

IMPLEMENTASI KONTROLER PID PADA SIMULATOR KONTROL ALIRAN

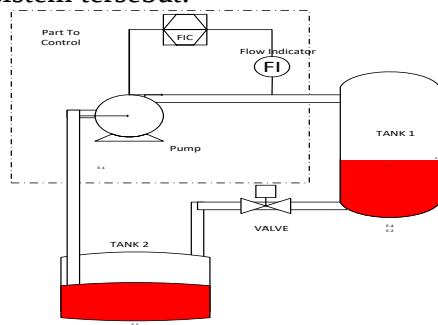
Asriyadi

Abstrak : Penelitian ini bertujuan untuk mengimplementasikan kontroler PID pada sebuah simulator kontrol aliran yang tidak terkontrol menjadi sebuah sistem yang terkontrol. Proses kontrol aliran (*flow Control*) merupakan sebuah proses yang biasanya ada pada dunia industri. Adapun tahapan penelitian dilakukan dengan lima tahapan yang pertama adalah dengan mengidentifikasi sistem, memodelkan sistem, menganalisa model sistem, merancang sebuah kontroler PID dan mengimplementasikan kontroler tersebut ke sistem agar sistem menjadi sebuah sistem yang terkontrol. Untuk mengidentifikasi sistem dan implementasi PID kontroler dilakukan dengan menggunakan HAN PID Software, sedangkan untuk memodelkan, menganalisa model dan merancang kontroler digunakan software Matlab 2012a. Hasil penelitian menunjukkan bahwa PID kontroler yang dirancang memberikan hasil yang bagus pada paramer untuk $K_p=1.25, \tau_i=0,5$ and $\tau_d=0.068$ baik dari sisi simulasi software matlab dan implementasinya pada sistem.

Kata Kunci : Kontroler PID, *flow Control*, Matlab 2012a, HAN PID Software

PENDAHULUAN

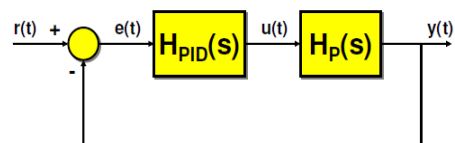
Sistem yang digunakan pada penelitian ini adalah sebuah *mini plant proses "Fluid Level controller system"*. Akan tetapi, tujuan utama pada penelitian ini adalah untuk menginvestigasi dan mengontrol *flow loop* dari sistem. Perancangan kontroler PID menggunakan metode *root locus* yang dilakukan dengan mengestimasi sistem tersebut dan kemudian melakukan tune pada kontroler pada saat proses berlangsung pada sistem. Gambar 1 berikut memperlihatkan area sistem yang akan dikontrol pada sistem tersebut.



Gambar 1. Flow Control Loop

PID Controller

Perancangan kontroler yang digunakan disini adalah sebuah metode konvensional P-, I- and D- actions. Gambar 2 memperlihatkan bentuk desain sistem kontroler PID.



Gambar 2. PID controller loop.

Berikut persamaan *transfer function* dari kontroler PID:

$$H_{PID}(s) = K_p \left(\frac{1 + \tau_i s}{\tau_i s} \right) \left(\frac{1 + \tau_d s}{1 + \frac{\tau_d}{a} s} \right) \quad (1)$$

Asriyadi adalah dosen Program Studi Teknik Listrik Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang, Jl. Perintis Kemerdekaan Km.10, Tamalanrea Makassar 90245

Dimana:

a = tamed factor (=10)

