{2 \cdot g \cdot H_1} = \beta_1 \cdot a \cdot \sqrt{2 \cdot g \cdot H_1} \quad (2.3)$$

$$Q_{o1} = \beta_2 \cdot a \cdot \sqrt{2 \cdot g \cdot H_2} = \beta_2 \cdot a \cdot \sqrt{2 \cdot g \cdot H_2} \quad (2.4)$$

$$Q_{o1} = \beta_x \cdot a \cdot \sqrt{2 \cdot g \cdot |H_2 - H_1|} = \beta_x \cdot a \cdot \sqrt{2 \cdot g \cdot |H_2 - H_1|} \quad (2.5)$$

Dimana, a_1 , a_2 , dan a_3 merupakan konstanta proporsional dimana bergantung dari koefisien debit area yang saling silang dan konstan gravitasi. Dengan menggunakan nilai dari (2.3) ke (2.5) di (2.1) dan (2.2) persamaan nonlinear menggambarkan dinamika multi-*input* dan multi-*output* sistem berasal.[2]

$$A_1 \frac{dH_1}{dt} = Q_{i1} - \beta_1 \cdot a \cdot \sqrt{2 \cdot g \cdot H_1} + \dots \dots \dots \quad (2.6)$$

$$\dots \dots \dots \beta_x \cdot a \cdot \sqrt{2 \cdot g \cdot |H_2 - H_1|}$$

$$A_2 \frac{dH_2}{dt} = Q_{i2} - \beta_2 \cdot a \cdot \sqrt{2 \cdot g \cdot H_1} + \dots \dots \dots \quad (2.7)$$

$$\dots \dots \dots \beta_x \cdot a \cdot \sqrt{2 \cdot g \cdot |H_2 - H_1|}$$

Mengingat adanya perubahan yang kecil dari q_1 dan q_2 di kedua *input control* masing-masing, h_1 dan h_2 akan menjadi hasil perubahan dari *level* dua tangki. Model linierisasi didapat pada Persamaan (2.8) dan Persamaan (2.9).

$$A_1 \frac{dH_1}{dt} = q_1 - \frac{\beta_1 a \sqrt{g}}{2\sqrt{H_1}} h_1 + \frac{\beta_x a \sqrt{g}}{2\sqrt{|H_2 - H_1|}} (h_2 - h_1) \quad (2.8)$$

$$A_2 \frac{dH_2}{dt} = q_2 - \frac{\beta_2 a \sqrt{g}}{2\sqrt{H_2}} h_2 - \frac{\beta_x a \sqrt{g}}{2\sqrt{|H_2 - H_1|}} (h_2 - h_1) \quad (2.9)$$

Dinamika *plant* ini dapat dimodelkan dengan persamaan diferensial pada Persamaan (2.10) dan Persamaan (2.11)

$$\frac{dH_1(t)}{dt} = \frac{k_1}{A} U_1(t) - \frac{\beta_1 a}{A} \sqrt{\frac{g}{2h_1}} H_1(t) + \frac{\beta_x a}{A} \sqrt{\frac{g}{2|H_2 - H_1|}} [H_2(t) - H_1(t)] \quad (2.10)$$

$$\frac{dH_2(t)}{dt} = \frac{k_2}{A} U_2(t) - \frac{\beta_2 a}{A} \sqrt{\frac{g}{2h_2}} H_2(t) - \frac{\beta_x a}{A} \sqrt{\frac{g}{2\|H_2 - H_1\|}} [H_2(t) - H_1(t)] \quad (2.11)$$

2.3 Decoupling [1], [3]

Pada dunia industri, sering terjadi interaksi antara variabel *input-output*. Perubahan suatu *input* kadang tidak hanya berpengaruh pada satu *output* saja, melainkan berpengaruh juga pada *output* lain, atau yang disebut dengan *Multi Input Multi Output* (MIMO).

Plant Coupled-Tank memiliki dua *input* dan *output* yang punya pengaruh silang. Pada Gambar 2.4, *output* $h_1(t)$ tidak hanya dipengaruhi oleh $u_1(t)$ tetapi juga dipengaruhi oleh $u_2(t)$, begitu juga dengan $h_2(t)$. *Plant* pada proses semacam ini dapat disederhanakan seakan-akan seperti proses SISO dengan teknik *Decoupling*. *Decoupling* mentransformasikan model MIMO ke dalam SISO untuk memudahkan analisis dan perancangan kontroler. Hubungan suatu *input* berpengaruh pada *output loop* lainnya. Penulisan model untuk Gambar 2.4 pada Persamaan (2.12) dan Persamaan (2.13).

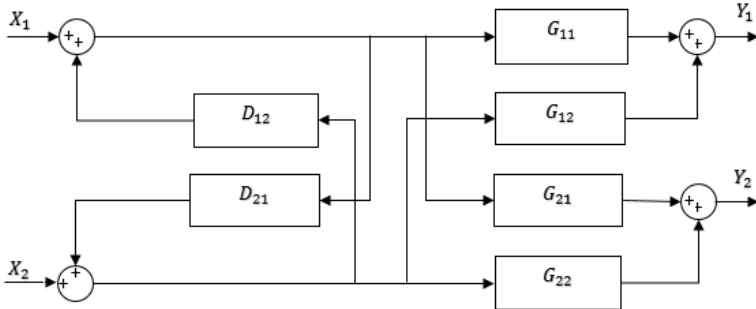
$$\text{Loop 1: } Y_1 = G_{11}X_1 + G_{12}X_2 \quad (2.12)$$

$$\text{Loop 2: } Y_2 = G_{21}X_1 + G_{22}X_2 \quad (2.13)$$

Untuk mengurangi interaksi *control loop*, dapat ditambahkan dekopler pada konfigurasi multi *loop* konvensional. Sistem kontrol dekopling memberi dua keuntungan:

- Interaksi *control loop input* lain dihilangkan sehingga stabilitas sistem *closed loop* sistem ditentukan oleh karakteristik *feedback closed loop input* itu sendiri
- Perubahan *setpoint* pada satu pengubah terkendali tidak mempengaruhi pengubah-pengubah terkendali yang lain.

Salah satu jenis kontrol dekoupling untuk proses dengan dua *input* dan dua *output* diperlihatkan pada Gambar 2.4. Tampak bahwa terdapat 2 dekopler D_{12} dan D_{21} yang berada sebelum diagram blok sistem MIMO.[3]



Gambar 2.4 Sistem Dekoupling pada Coupled Tank.[3]

Dekoupler dirancang untuk mengkompensasi interaksi proses yang tidak diinginkan. Untuk memperoleh persamaan pada tiap dekoupler maka dilakukan analisa terpisah pada tiap *input-output*, sehingga diperoleh nilai D_{12} , pada Persamaan (2.14).

$$\begin{aligned}
 Y_1 &= G_{11}X_1 + G_{12}X_2 \\
 X_1^* &= X_1 + D_{12}X_2 \\
 Y_1 &= G_{11}X_1 + G_{11}D_{12}X_2 + G_{12}X_2 \\
 Y_1 &= G_{11}X_1 + (G_{11}D_{12} + G_{12})X_2
 \end{aligned}
 \tag{2.14}$$

Agar *output* Y_1 hanya dipengaruhi nilainya dari *input* X_1 , nilai *input* X_2 harus sama dengan 0 ($X_2=0$), sehingga persamaan menjadi Persamaan (2.15).

$$\begin{aligned}
 (G_{11}D_{12} + G_{12})X_2 &= 0 \\
 G_{11}D_{12} + G_{12} &= 0 \\
 D_{12} &= -G_{12} / G_{11}
 \end{aligned}
 \tag{2.15}$$

Setelah mendapatkan nilai D_{12} dari Persamaan (2.15) maka sistem *SISO (Single Input Single Output)* untuk Y_1 bisa didapatkan pada Persamaan (2.16).

$$\begin{aligned}
 Y_1 &= G_{11}X_1 + G_{12}X_2 \\
 X_1^* &= X_1 + D_{12}X_2 \\
 Y_1 &= G_{11}X_1 + G_{11}D_{12}X_2 + G_{12}X_2 \\
 Y_1 &= G_{11}X_1 + (G_{11}D_{12} + G_{12})X_2 \\
 Y_1 &= G_{11}X_1 + (G_{11} \frac{-G_{12}}{G_{11}} + G_{12})X_2 \\
 Y_1 &= G_{11}X_1 + (-G_{12} + G_{12})X_2 \\
 Y_1 &= G_{11}X_1 + (0)X_2
 \end{aligned} \tag{2.16}$$

Dari Persamaan (2.16) dapat dilihat bahwa sistem untuk Y_1 bersifat *SISO (Single Output Single Input)*.

Untuk nilai D_{21} diperoleh dari Persamaan (2.17) sebagai berikut:

$$\begin{aligned}
 Y_2 &= G_{22}X_2^* + G_{21}X_1 \\
 X_2^* &= D_{21}X_1 + X_2 \\
 Y_2 &= G_{22}D_{21}X_1 + G_{22}X_2 + G_{21}X_1 \\
 Y_1 &= G_{22}X_2 + (G_{22}D_{21} + G_{21})X_1
 \end{aligned} \tag{2.17}$$

Agar *output* Y_2 hanya dipengaruhi nilainya dari *input* X_2 , nilai *input* X_1 harus sama dengan 0 ($X_1=0$), sehingga persamaan menjadi Persamaan (2.18).

$$\begin{aligned}
 (G_{22}D_{21} + G_{21})X_1 &= 0 \\
 G_{22}D_{21} + G_{21} &= 0 \\
 D_{21} &= -G_{21} / G_{22}
 \end{aligned} \tag{2.18}$$

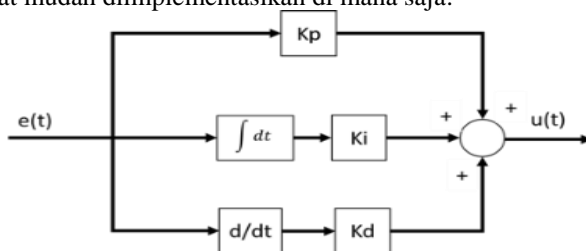
Setelah mendapatkan nilai D_{21} dari Persamaan (2.18) maka sistem *SISO (Single Input Single Output)* untuk Y_1 bisa didapatkan pada Persamaan (2.19).

$$\begin{aligned}
Y_2 &= G_{22}X_2^* + G_{21}X_1 \\
X_2^* &= X_2 + D_{21}X_1 \\
Y_2 &= G_{22}X_2 + G_{22}D_{21}X_1 + G_{22}X_2 \\
Y_2 &= G_{22}X_2 + (G_{22}D_{21} + G_{21})X_1 \\
Y_2 &= G_{22}X_2 + (G_{22} \frac{-G_{21}}{G_{22}} + G_{22})X_1 \\
Y_2 &= G_{22}X_2 + (-G_{21} + G_{22})X_1 \\
Y_2 &= G_{22}X_2 + (0)X_1
\end{aligned}
\tag{2.19}$$

Dari Persamaan (2.19) dapat dilihat bahwa sistem untuk Y_2 sudah bersifat *SISO* (*Single Output Single Input*).

2.4 Kontroler PID [4]

Kontroler PID merupakan salah satu jenis kontroler konvensional yang terbukti handal dan banyak digunakan di industri. Kesederhanaan struktur kontroler PID juga menjadi salah satu alasan mengapa kontroler ini sangat mudah diimplementasikan di mana saja.



Gambar 2.5 Blok Diagram Kontroler PID

Kontroler PID memiliki tiga komponen penyusun utama yaitu konstanta proporsional yang dilambangkan dengan K_p , konstanta waktu integral dengan τ_i , dan konstanta waktu differensial τ_d dengan seperti pada Gambar 2.5. Dalam persamaan matematis, output dari kontroler PID dapat dirumuskan pada Persamaan (2.20)

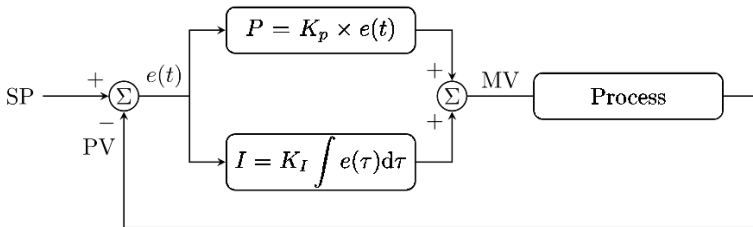
$$u(t) = K_p \left[e(t) + \frac{1}{\tau_i} \int_0^t e(t) dt + \tau_d \frac{d(e(t))}{dt} \right] \quad (2.20)$$

Ketiga konstanta tersebut yang kemudian disebut dengan parameter kontroler PID. Kelebihan dari kontroler PID salah satunya adalah menghilangkan *error steady state* yang terjadi pada kontroler proporsional.

Performa kontroler PID sangat ditentukan oleh nilai parameter PID. Untuk memenuhi spesifikasi sistem pengaturan yang diharapkan, parameter PID harus dihitung terlebih dahulu. Ada berbagai macam metode perhitungan nilai parameter PID, yaitu metode analitis, Zieger Nichols, Cohen-Coon, dan lain sebagainya.

2.4.1 Kontroler *Proportional Integral* (PI)

Kontroler PI berguna menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Kontroler PI adalah kontroler konvensional yang banyak dipakai dalam dunia industri. Blok diagram dan persamaan kontroler PI ditunjukkan pada Gambar 2.6 dan Persamaan (2.21) . [5]



Gambar 2.6 Blok Diagram Kontroler PI [5]

$$u(t) = K_p \left[e(t) + \frac{1}{\tau_i} \int_0^t e(t) dt \right] \quad (2.21)$$

Komponen kontroler PI terdiri dari dua jenis yaitu proporsional dan integratif. Komponen proporsional dan integral dapat dipakai bersamaan maupun sendiri-sendiri tergantung dari respon yang kita inginkan terhadap suatu *plant*. [5]

2.4.1.1 Kontroler Proporsional [5]

Apabila pada kontroler proporsional nilai $G(s) = K_p$ maka K_p berlaku sebagai *gain* (penguat) saja tanpa memberikan efek dinamik kepada kinerja kontroler. Penggunaan kontroler proporsional memiliki berbagai keterbatasan karena sifat kontrol yang tidak dinamik. Walaupun demikian dalam aplikasi-aplikasi dasar yang sederhana kontroler proporsional mampu memperbaiki respon transien khususnya rise time dan *settling time*. Kontroler proporsional memiliki keluaran yang sebanding dengan besarnya sinyal kesalahan (selisih antara besaran yang diinginkan dengan *output*). [5]

Ciri-ciri kontroler proporsional:

- Apabila nilai K_p kecil, kontroler proporsional hanya mampu melakukan koreksi kesalahan yang kecil, sehingga akan menghasilkan respon sistem yang lambat (menambah rise time)
- Apabila nilai K_p dinaikkan, respon sistem akan semakin cepat mencapai keadaan *steady state* (mengurangi rise time)
- Namun, apabila nilai K_p diperbesar sehingga mencapai harga yang berlebihan, akan mengakibatkan sistem bekerja tidak stabil atau respon sistem akan berosilasi
- Apabila nilai K_p dapat di-*set* sedemikian sehingga mengurangi *error steady state*, tetapi tidak menghilangkannya.

2.4.1.2 Kontroler Integral [5]

Kontroler integral berfungsi untuk menghasilkan respon sistem yang memiliki *error steady state* nol. Apabila sebuah kontroler tidak memiliki unsur integrator, kontroler proporsional tidak mampu menjamin memiliki keluaran sistem dengan *error steady state* nol.

Ketika nilai *error steady state* mendekati nol maka efek kontrol integral semakin kecil. Kontroler integral dapat menghilangkan *error steady-state*, namun pemilihan K_i yang tidak tepat dapat menyebabkan ketidakstabilan sistem. Pemilihan K_i yang sangat tinggi dapat menyebabkan *output* berosilasi karena menambah orde sistem.

Keluaran kontroler integral merupakan hasil penjumlahan yang terus menerus dari perubahan *input*. Ketika sinyal kesalahan tidak mengalami perubahan, maka kontroler integral menjaga keadaan *output* seperti sebelum terjadinya perubahan *input*. Sinyal keluaran kontroler integral merupakan luas bidang yang dibentuk oleh kurva *error*. [5]

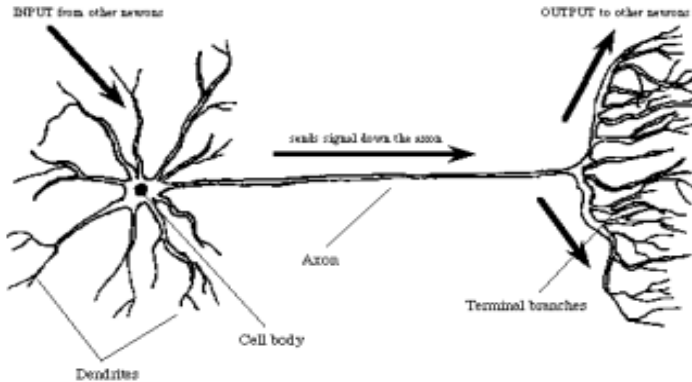
Ciri-ciri kontroler integral:

- A. Keluaran kontroler integral membutuhkan selang waktu tertentu, sehingga kontroler integral cenderung memperlambat respon
- B. Ketika sinyal kesalahan berharga nol maka keluaran kontroler bertahan pada nilai sebelumnya
- C. Apabila sinyal kesalahan tidak berharga nol, keluaran menunjukkan kenaikan atau penurunan yang dipengaruhi oleh besarnya sinyal kesalahan dan nilai K_i
- D. K_i yang berharga besar mempercepat hilangnya *offset* tetapi semakin besar nilai konstanta K_i mengakibatkan peningkatan osilasi dari sinyal keluaran kontroler.

2.5 Jaringan Syaraf Tiruan (*Artificial Neural Network*)

Artificial Neural network dimulai ketika tahun 1940-an para ilmuwan menemukan bahwa psikologi otak manusia sama dengan mode pemrosesan yang dilakukan oleh komputer. Tahun 1943 McCulloch dan Pitts memperkenalkan model *neural network* untuk pertama kalinya. Pada tahun 1954 Farley dan Clark yang pertama kali berhasil menjalankan simulasi komputer *Neural Network*, kemudian pada tahun 1958 Frank Rosenblatt mengembangkan konsep dasar tentang *perceptrons* untuk klasifikasi pola. *Artificial Neural Network* bekerja menurut prinsip kerja jaringan syaraf manusia. Dari Gambar 2.7 dapat dilihat, bahwa sebuah neuron mendapatkan *input* dari dendrit dan memberikan *output* melalui akson. Informasi masuk kedalam syaraf melalui dendrit. Dendrit ini bertugas mengumpulkan informasi dan meneruskannya ke badan sel. Pada badan sel, informasi yang telah dikumpulkan diolah dengan proses tertentu dan menghasilkan sinyal *output*. *Output* dari badan sel diteruskan ke sel syaraf berikutnya melalui akson. Akson ini akan terhubung dengan dendrit sel syaraf berikutnya. Sambungan akson sel syaraf satu dengan dendrit sel syaraf lainnya disebut sinapsis. Kekuatan sambungan pada sinapsis menentukan seberapa besar informasi yang diteruskan oleh sel

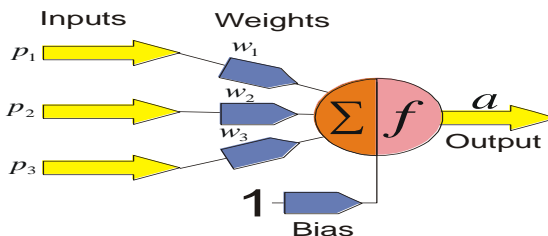
syaraf sebelumnya dapat diterima oleh sel syaraf berikutnya, yang kemudian disebut dengan bobot. [6]



Gambar 2.7 Sel Syaraf Manusia.[6]

2.5.1 Konsep Dasar Pemodelan *Neural Network* [7]

Tiruan neuron dalam struktur *Neural Network* adalah sebagai elemen proses yang dapat berfungsi seperti halnya sebuah neuron. Model dari neuron pada *Neural Network* ini seperti pada Gambar 2.8, dimana p_i merupakan *input* pada neuron dan w_i merupakan bobot atau *weight*. Sinyal input p_i ini akan dikalikan dengan nilai w_i . Nilai *output* dari neuron ini dapat dirumuskan seperti pada Persamaan (2.34).



Gambar 2.8 Model Neuron [7]

$$a = f(z) = f\left(\left(\sum_{i=1}^n p_i w_i\right) - \theta\right) \quad (2.22)$$

Dimana θ merupakan nilai bias, dan f merupakan fungsi aktivasi pada neuron. Tingkat aktivasi dari neuron dapat berharga diskrit yaitu 0 dan 1, atau kontinu yaitu antara 0 dan 1. Hal tersebut bergantung pada fungsi aktivasi yang digunakan. Ada berbagai macam fungsi aktivasi yang digunakan yaitu:

- A. *Threshold function*
- B. *Hard limit function*
- C. *Sigmoid function*
- D. *Linear Function*

Dari macam-macam fungsi aktivasi pada neuron yang paling sering digunakan adalah *sigmoid function* karena dianggap lebih mendekati dari kinerja sinyal pada otak manusia.

2.5.2 *Learning pada Neural Network* [7]

Learning pada ANN dilakukan dengan merubah bobot agar mencapai target yang diinginkan. Ada tiga metode dalam *learning*, yaitu.

- A. *Supervised learning*
Supervised learning adalah metode pembelajaran yang diarahkan untuk mempelajari contoh-contoh yang telah disediakan.
- B. *Unsupervised learning*
Unsupervised learning mempelajari pola informasi dari *input* namun tidak ada *feedback* yang secara tegas menjelaskan target yang akan dicapai.
- C. *Reinforced learning*
Reinforced learning mempelajari pola *input* dan *output* berdasarkan konsep *reward and punishment*. Jika *output* bergerak menuju target yang diinginkan, maka diberikan *reward*. Namun jika *output* bergerak menjauhi target yang diinginkan, maka diberikan *punishment*.

Halaman ini sengaja dikosongkan

BAB 3 PERANCANGAN SISTEM

Pada bab ini dibahas mengenai perancangan sistem pengaturan *level* fluida pada simulator *plant coupled tanks*. Dimulai dari pemodelan *plant coupled tank*, perancangan kontroler PI *Neural Network*. Kemudian simulasi menggunakan Software *MATLAB*.

3.1 Pemodelan Sistem

Pemodelan sistem diperlukan untuk mendapatkan model matematika dari sistem. Pada bagian ini akan dibahas bagaimana mendapatkan model dinamis. Pada model *coupled tank* kali ini untuk mendapatkan model matematis pada *plant* menggunakan permodelan fisika dari *plant* yang berupa model nonlinear setelah mendapatkan model matematis dari *plant* yang berupa model nonlinier kemudian model tersebut disimulasikan dalam *software MATLAB* dalam bentuk *close loop* untuk mendapatkan *transfer function* dalam bentuk *laplace function* dengan cara metode pendekatan respon *steady state* dari simulasi *close loop* model nonlinear.