K_p = Proportional action

τ_i = Integral action

τ_d = Derivative action

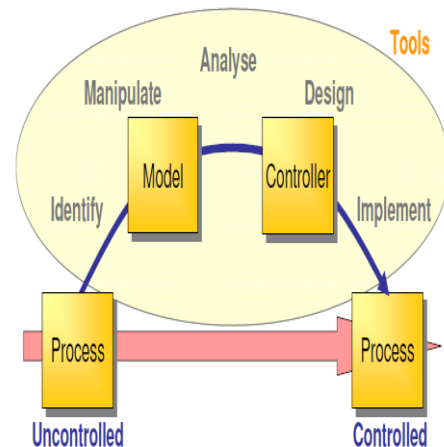
Berdasarkan persamaan 1, Kontroler PID akan dirancang dengan menggunakan *root locus tool*. Untuk menyederhanakan persamaannya maka persamaan di konversi ke representasi pole-zero:

$$H_{PID}(s) = K_p \cdot a \cdot \left(\frac{s + \frac{1}{\tau_i}}{s} \right) \left(\frac{s + \frac{1}{\tau_d}}{s + \frac{a}{\tau_d}} \right) \quad (2)$$

METODE PENELITIAN

Untuk menyelesaikan masalah tersebut, dilakukan dalam lima tahapan. Pertama, mengidentifikasi sistem, memodelkan sistem, menganalisis model sistem, merancang kontroler dan menerapkan PID kontroler ke sistem. Dalam kegiatan ini kami menggunakan perangkat lunak yaitu Program HAN PID. Perangkat lunak ini memungkinkan untuk mengontrol proses dari komputer secara manual atau otomatis untuk pengukuran, *steering signals* dan akhirnya implementasi kontroler PID. Software Matlab digunakan untuk pemodelan, analisis dan desain.

Secara umum, prosedur sistem kontrol dapat dilihat pada gambar berikut.



Gambar 3. Procedure Sistem Kontrol.

Pertama, Mengidentifikasi sistem atau proses untuk mendapatkan model dari sistem. Tahapan ini disebut dengan "*Modeling*". Ada banyak metode yang bisa digunakan dalam pemodelan sebuah proses, akan tetapi pada penelitian ini akan dilakukan beberapa metode atau prosedur dalam proses pemodelan yaitu *steady state model*, *step response*, *bode diagram*, and *noise measurements*.

Kedua, Melakukan perancangan kontroler dimana berdasarkan model yang diperoleh PID kontroler akan didesain dan akan disimulasikan dengan menggunakan *Software MATLAB 2012a*. Adapun response dari sistem kontrol akan dianalisa untuk mendapatkan tujuan dari karakteristik sistem yang diinginkan. Kemudian mengimplementasikan kontroler ke sistem dengan *HAN PID software*.

Kebutuhan Spesifikasi Sistem:

1. *Stable*
2. *Robust : Phase Margin*
(PM) $> 45^\circ$ *C and Gain Margin*
(GM) $> 6dB$
3. *Steady state error : zero*
4. *Overshoot : less than 4.3%*
5. *Settling time for 10% : as fast as possible.*

HASIL DAN PEMBAHASAN

Pada bagian ini, akan dijelaskan hasil dari lima tahapan atau prosedur yang digunakan untuk menyelesaikan permasalahan pada penelitian ini..

Identifikasi and Pemodelan Sistem.

Dinamika sebuah sistem dapat diperoleh pada pemodelan proses, dalam hal ini, sistem dapat dianalisa dan dimodelkan dengan karakteristik yang sama. Dalam hal ini digunakan *Linear Time Invariant models (LTI- models)*, hasil dari pemodelan hanya pada deviasi *linear control range*. Oleh karena itu, *point of operation* seharusnya ada pada area tersebut. Agar dapat memaksimalkan kontrol pada sistem tersebut, maka untuk *point of operation* dipilih titik tengah pada area tersebut.