3.2 Model Matematika Coupled Tank

Model matematis dari sistem dengan memanfaatkan hukum fisika maka didapat

$$A_1 \frac{dH_1}{dt} = Q_{i1} - Q_{o1} + Q_{o3} \quad (3.1)$$

$$A_2 \frac{dH_2}{dt} = Q_{i2} - Q_{o2} - Q_{o3} \quad (3.2)$$

$$Q_{o1} = \beta_1 \cdot a \cdot \sqrt{2 \cdot g \cdot H_1} = \beta_1 \cdot a \cdot \sqrt{2 \cdot g \cdot H_1} \quad (3.3)$$

$$Q_{o2} = \beta_2 \cdot a \cdot \sqrt{2 \cdot g \cdot H_2} = \beta_2 \cdot a \cdot \sqrt{2 \cdot g \cdot H_2} \quad (3.4)$$

$$Q_{o3} = \beta x \cdot a \cdot \sqrt{2 \cdot g \cdot \sqrt{|H_1 - H_2|}} = \beta x \cdot a \cdot \sqrt{2 \cdot g \cdot |H_1 - H_2|} \quad (3.5)$$

Dengan menggunakan nilai dari Persamaan (3.3) sampai (3.5) ke dalam Persamaan (3.1) dan (3.2) maka diperoleh Persamaan *nonlinear* yang menggambarkan dinamika *multi-input* dan *multi-output*. Maka bisa

didapatkan permodelan nonlinier dari *plant* yang terdapat pada Persamaan (3.6) dan (3.7) .

$$A_1 \frac{dH_1}{dt} = Q_{i1} - \beta_1 \cdot a \sqrt{2 \cdot g \cdot H_1} + \dots \dots \dots \quad (3.6)$$

$$\dots \dots \dots \cdot \beta_x \cdot a \sqrt{2 \cdot g \cdot |H_1 - H_2|}$$

$$A_2 \frac{dH_2}{dt} = Q_{i2} - \beta_2 \cdot a \sqrt{2 \cdot g \cdot H_1} + \dots \dots \dots \quad (3.7)$$

$$\dots \dots \dots \cdot \beta_x \cdot a \sqrt{2 \cdot g \cdot |H_1 - H_2|}$$

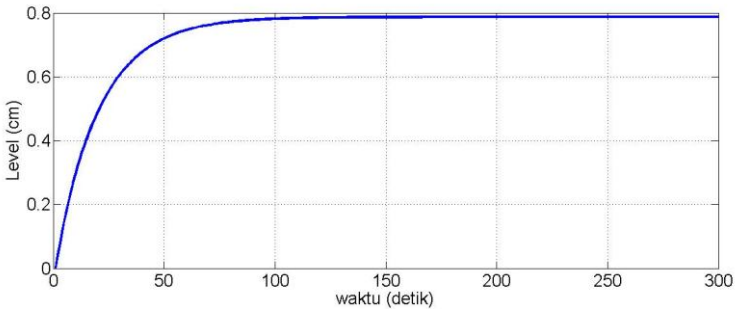
Dimana:

Tabel 3.1 Paramater Pemodelan *Plant* Nonlinier

Simbol	Keterangan	Satuan	nilai
A	Luas penampang	cm ²	66,25
A	Luas penampang lubang keluaran tangki 1 dan 2 dan saluran penghubung antara tangki 1 dan 2	cm ²	0,1963
$\beta_1,$ β_2	Rasio bukaan katup (<i>valve</i>) pada lubang keluaran tangki 1 dan 2		0,35903
β_x	Rasio <i>valve</i> antara tangki 1 dan 2		0,38705
G	Gravitasi	m ² /s	9,8
Q_{i1}	Debit air yang masuk pada tangki 1	Cm ³ /s	2,88
Q_{i1}	Debit air yang masuk pada tangki 2	Cm ³ /s	2,588

Setelah memasukkan semua parameter pada model nonlinier kemudian disimulasikan pada *software matlab* untuk mendapatkan model

linier dalam bentuk transfer function yang berupa transformasi *laplace* pada *plant model coupled tank* dengan cara pendekatan model respon *steady state* dimana respon *close loop* dari model nonlinier diberi masukan *step* 1 untuk tangki 1 dan tangki 2 untuk mendapatkan model linier dari plant dalam bentuk transformasi *laplace*.

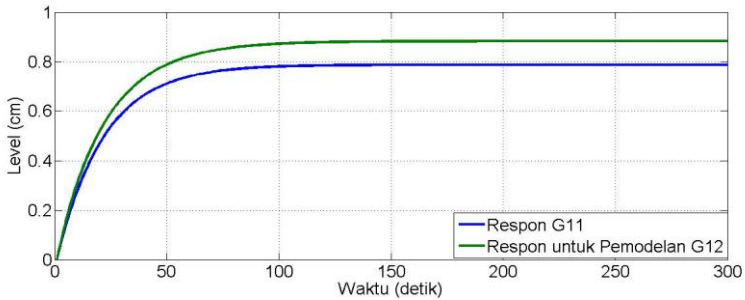


Gambar 3.1 Respon Nonlinier untuk Pemodelan Linier G_{11}

Dengan hasil *close loop* pada model nonlinier pada Gambar 3.2 dengan memberi masukan berupa *step* pada tangki 1 dengan nilai 1 dan 0 pada tangki 2 maka didapatkan nilai G_{11} kemudian melalui hasil dari respon kita *tuning* untuk mendapatkan nilai K dan τ . Untuk nilai τ didapatkan dengan melihat waktu respon sudah memasuki nilai 0,632 respon *steady state*:

$$G_{11} = \frac{0,7874}{21s + 1} \quad (3.8)$$

Kemudian untuk mendapatkan nilai G_{12} melalui proses yang sama, yaitu simulasi *close loop* dari model nonlinier *plant* dengan diberi masukan *step* untuk tangki 1 dan tangki 2 dengan *unit step* 1 sehingga didapatkan model linier dari *plant* untuk G_{12}

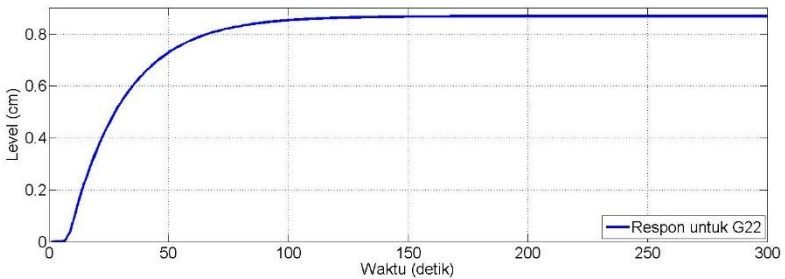


Gambar 3.2 Respon Nonlinier untuk Pemodelan Linier G_{12}

Dengan hasil *close loop* pada model nonlinier pada Gambar 3.2 maka didapatkan nilai G_{12} melalui hasil dari respon pada Gambar 3.2 yang kita *tuning* untuk mendapatkan nilai K dan τ kemudian untuk mendapatkan nilai K yang sebesar 0,0946 didapat dari menghitung selisih hasil dari pemodelan G_{11} yaitu 0,7874 dengan hasil respon *close loop* tangki 1 maka didapatkan nilai G_{12} dan untuk nilai τ didapatkan dengan melihat waktu respon sudah memasuki nilai 0,632 *respon steady state*.

$$G_{12} = \frac{0,0946}{31s + 1} \quad (3.9)$$

Untuk mendapatkan nilai G_{22} dilakukan proses yang sama yaitu melalui simulasi *close loop* pada model nonlinier dengan diberi masukan *unit step* 0 untuk tangki 1 dan 1 untuk tangki 2 sehingga didapatkan respon *steady state plant* untuk G_{22}

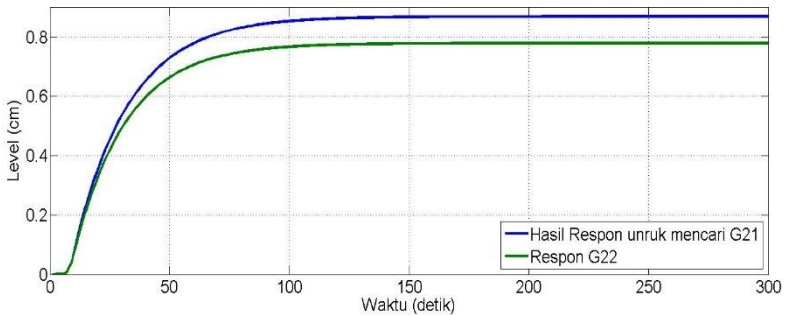


Gambar 3.3 Respon Nonlinier untuk Pemodelan Linier G_{22}

Dengan hasil *close loop* pada model nonlinier diatas pada Gambar 3.3 maka didapatkan nilai G_{22} kemudian melalui hasil dari respon kita *tuning* untuk mendapatkan nilai K dan nilai τ . Untuk nilai τ didapatkan dengan melihat waktu respon sudah memasuki nilai 0,632 respon *steady state*:

$$G_{22} = \frac{0,7829}{22s + 1} \quad (3.10)$$

Untuk mendapatkan nilai G_{21} dilakukan proses yang sama yaitu melalui simulasi *close loop* pada model nonlinier dengan diberi masukan *unit step* 1 untuk tangki 1 dan untuk tangki 2 sehingga didapatkan respon *steady state plant* untuk G_{21} :



Gambar 3.4 Respon Nonlinier untuk Pemodelan linier G_{21}

Dengan hasil *close loop* pada model nonlinier pada Gambar 3.4 maka didapatkan nilai G_{21} kemudian melalui hasil dari respon kita *tuning* untuk mendapatkan nilai K dan nilai τ maka model G_{21} bisa didapatkan :

$$G_{21} = \frac{0,08915}{29s + 1} \quad (3.11)$$

3.3 *Transfer Function Coupled Tank*

Bentuk *transfer function* hasil dari identifikasi sistem sebelumnya yaitu :

$$\begin{bmatrix} h_1(s) \\ h_2(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} \quad (3.12)$$

Dengan u_1, u_2 , tangki 1, h_2 merupakan nilai dari debit air yang masuk untuk tangki 1 dan 2 serta tinggi tangki 1 dan tangki 2 maka didapatkan nilai G_{11} sampai G_{22} pada Persamaan (3.13) sampai Persamaan (3.16) :

$$G_{11} = \frac{0,7874}{21s+1} \quad (3.13)$$

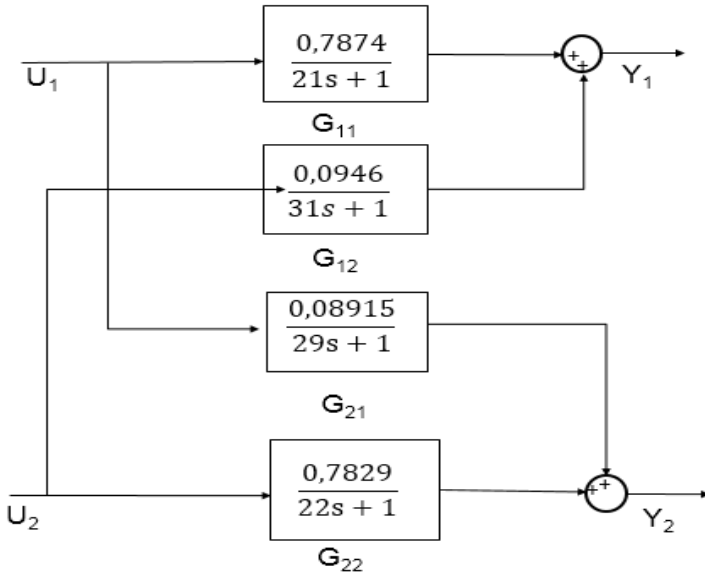
$$G_{12} = \frac{0,0946}{31s+1} \quad (3.14)$$

$$G_{21} = \frac{0,08915}{29s+1} \quad (3.15)$$

$$G_{22} = \frac{0,7829}{22s+1} \quad (3.16)$$

3.4 Desain *Decoupling*

Pada sistem *Coupled Tank Plant* ini, masing-masing input mempengaruhi kedua output. Oleh karena itu, dirancang suatu metode dekopling untuk menghilangkan pengaruh interaksi input-output. Dekopling yang akan dirancang ini, dapat meminimalisir atau menghilangkan gangguan pada *plant* akibat interaksi antara 2 *input*. Ketika salah satu *input* diberi nilai referensi yang berubah-ubah tidak mempengaruhi hasil respon yang tetap sehingga membuat *plant* menjadi lebih stabil.



Gambar 3.5 Hasil *Transfer Function* dengan Pendekatan Respon *Steady State Close Loop Model Nonlinier*

Gambar 3.5 merupakan bentuk konfigurasi MIMO (*Multi Input Multi Output*) untuk *transfer function* dari *Coupled Tank Plant*. Persamaan (3.13) sampai Persamaan (3.16) merupakan *transfer function Coupled Tank Plant* hasil identifikasi sistem.

Persamaan dekoupling didapat dari memasukkan nilai *transfer function Coupled Tank Plant* yang telah didapatkan dari pemodelan linier yang sudah dalam bentuk transformasi *laplace* yaitu pada Persamaan (3.13) sampai Persamaan (3.16) kedalam Persamaan matriks pada Persamaan (3.17) maka nilai dekoupling bisa didapatkan.

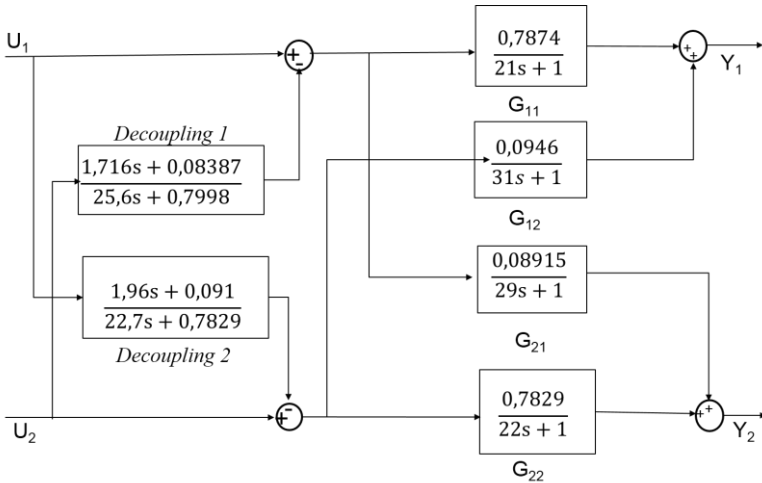
$$D(s) = \begin{bmatrix} 1 & \frac{-G_{12}(s)}{G_{11}(s)} \\ \frac{-G_{21}(s)}{G_{22}(s)} & 1 \end{bmatrix} \quad (3.17)$$

Transfer Function yang didapat dari memasukkan nilai Persamaan (3.13) sampai Persamaan (3.16) kedalam Persamaan (3.17) adalah sebagai berikut:

$$D_{12}(s) = \frac{1,716s + 0,08387}{25,6s + 0,7998} \quad (3.18)$$

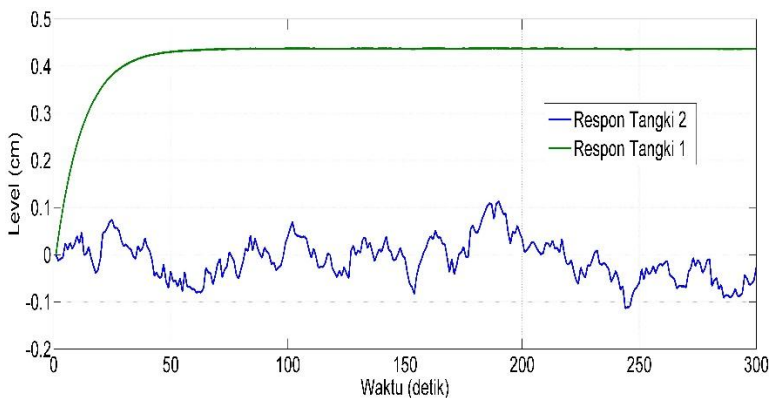
$$D_{21}(s) = \frac{1,96s + 0,0891}{22,7s + 0,7829} \quad (3.19)$$

Gambar dekopling sistem pada penelitian ini ditunjukkan pada Gambar 3.6. Dengan adanya metode dekopling pada sistem, maka *output* Y1 hanya dipengaruhi oleh *input* u1. Sedangkan *output* Y2 dipengaruhi oleh *input* u.



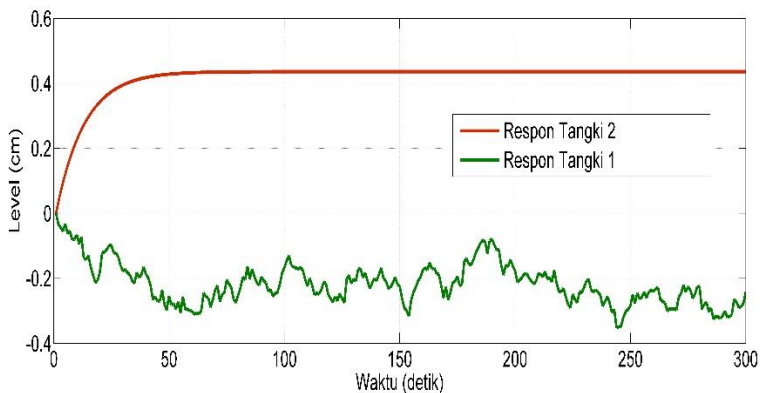
Gambar 3.6 Hasil dari *Decoupling* untuk *Plant*

Respon sistem setelah dipasang Dekopling ditunjukkan pada Gambar 3.7 dan Gambar 3.8.



Gambar 3.7 Hasil Respon setelah *Decoupling* pada Tangki 1

Setelah diperoleh nilai dekopler D_{12} dan D_{21} , maka perlu diuji apakah dekopler sudah dapat menghilangkan interaksi antar *input*. Pengujian dekopling dilakukan dengan memberi sinyal random untuk *input* u_2 dengan nilai *random* antara -0.2 sampai 0.2. Hasil respon terlihat pada Gambar 3.7 bahwa saat sistem saat *input* U_2 diberi nilai *setpoint* yang berubah ubah tidak mempengaruhi nilai *setpoint* dari U_1 hal itu ditunjukkan pada Gambar 3.7 bahwa dengan adanya dekopling dapat menghilangkan efek gangguan dari U_2 .



Gambar 3.8 Hasil Respon setelah *Decoupling* pada H_2

Dari Gambar 3.8 terlihat apabila perubahan *setpoint* diberikan pada sistem *level* pada salah *input*, maka *output* yang terpengaruh hanya pada sistem *level* yang mengalami perubahan *setpoint*, sedangkan pada sistem *pressure* tidak terpengaruh dari gangguan. Sedangkan ketika pengujian dekopling dilakukan dengan memberi sinyal random dari *input* U₁. Hasil respon pada sistem *level* tangki 2 tidak ikut terpengaruh.

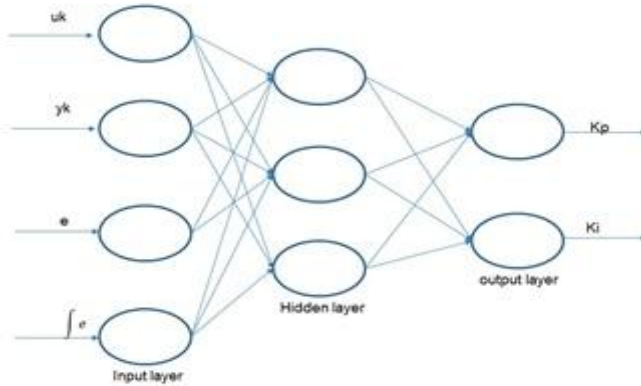
Dari Gambar 3.7 dan Gambar 3.8 bahwa dengan adanya dekopling ialah menghilangkan atau mengurangi interaksi antara tangki 1 dan tangki 2 sehingga ketika ada salah satu *setpoint* dari tangki 1 atau 2 yang nilainya berubah ubah tidak memberikan pengaruh kepada tangki 1 atau 2 yang nilai *setpoint*nya tetap.

3.5 Perancangan Kontroler PI *Neural Network*

Kontroler PI-*Neural Network* terdiri atas kontroler PI sebagai kontroler utama dan *Neural Network* sebagai algoritma cerdas yang dapat melakukan *tuning* parameter kontroler PI. Dalam mendesain kontroler PI-*Neural Network* adapun langkahnya sebagai berikut: [7]

- A. Menentukan algoritma pembelajaran yang akan digunakan pada *Neural Network*, algoritma pembelajaran yang digunakan pada perancangan kontroler ini adalah *backpropagation*.
- B. Menentukan struktur dari *Neural Network*, seperti jumlah *layer* yang digunakan dan jumlah neuron pada setiap *layer*.
- C. Melakukan inisialisasi nilai bobot yang digunakan pada *Neural Network* pada bagian antara input layer menuju hidden layer dan hidden layer menuju output layer.
- D. Menentukan nilai *learning rate* (η).

Pada kontroler PI *Neural Network* yang dirancang terdiri atas tiga *layer*, yaitu satu *input layer*, satu *hidden layer*, dan satu *output layer*. Jumlah neuron yang digunakan pada *input layer* terdiri atas empat neuron, pada *hidden layer* lima neuron, dan tiga neuron pada *output layer* yang merepresentasikan dari tiga parameter kontroler PI yaitu K_p, K_i. Struktur dari *Neural Network* yang digunakan seperti pada Gambar 3.9.



Gambar 3.9 Struktur *Neural Network* untuk *Autotuning PI*

Pada *Neural Network* ini terdapat mekanisme algoritma *feed forward* dan algoritma *backpropagation*. Pada algoritma *feed forward* dilakukan perhitungan maju dari *input layer* menuju *output layer* untuk mendapatkan nilai keluaran dari *Neural Network*. Nilai keluaran ini kemudian dibandingkan dengan nilai referensi yang digunakan.