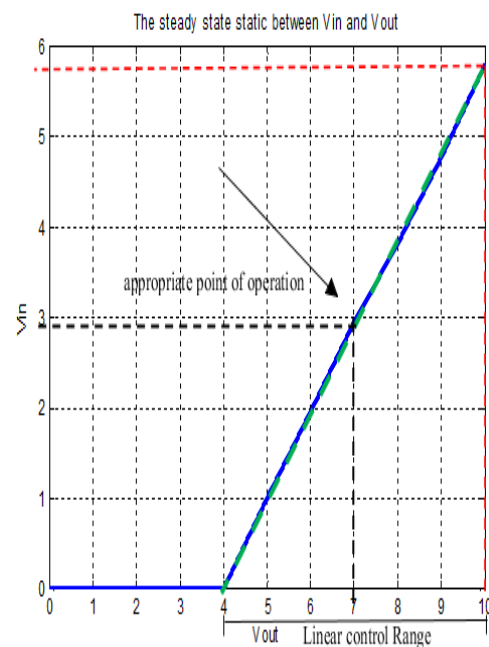
Pada dasarnya, model dari sistem seharusnya berkorenspondensi dengan proses sebenarnya. Oleh karena tu, pada aktivitas pemodelan, penting untuk didefenisikan proses dinamik dari sebuah sistem yang merupakan karesteristik utama dari sistem. Ketika karesteristik dinamik dari sebuah prose didapatkan, model kemudian akan divalidasi untuk membuktikan keakurasiannya.

Steady state model

Tujuan utama dari langkah ini adalah untuk mendapatkan *point of operation* yang valid dan *the steady state gain K_{DC}* . Pada tahapan ini, dilakukan pengukuran *steady state (static) relation* antara input (V_{in}) and output (V_{out}) dengan range dari 0 to 10 Volts pada tiap step 1 volt. Tabel 1 memperlihatkan data hasil pengukuran.

Table1. Pengukuran *Steady State*

$V_{in}(V)$	0	1	2	3	4	5	6	7	8	9	10
$V_{out}(V)$	0	0	0	0	0	1.01	1.95	2.93	3.82	4.77	5.79



Gambar 4. Hubungan *Steady state (static)* antara V_{in} and V_{out}

Dari hasil pengukuran, dapat dilihat bahwa respon dari sistem adalah sangat linear dari 4 sampai 10 dan oleh karena itu digunakan range antara 4-10 V. Untuk 10 V, sistem tersaturasi dan tidak dapat lagi menyediakan sinyal yang lebih besar.

Linear control range

Dari Gambar 4, diperoleh *linear operation* antara 4 V dan 10 V dari nilai input dimana garis hijau mengindikasikan linear range yang berkorespondensi dengan garis biru yang merupakan respon sistem, dimana V_{in} : 4 volt sampai 10 volt yang berkorespondensi dengan V_{out} : 0 sampai 5.79 Volt.

Point of operation

Berdasarkan *linear area operation*, kemudian dipilih bahwa the point of operation adalah **7 volt** dimana merupakan titik tengah dari linear range untuk mendapatkan area terbesar dari *point of operation*.

Steady state gain K_{DC}

Berdasarkan gambar 4, dapat juga diperoleh *steady state gain* K_{DC} of dari the process yang dapat diperoleh dengan persamaan berikut:

$$K_{DC} = \frac{\text{Output range}}{\text{Input Range}} \quad (3)$$

$$K_{DC} = \frac{5.79 - 0}{10 - 4} = 0.965$$

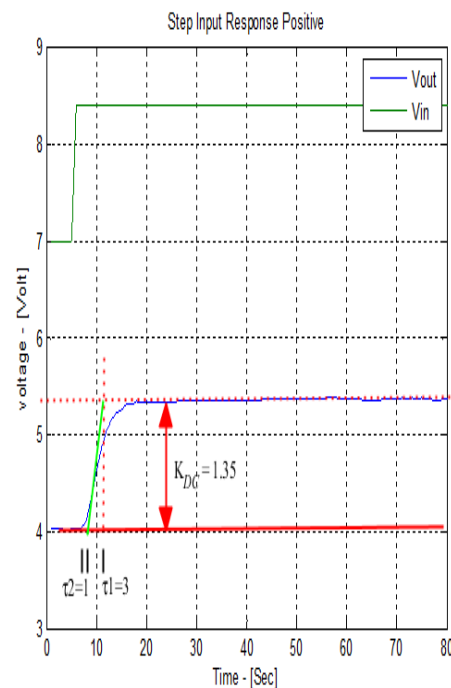
$$K[\text{dB}] = 20^{10} \log(K_{DC}) = -0.309 \text{ dB}$$

Step Response

Untuk melihat respon sistem berdasarkan *step response* maka dilakukan simulasi untuk *positive* dan *negative step response*, Lihat gambar 5 dan gambar 6.