Pada algoritma *feed forward* ini pertama dibuat persamaan untuk masukan pada *input layer* seperti pada Persamaan (3.20)

$$O_j^1 = x(j), (j = 1, 2, \dots, M) \quad (3.20)$$

M merupakan jumlah variabel masukan pada *input layer*, pada *PI-Neural Network* ini digunakan empat variabel masukan yaitu *error* yaitu selisih antara nilai referensi dan keluaran dari *plant*, Y_k yaitu keluaran dari *plant*, u_k nilai *setpoint* atau nilai referensi yang diberikan pada *plant* dan m_{ref} merupakan nilai dari model referensi yang diinginkan. Masukan pada *hidden layer* dapat dituliskan seperti pada Persamaan (3.21) dan keluaran dari *hidden layer* seperti pada Persamaan (3.22)

$$net_1^{(2)} = \sum_{j=0}^M w_{ij}^{(2)} O_j^{(1)} \quad (3.21)$$

$$O_i^2(k) = f(net_1^{(2)}(k)) \quad (i = 1, 2, \dots, Q) \quad (3.22)$$

Q merupakan jumlah neuron yang terdapat pada *hidden layer*, dan f merupakan fungsi aktivasi pada *hidden layer*, w_{ji} merupakan nilai bobot pada jaringan antara *input layer* menuju *hidden layer*. Fungsi aktivasi yang digunakan pada *hidden layer* ini adalah *piecewise linear function* seperti pada Persamaan (3.23)

$$f(x) = \text{purelin}(x) \quad (3.23)$$

Untuk masukan dan keluaran pada *output layer* dapat dituliskan dalam persamaan matematis seperti pada Persamaan (3.24) dan (3.25)

$$\text{net}_1^{(3)}(k) = \sum_{i=0}^Q w_{ij}^{(3)} O_i^{(2)}(k) \quad (3.24)$$

$$O_l^{(3)}(k) = g(\text{net}_l^{(3)}(k)) \quad (l = 1, 2, 3) \quad (3.25)$$

Dimana g merupakan fungsi aktivasi yang digunakan pada *output layer*. Karena nilai dari $O_l^{(3)}(k)$ *piecewise linear function* seperti pada Persamaan (3.22)

$$g(x) = \text{purelin}(x) \quad (3.26)$$

Keluaran dari *output layer* ini merupakan nilai parameter kontroler PI yaitu K_p dan K_i seperti pada Persamaan (3.28)

$$\begin{aligned} O_1^{(3)}(k) &= K_p \\ O_2^{(3)}(k) &= K_i \end{aligned} \quad (3.27)$$

Kemudian dilakukan mekanisme *Backpropagation* untuk perubahan nilai bobot pada *Neural Network*. Pada algoritma ini fungsi *error* atau indeks performa pada *keluaran* sistem didefinisikan seperti pada Persamaan (3.28)

$$E(k) = \frac{1}{2} (r(k) - y(k))^2 = \frac{1}{2} e^2(k) \quad (3.28)$$

Perubahan bobot pada *hidden layer* menuju *output layer* seperti pada Persamaan (3.29)

$$\Delta w_{li}^{(3)}(k) = -\eta \frac{\partial E(k)}{\partial w_{li}^{(3)}} + \alpha w_{li}^{(3)}(k-1) \quad (3.29)$$

Dimana η merupakan *learning rate*. *Learning rate* ini merupakan parameter laju pembelajaran pada algoritma *Neural Network* yang mempengaruhi kecepatan algoritma *Neural Network* mendapatkan nilai solusi minimum.

Algoritma pembelajaran untuk merubah bobot pada *output layer* dapat dituliskan seperti pada Persamaan (3.30), (3.31), dan (3.32)

$$w_{li}^{(3)}(k+1) = w_{li}^{(3)}(k) + \Delta w_{li}^{(3)}(k) \quad (3.30)$$

$$\Delta w_{li}^{(3)}(k) = \alpha \Delta w_{li}^{(3)}(k-1) \cdot \delta_l^{(3)} \cdot O_i^{(2)}(k) \quad (3.31)$$

$$\delta_l^{(3)} = e(k) \cdot \frac{\partial y(k)}{\partial u(k)} \cdot \frac{\partial u(k)}{\partial O_i^{(3)}(k)} \cdot g'(in_l^{(3)}(k)) \quad (3.32)$$

dimana turunan pertama dari $g(x)$ seperti pada Persamaan (3.32)

$$g'(x) = g(x)(1-g(x)) \quad (3.33)$$

Untuk algoritma pembelajaran perubahan bobot pada *hidden layer* seperti pada Persamaan (3.34), (3.35), dan (3.36)

$$w_{ij}^{(3)}(k+1) = w_{ij}^{(2)}(k) + \Delta w_{ij}^{(2)}(k) \quad (3.34)$$

$$\Delta w_{ij}^{(2)}(k) = \alpha \Delta w_{ij}^{(2)}(k-1) \cdot \delta_i^{(2)} \cdot O_j^{(1)}(k) \quad (3.35)$$

$$\delta_l^{(2)} = f'(in_l^{(2)}(k)) \cdot \sum_{l=1}^3 \delta_l^{(3)} w_{li}^{(3)}(k) \quad (3.36)$$

Halaman ini sengaja dikosongkan

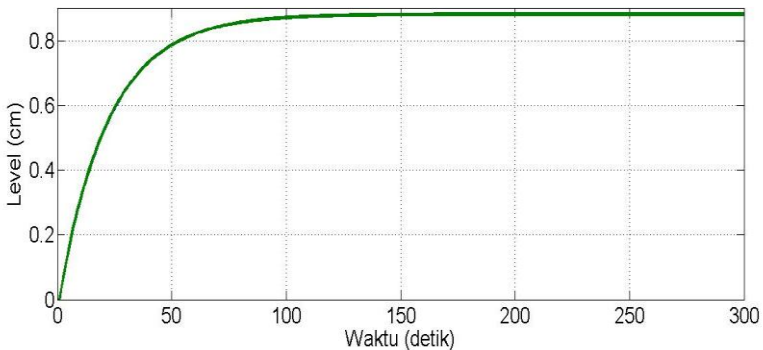
BAB 4 PENGUJIAN DAN ANALISIS

Bab ini berisi tentang hasil simulasi dan implementasi serta *analisis* dari perancangan yang sudah dilakukan pada Bab IV. Pengujian akan dilakukan dengan kontroler PI *Neural Network* untuk model linier dan model nonlinier dari *plant*. Pengujian kontroler juga dengan memberikan beban pada sistem. Kemudian dilakukan analisa terhadap data hasil simulasi yang terbagi menjadi simulasi sistem, uji beban .

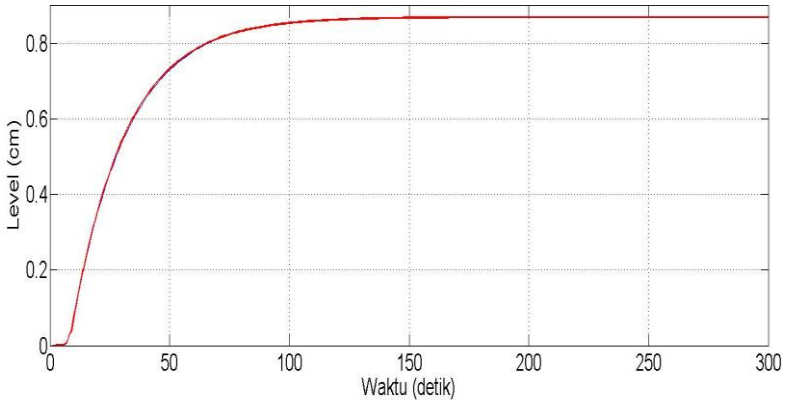
4.1 Simulasi Validasi Model linier

Simulasi model nonlinier dan linier dari sistem dilakukan untuk mengetahui karakteristik dari *Coupled Tank* dari model linear apakah sudah sesuai dengan dengan model nonlinear sehingga nilai-nilai yang diambil pada model linear hasilnya sama dengan model nonlinear. Gambar 4.1.

Pengujian ini dilakukan untuk melihat hubungan antara input yang berupa tegangan dan *output* berupa ketinggian *level* air pada tangki. Berikut merupakan respon untuk tangki 1 dan tangki 2 pada perbandingan model nonlinier dengan model linier untuk memvalidasi pemodelan secara pendekatan respon *close loop* model nonlinier.



Gambar 4.1. Respon antara Model Nonlinier dan Linier untuk Tangki 1



Gambar 4.2. Respon antara Model Nonlinier dan Linier untuk Tangki 2

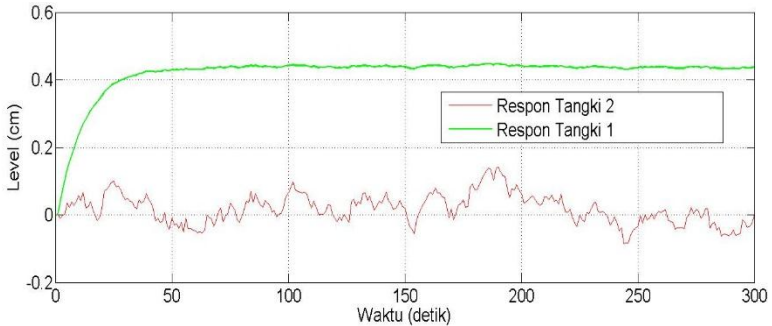
Dari hasil pada Gambar 4.1 dan Gambar 4.2 untuk simulasi kedua tangki dapat dilihat bahwa hasil permodelan linier dengan cara pendekatan respon *steady state* memiliki respon yang sama dengan model nonlinear. Hasil perhitungan *error steady state* antara model nonlinear dan model linier terdapat *error steady state* sebesar 0,001 % meskipun masih terdapat *error steady state* respon model linier sudah dapat mendekati respon *close loop* model nonlinear dari *plant*.

4.2 Simulasi Model Linier tanpa Kontroler

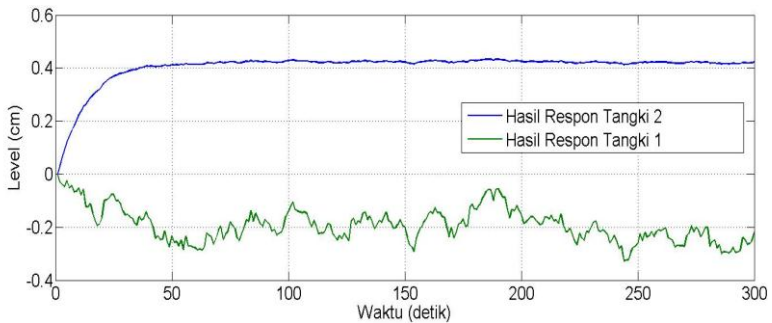
Pada kali ini akan membandingkan antara hasil simulasi model linier pada coupled tanks antara dengan adanya dekopling maupun tidak dengan menggunakan dekopling.

4.2.1 Simulasi Model Linier tanpa Dekopling

Simulasi model linier tanpa dekopling digunakan untuk menunjukkan bahwa tanpa adanya dekopling pada model *coupled tank* dapat mengakibatkan adanya interaksi antara tangki 1 dan tangki 2 apabila ada perubahan *setpoint* dikarenakan model *plant coupled tank* ini memiliki 2 nilai masukan yang saling berinteraksi untuk membuktikan adanya interaksi antara tangki 1 dan tangki 2 bisa dilihat pada Gambar 4.3 dan Gambar 4.4



Gambar 4.3 Respon Linier tanpa Dekoupling Tangki 1

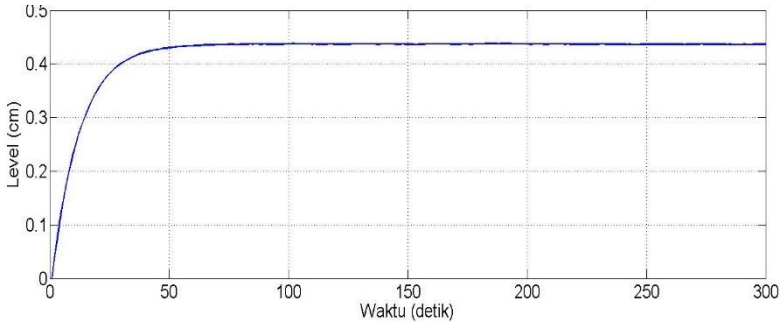


Gambar 4.4 Respon Linier tanpa Dekoupling Tangki 2

Dari hasil simulasi tanpa dekopling dapat dilihat pada Gambar 4.3 untuk *level* pada tangki 1 dan pada Gambar 4.4 untuk *level* pada tangki 2. Pada kedua gambar yaitu Gambar 4.3 dan Gambar 4.4 bisa menunjukkan bahwa masih ada interaksi antara tangki 1 dan tangki 2 ketika salah satu diberi gangguan. Pada gambar 4.3 terlihat bahwa hasil respon *level* dari tangki 1 mengalami gangguan akibat adanya *noise* dari tangki 2. Pada Gambar 4.4 juga terlihat sama dengan Gambar 4.4 bahwa *level* tangki 2 juga terganggu akibat ada *noise* dari tangki 1. Pada kedua gambar yaitu Gambar 4.3 dan Gambar 4.4 bisa dilihat bahwa respon belum memenuhi *setpoint* yaitu 4 untuk *level* tangki 1 dan 6 untuk *level* tangki 2 maka dari itu diperlukan kontroler agar kedua *level* tangki dapat memenuhi *setpoint*.

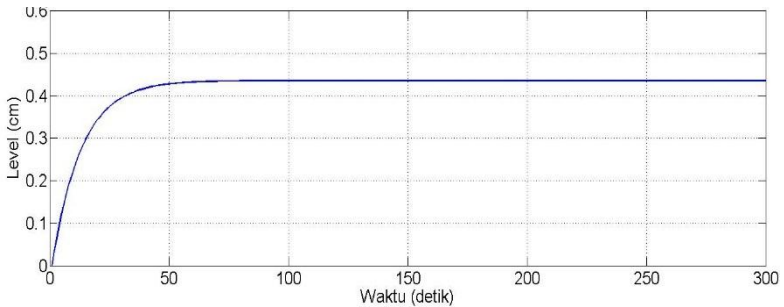
4.2.2 Simulasi Model Linier dengan Dekopling

Simulasi model linear tanpa dekopling digunakan untuk menunjukkan bahwa dengan adanya dekopling pada sistem *coupled tank* dapat menghilangkan atau meminimalisir efek *setpoint* yang berubah ubah pada salah satu *input* pada sistem *coupled tank*.



Gambar 4.5 Respon Linier dengan Dekopling Tangki 1

Dari hasil simulasi dengan dekopling pada model linier *plant* dapat dilihat pada Gambar 4.5 untuk *level* pada tangki 1. Pada Gambar 4.5 bisa menunjukkan bahwa dekopling dapat meminimalisir interaksi antara tangki 1 dan tangki 2 ketika salah satu diberi gangguan. Pada Gambar 4.3 terlihat bahwa hasil respon *level* dari tangki 1 dapat meminimalisir gangguan akibat adanya nilai *setpoint* yang berubah ubah.



Gambar 4.6 Respon Linier dengan Dekopling Tangki 2

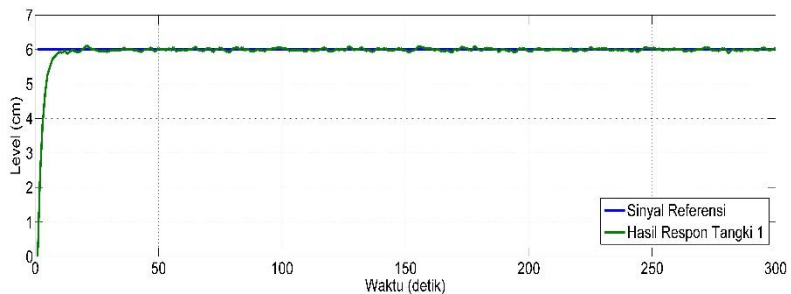
Dari hasil simulasi dengan dekopling dapat dilihat pada Gambar 4.3 untuk *level* pada tangki 1 dan pada Gambar 4.4 untuk *level* pada tangki

2. Pada kedua gambar yaitu Gambar 4.3 dan Gambar 4.4 bisa menunjukkan bahwa dekoupling dapat meminimalisir/menghilangkan interaksi antara tangki 1 dan tangki 2 ketika salah satu diberi gangguan. Pada Gambar 4.3 terlihat bahwa hasil respon *level* dari tangki 1 dapat meminimalisir gangguan akibat adanya *noise* dari tangki 2. Pada Gambar 4.4 juga terlihat sama dengan Gambar 4.3 bahwa *level* tangki 2 sudah dapat meminimalisir gangguan dari tangki 1 bisa ditarik kesimpulan bahwa dekoupling dapat meminimalisir perubahan nilai dari masukan sistem satu dengan yang lain. Pada kedua gambar yaitu Gambar 4.3 dan Gambar 4.4 bisa dilihat bahwa respon belum memenuhi *setpoint* yaitu 1 untuk *level* tangki 1 dan 1 untuk *level* tangki 2 maka dari itu diperlukan kontroler agar kedua *level* tangki dapat memenuhi *setpoint*

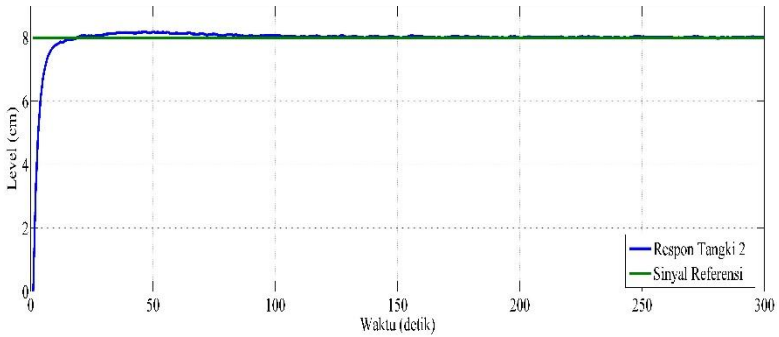
4.3 Simulasi Model Linier dengan Kontroler *Neural Network PI*

Pada simulasi kali ini kita akan membandingkan hasil simulasi dari model linier hasil dari linearisasi pendekatan respon close loop model nonlinear dengan input sinyal step. Simulasi model linier *plant* menggunakan kontroler *Neural Network PI*. Pada pengujian ini membandingkan respon model linier *plant* antara menggunakan dekoupling dan tanpa dekoupling.

4.3.1 Simulasi Model Linier dengan Kontroler tanpa Dekoupling



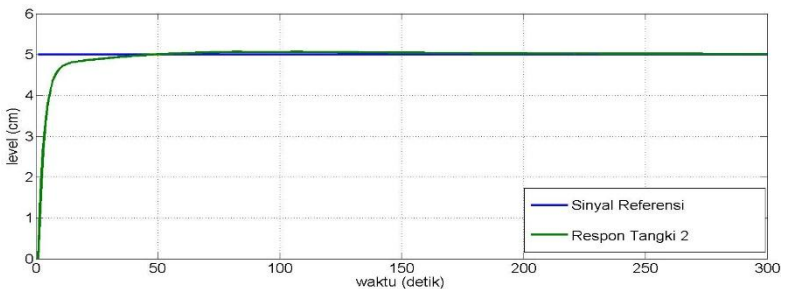
Gambar 4.7 Hasil Respon Tangki 1 dengan Perubahan *Setpoint* pada Tangki 2



Gambar 4.8 Hasil Respon Tangki 2 dengan Perubahan *Setpoint* pada Tangki 1

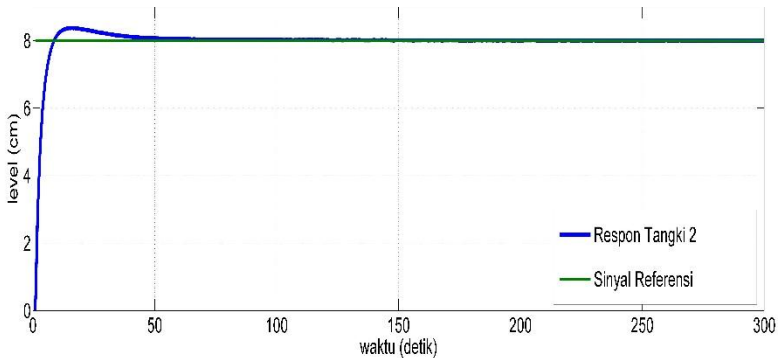
Setelah dilakukan simulasi yang ditunjukkan oleh Gambar 4.8 dan Gambar 4.7. Bahwa respon *plant* dengan model linier tanpa dekoupling masih mengalami gangguan apabila salah satu masukan diberi gangguan. Pada Gambar 4.7 menunjukkan hasil respon untuk tangki 1 mengalami *error steady state* sebesar 5,3% akibat dari sinyal dari tangki 2 yang berubah ubah, begitu juga dengan hasil dari tangki 2 akan mengalami *error steady state* sebesar 4,8% yang terlihat pada Gambar 4.8. Hasil simulasi pada Gambar 4.8 didapat nilai K_p dan K_i untuk tangki 1 ialah sebesar 13,0481 dan 0,5703, kemudian untuk tangki 1 K_p sebesar 16,308 dan K_i sebesar 0,8052.

4.3.2 Simulasi Model Linier dengan Kontroler dan Dekoupling



Gambar 4.9 Hasil Respon Tangki 1 dengan Perubahan *Setpoint* pada Tangki 2

Dari hasil simulasi pada Gambar 4.9 respon *level* pada tangki 1 dengan diberi nilai *input* untuk *step* tetap dan *input* untuk tangki 2 yang berubah - ubah sebesar 30% memberikan hasil untuk tangki 1 berupa respon tanpa memiliki *overshoot* dengan *rise time* sebesar 6,3 detik dan *settling time* sebesar 72,3 detik dan terdapat *error steady state* sebesar 0,13%, kemudian untuk nilai K_p 13,0031 dan K_i 0,6013. Hasil simulasi pada Gambar 4.9 dapat menunjukkan bahwa dekoupling dapat meminimalisir interaksi ketika ada perubahan *setpoint* dari salah satu *level* tangki.

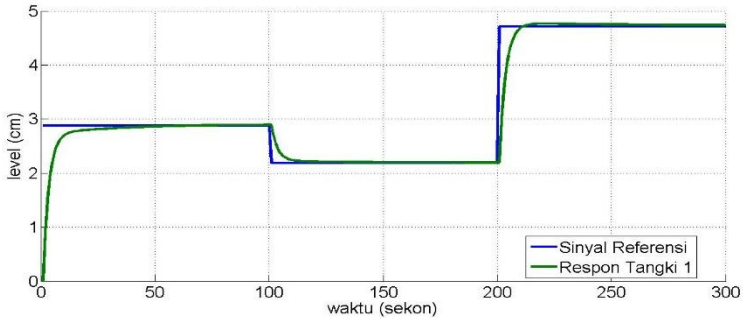


Gambar 4.10 Hasil Respon Tangki 2 dengan Perubahan *Setpoint* pada Tangki 1

Dari hasil simulasi pada Gambar 4.10 respon *level* pada tangki 2 dengan diberi nilai *input* untuk *step* tetap dan *input* untuk tangki 1 yang berubah - ubah sebesar 30% memberikan hasil untuk tangki 2 berupa respon memiliki *overshoot* sebesar 5,67% dengan *rise time* sebesar 4,9 detik dan *settling time* sebesar 30,53 detik dan terdapat tidak terdapat *error steady state*, kemudian untuk nilai K_p 13,0372 dan nilai K_i sebesar 3,1021

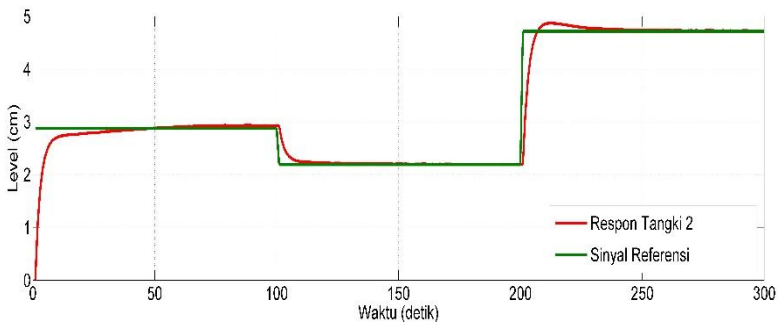
Efek dari sistem dekoupling terlihat pada simulasi model linier yang ditunjukkan oleh Gambar 4.9 dan Gambar 4.10 bahwa dekoupling dapat menghilangkan/mengurangi interaksi antara tangki 1 dan 2. Dari Gambar 4.9 dan 4.10 dapat membuktikan bahwa dengan adanya dekoupling dapat mengurangi bahkan menghilangkan interaksi antara 2 *input* pada *plant coupled tank* yang saling berinteraksi. Pada Gambar 4.9 dan Gambar 4.10 telah menunjukkan efek dekoupling pada model linier dari *plant*.

4.3.3 Simulasi Model Linier dengan Dekopling *Setpoint* yang Berubah Ubah



Gambar 4.11 Respon Tangki 1 dengan *Setpoint* yang Berubah Ubah

Dari hasil simulasi pada Gambar 4.11 respon *level* pada tangki 1 dengan *setpoint* berubah ubah memiliki *error steady state* sebesar 1,67% dengan *rise time* sebesar 6,2 detik dan *settling time* sebesar 72,3 detik dan menghasilkan nilai K_p sebesar 13,033 dan K_i sebesar 0,3196.



Gambar 4.12 Respon Tangki 2 dengan *Setpoint* yang Berubah Ubah

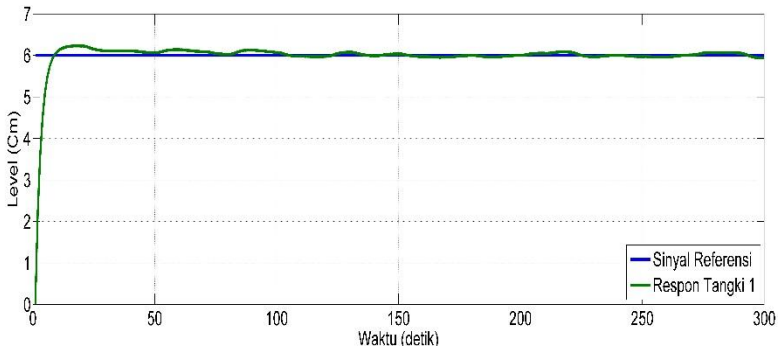
Dari hasil simulasi pada Gambar 4.12 respon *level* pada tangki 2 dengan *overshoot* sebesar 1,04% kemudian *rise time* sebesar 6,2 detik dan *settling time* sebesar 47,3 detik dan didapatkan nilai K_p sebesar 15,043 dan nilai K_i sebesar 0,8107

4.4 Simulasi Sistem *Coupled Tank Model Nonlinier Kontroler Neural Network PI*

Simulasi pengujian kontroler PI-NN pada tangki 1 dan tangki 2 dengan model *plant* nonlinier. Pada pengujian kali ini menggunakan *coupled tank* dengan pemodelan nonlinier.

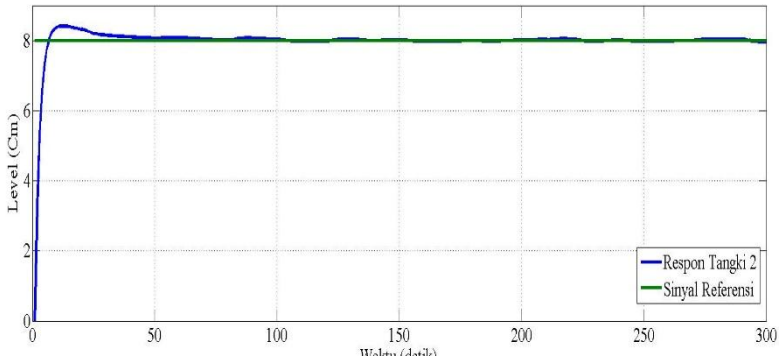
4.4.1 Simulasi Model Nonlinier tanpa Beban

Simulasi kali ini akan melakukan pengujian terhadap kontroler *neural network PI* dengan menggunakan model nonlinier tanpa memberikan beban pada *plant*.



Gambar 4.13 Respon Simulasi Nonlinier pada Tangki 1 dengan *Decoupling*

Dari hasil simulasi pada Gambar 4.13 dapat disimpulkan ketika ada gangguan dari U_2 berupa perubahan nilai *setpoint* bisa diminimalisir dan pada hasil simulasi pada model nonlinier respon *plant* sudah dapat mengikuti *setpoint* yang diberikan terlihat pada Gambar 4.13. Didapatkan nilai untuk *error steady state* sebesar 0,03 % dan nilai *rise time* 4,73 detik kemudian *settling time* 55 detik. Respon agak lama untuk mencapai nilai *steady state* dikarenakan *plant* proses membutuhkan waktu yang cukup lama untuk mencapai *steady state* kemudian didapatkan nilai K_p sebesar 14,9993 dan K_i sebesar 0,6201 .



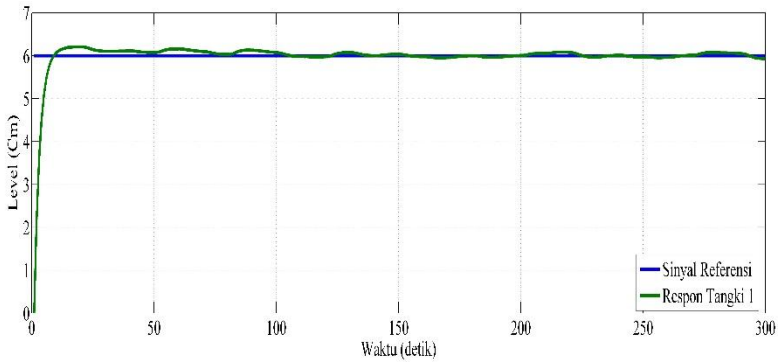
Gambar 4.14 Hasil Simulasi Nonlinier pada Tangki 2 dengan *Decoupling*

Dari hasil simulasi pada Gambar 4.14 didapatkan bahwa gangguan dari U_1 sebesar 30% dari nilai masukan sinyal U_2 bisa meminimalisir dan pada hasil simulasi pada model nonlinier respon *plant* sudah dapat menuju setpoint yang diberikan dan didapatkan nilai dengan *error steady state* sebesar 0,0315% dan nilai *rise time* 4,7 detik kemudian *settling time* 42 detik respon agak lama untuk mencapai nilai *steady state* dikarenakan *plant* proses waktu yang dibutuhkan untuk *steady state* cukup lama setelah dilakukan simulasi didapatkan nilai K_p dan K_i 10,003 dan 0,8053 untuk respon tangki 2.

Dari hasil Gambar 4.13 dan Gambar 4.14 dapat disimpulkan bahwa dengan menggunakan *plant* model nonlinier dari *coupled tank* dengan menggunakan *decoupling* linier bisa meminimalisir interaksi antara tangki 1 dan tangki 2 sehingga tidak memberikan pengaruh yang besar terhadap perubahan hasil respon yang diinginkan.

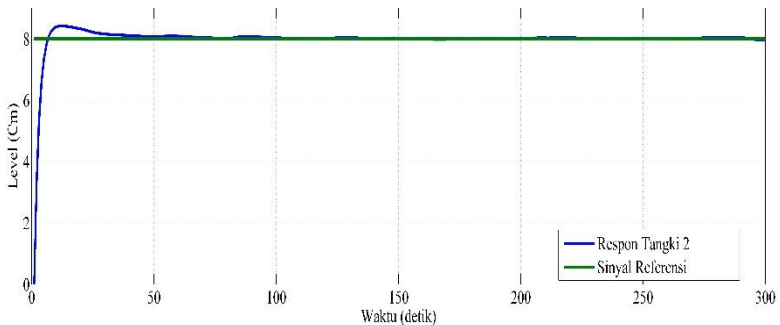
4.4.2 Simulasi Model Nonlinier dengan Beban Statis

Simulasi pengujian kontroler PI-NN pada tangki 1 dan tangki 2 dengan model *plant* nonlinier dan diberi beban statis senilai 0,08. Tujuan melakukan simulasi model nonlinier dengan pemberian beban statis ialah untuk melihat apakah kontroler dapat bekerja sesuai dengan parameter yang ada serta menuju *setpoint* yang telah ditentukan apabila pada *plant* terjadi perubahan beban dan dapat menghilangkan/meminimalisir gangguan akibat interaksi antara tangki 1 dan tangki 2. Hasil simulasi terdapat pada Gambar 4.15 dan Gambar 4.16.



Gambar 4.15 Respon Tangki 1. dengan Pembebanan Statis 0,08

Dari hasil simulasi pada Gambar 4.15 didapatkan bahwa ada perubahan *setpoint* dari U_2 bisa diminimalisir dan pada hasil simulasi pada model nonlinier respon *plant* sudah dapat mencapai nilai *setpoint* yang diinginkan dan didapatkan nilai *error steady state* yaitu 0,031 % dan nilai *rise time* 6,2 detik kemudian *settling time* 71,5 detik nilai K_p dan K_i 13,003 dan 0,827 serta terdapat *overshoot* sebesar 5,62% untuk tangki 1.



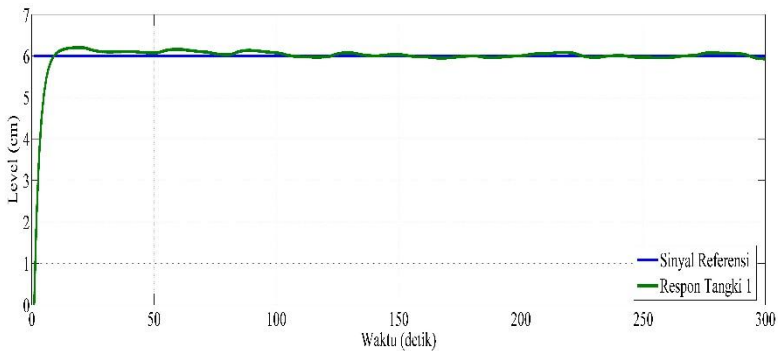
Gambar 4.16 Hasil Respon Tangki 2 dengan Pembebanan Statis 0,08

Dari hasil simulasi pada Gambar 4.16 didapatkan bahwa ada perubahan *setpoint* dari U_1 bisa diminimalisir dan pada hasil simulasi pada model nonlinier respon *plant* sudah dapat mengikuti *model reference* yang diberikan dan didapatkan nilai *error steady state* yaitu 0,013 % dan nilai *rise time* 6,07 detik kemudian *settling time* 72,5 detik nilai K_p

sebesar 13,0171 dan K_i sebesar 1,104 memiliki *overshoot* sebesar 4,75% untuk tangki 2.

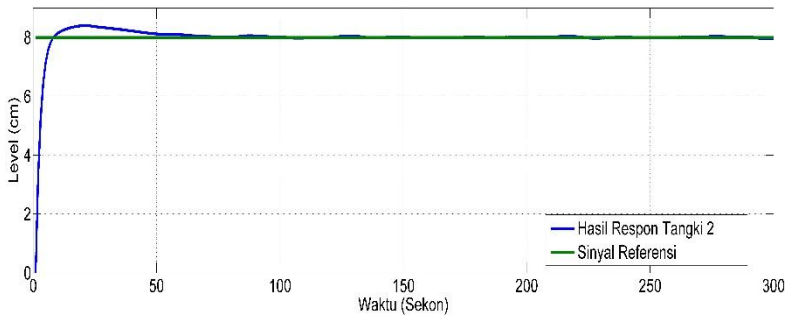
4.4.3 Simulasi Model Nonlinier dengan Beban Random

Simulasi pengujian kontroler PI-NN pada tangki 1 dan tangki 2 dengan model plant nonlinier dan diberi beban *random*. Tujuan melakukan simulasi model nonlinier dengan pemberian beban *random* ialah untuk melihat apakah kontroler dapat bekerja sesuai dengan parameter yang ada serta menuju *setpoint* yang telah ditentukan apabila pada *plant* terjadi perubahan beban setiap saat dimana nilai dari beban tersebut tidak pasti dan dapat menghilangkan/meminimalisir gangguan akibat interaksi yang terjadi antara tangki 1 dan tangki 2. Hasil simulasi terdapat pada Gambar 4.17 dan Gambar 4.18.



Gambar 4.17 Hasil Respon tangki 1, dengan Pembebanan *Random Number*

Dari hasil simulasi pada Gambar 4.17 didapatkan bahwa nilai *setpoint* dari U_2 yang berubah ubah bisa diminimalisir dan pada hasil simulasi pada model nonlinier respon *plant* pada Gambar 4.17 sudah dapat menuju *setpoint* yang ditentukan dan didapatkan nilai *error steady state* yaitu 0,035 % dan nilai *rise time* 4,966 detik kemudian *settling time* 71,5 detik nilai K_p sebesar 12,9834 dan K_i sebesar 0,527 serta terdapat *overshoot* sebesar 3,62%. Dari hasil simulasi dan analisis untuk respon tangki 1 dengan diberi beban yang berubah ubah pada nilai antara -0,02 sampai 0,02 didapatkan bahwa dengan adanya pembebanan tidak memberikan pengaruh yang besar terhadap hasil respon tangki 1..



Gambar 4.18 Hasil Respon Tangki 2 dengan Pembebanan *Random Number*

Dari hasil simulasi pada Gambar 4.18 didapatkan bahwa gangguan dari U_1 bisa sedikit diminimalisir dan pada hasil simulasi pada model nonlinier respon plant sudah dapat menuju setpoint yang ditentukan dan didapatkan nilai *error steady state* yaitu 0,033 % dan nilai *rise time* 5,973 detik kemudian *settling time* 59,5 detik nilai K_p dan K_i 13,018 dan 1,18 serta terdapat *overshoot* sebesar 7,62% untuk tangki 2.

Halaman ini sengaja dikosongkan

BAB 5 PENUTUP

5.1 Kesimpulan

- a) Dari analisis yang telah dilakukan terhadap hasil simulasi maka dapat disimpulkan bahwa pengaturan *level* pada coupled tank didapatkan model decoupling secara linear sehingga bisa mengubah perilaku sistem yang sebelumnya bersifat MIMO menjadi SISO dengan gangguan sinyal random pada masukan untuk H_2 sebesar 30% sinyal referensi.
- b) Simulasi dengan menggunakan model linier dari *plant coupled tank* didapatkan nilai untuk K_p optimal yaitu 13,0031 dan K_i optimal yaitu 0,6013 untuk *level* tangki 1 sedangkan untuk *level* tangki 2 didapatkan nilai optimal untuk K_p yaitu 15,043 dan K_i yaitu 0,8171.
- c) Simulasi *plant coupled tank* disimulasikan dengan model nonlinier dengan menggunakan *decoupling* linier efek dari decoupling dapat mengurangi gangguan dari perubahan *setpoint* salah satu *input*, meskipun masih terdapat *error steady state* dengan rata – rata sebesar 0,03%. Pada simulasi model nonlinier menghasilkan nilai K_p optimal sebesar 12,983 dan K_i sebesar 0,527 untuk *level* tangki 1, kemudian untuk tangki 2 mendapat nilai K_p optimal sebesar 13,071, K_i optimal sebesar 1,18.

5.2 Saran

Untuk penelitian selanjutnya disarankan agar membuat model *decoupling* secara nonlinier agar bisa di aplikasikan pada model *plant real* dan dapat menghasilkan nilai *decoupler* yang dapat meminimalisir efek gangguan sekecil mungkin. Pada pengerjaan tugas akhir ini menggunakan sistem *decoupling* secara linear sehingga kurang bisa mengatasi gangguan ketika diaplikasikan pada model nonlinier.

Halaman ini sengaja dikosongkan

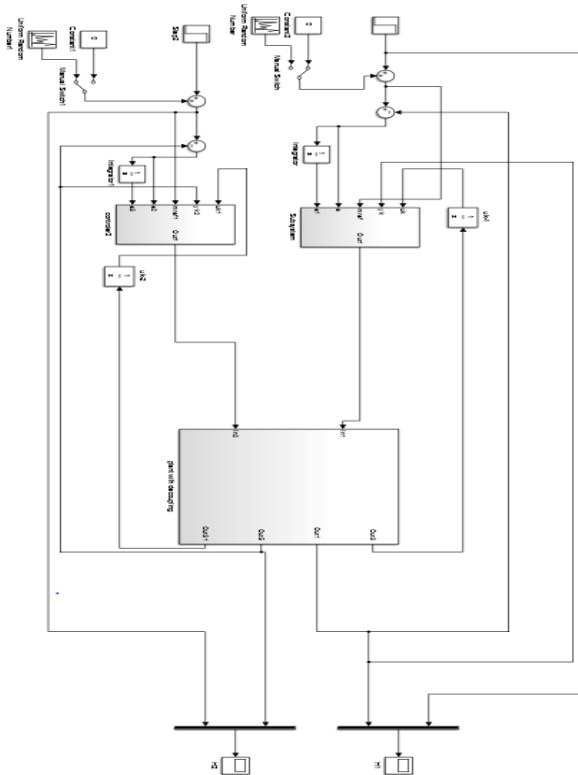
DAFTAR PUSTAKA

- [1] M.S. Ramli, M.A. Ahmad, R.M.T. Raja Ismail, "Comparison of Swarm Adaptive Neural Network Control of a Coupled Tank Liquid Level System", *International Conference on Computer Technology and Development*, Kinabalu, 13-15 November 2009.
- [2] Arjin Numsumran. Tianchai Sukri. Maitree Thumma, "Design 2 Degree Of Freedom (DOF) PI Controller With Decoupling For Coupled Tank Liquid Level System", *International Conference on Control and Automation System*, Seoul, 13-15 October 2007.
- [3] Aryani Fabiany, "Perancangan Kontroler PID-GA untuk Sistem Pengaturan Level dan Pressure pada Simulator Plant Boiler-Turbine dengan Metode Dekopling Sistem MIMO," *Tugas Akhir*, Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember, 2016.
- [4] Ogata, Katsuhiko, "*Modern Control Engineering*", Prentice Hall, New Jersey, 1970
- [5] Wicaksono, Agung Setyadi, "Perancangan dan Implementasi Sistem Pengaturan Kecepatan Motor Arus Searah Tanpa Sikat Menggunakan Kontroler PI Berbasis Neural-Fuzzy Hibrida Adaptif", *Tugas Akhir*, Jurusan Teknik Elektro ITS Surabaya, 2016.
- [6] Haykin, Simon. "*Neural Networks and Learning Machines: Third Edition*".United States of America,Pearson Prentice Hall,1999
- [7] Dharmawan, Rheza Qashmal, "Desain Autotuning Kontroler PID Berbasis Algoritma Neural Network untuk Sistem Pengaturan Cascade level dan Flow Liquid pada Plant Coupled tanks", *Tugas Akhir*, Jurusan Teknik Elektro ITS Surabaya, 2016

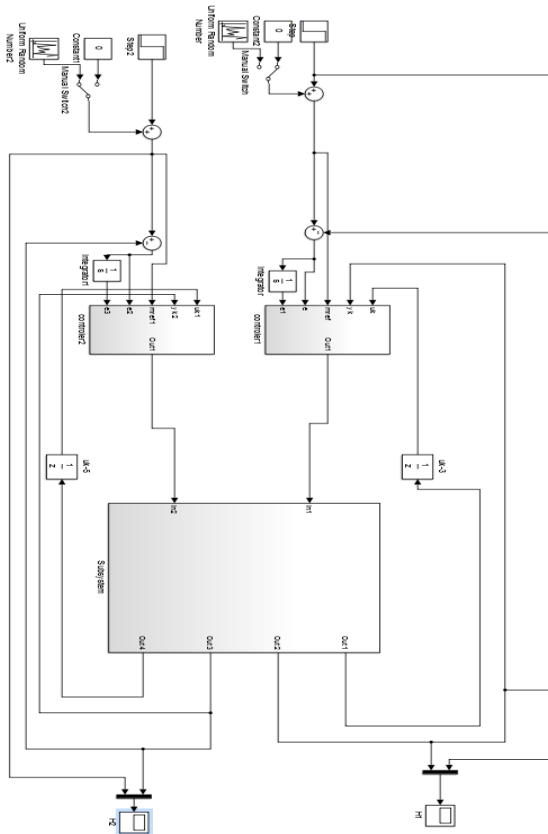
Halaman ini sengaja dikosongkan

LAMPIRAN

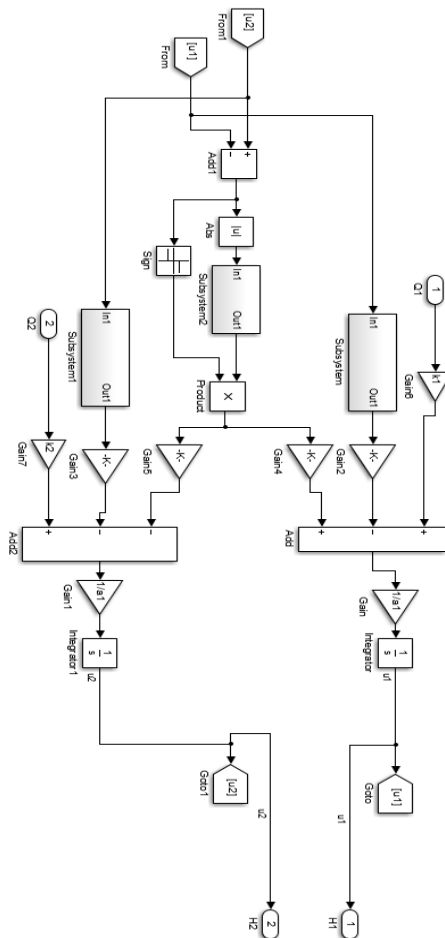
Lampiran 1. Diagram Simulink simulasi model nonlinier



Lampiran 2. Diagram Simulink simulasi model linier



Lampiran 2. Diagram Simulink simulasi pemodelan nonlinier



Halaman ini sengaja dikosongkan

RIWAYAT PENULIS



Kresna Kartika Surya Putra, 20 Januari 1994. Merupakan anak kedua dari 2 bersaudara pasangan ibu Rahayu Sulistyaningrum dan bapak Heri Suryoko. Penulis menamatkan pendidikan Sekolah Dasar di SDN Gedongan 1 Mojokerto, SMPN 1 Mojokerto, dan SMAN SBBS Sragen. Setelah menamatkan jenjang SMA, penulis melanjutkan kuliah S1 Teknik Elektro di Institut Teknologi Sepuluh Nopember melalui jalur Program ujian tulis SNMPTN. Penulis mengambil konsentrasi bidang studi Teknik Sistem Pengaturan. Pada bulan Januari, penulis mengikuti ujian Tugas Akhir sebagai syarat untuk menyelesaikan masa studi dan memperoleh gelar Sarjana Teknik Elektro. Penulis dapat dihubungi melalui *email* : kresnakartika@rocketmail.com

Halaman ini sengaja dikosongkan