Positive Step Test.

Step input dari **7 to 8.4** Volt diberikan pada sistem dan hasil respon diperoleh. Berdasarkan step tes data dapat direpresentasikan dalam bentuk grafik dan mendapatkan karakteristik dari respon dan digunakan untuk memodelkan *transfer function* dari sistem..



Gambar 5. Estimasi model dari *positive step response*

Berdasarkan karakteristik fisik dan respon dari step tes lihat gambar 4 dan membandingkan dengan grafik dari *Manual Lab Control*, dapat diperkirakan model sistem adalah *second order system*. Hal ini juga bersesuaian dengan kebutuhan dimana *second order* dapat diestimasi dengan baik pada $0.2 \tau_1 < \tau_2 < 0.02 \tau_1$ dimana τ_1 adalah *dominant time constant* dan *second order transfer function* bukan

merupakan *under critical damped*. Oleh karena itu, $\tau_1 = 3s$ $\tau_2 = 1s$ $K_{DC} = 1.35$

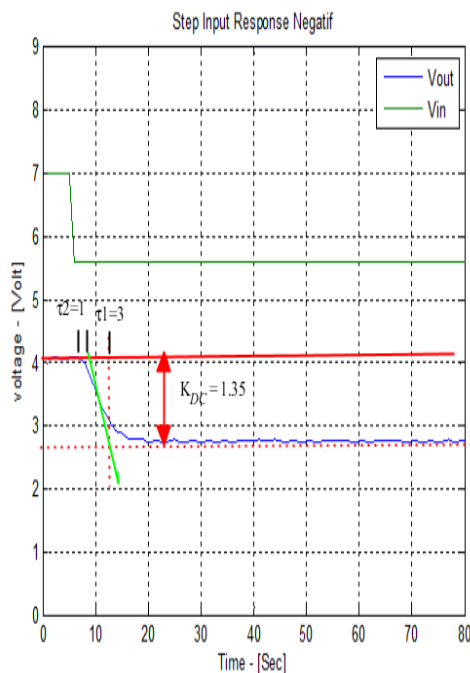
Transfer function diperoleh τ_1 , τ_2 and static gain K_{DC} dimana merupakan bentuk *second order system*.

Oleh karena itu *transfer function* $H_p(s)$ dari model adalah:

$$H_p(s) = \frac{1.35}{(1 + 3 \cdot s)(1 + s)} \quad (4)$$

Negative Step Test.

Untuk mendapatkan hasil yang lebih meyakinkan dari respon step tes, pada penelitian ini juga *negative step test input*. The *negative step* dari 7 ke 5.6 Volt diinputkan ke sistem dan hasil respon dikumpulkan. Berdasarkan *negative step test*, diperoleh karakteristik sistem:



Gambar 6. Estimasi Model *negative step response*

Sama seperti pada step positive, kami berkesimpulan bahwa sistem adalah sistem orde dua, hal ini dapat dilihat pada grafik yang diperlihatkan pada gambar 6, kemudian dimodelkan dengan pendekatan sistem orde dua. Ini juga bersesuaian bahwa sistem orde dua dapat diestimasi dengan tepat jika $0.2 \tau_1 < \tau_2 < 0.02 \tau_1$ dimana τ_1 adalah *dominant time constant and transfer function* orde dua adalah tidak *undercriticaldamped*. Oleh karena itu, $\tau_1 = 3s$ $\tau_2 = 1s$ $K_{DC} = 1.35$

Transfer function diperoleh dari from τ_1 , τ_2 and static gain K_{DC} yang membentuk *second order system transfer function*

Oleh karena itu, *transfer function* model adalah $H_p(s)$:

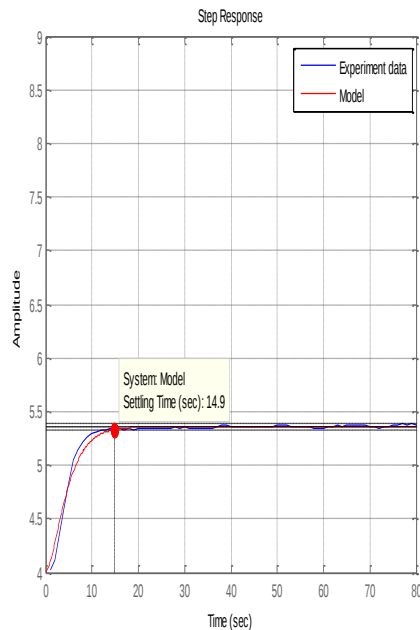
$$H_p(s) = \frac{1.35}{(1 + 3 \cdot s)(1 + s)} \quad (5)$$

Berdasarkan hasil fungsi transfer dari keduanya baik *step positive and step negative* memberikan hasil yang serupa yaitu model transfer function dari $H_p(s)$ adalah:

$$H_p(s) = \frac{1.35}{(1 + 3 \cdot s)(1 + s)} \quad (6)$$

Untuk menvalidasi model sistem, To validated the model, hasil response dari step tes dibandingkan dengan hasil pengukuran. Hasil

respon diperlihatkan pada gambar 7



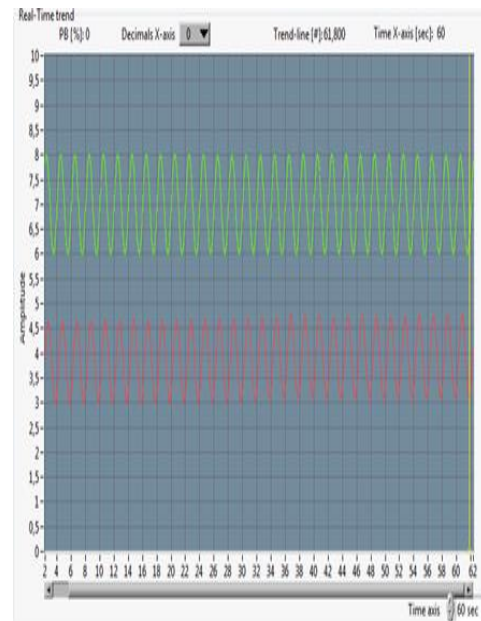
Gambar 7. Validasi Model

Dari hasil perbandingan, dapat dilihat bahwa hasil simulasi model memperlihatkan hasil yang serupa dengan hasil data pengukuran.

Bode Diagram

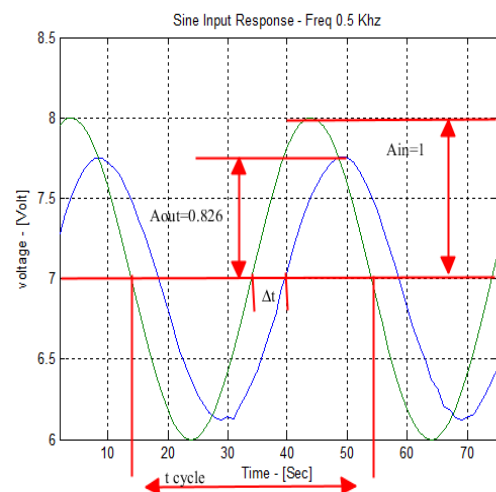
Bode plot menggambarkan respon sebuah proses dengan *input frequency*. Semakin tinggi frekuensi semakin kecil respon dari sistem. Ketergantungan ini tidaklah linear oleh karena itu perlu *diketahui break frequencies* untuk menentukan karakteristik yang pasti dari sistem. *Sinusoidal test* dilakukan dengan menggunakan *magnitude* sebesar 1 V dari point of operation, 7 V. Dalam kasus ini, semua amplitudo diset 1 V, oleh karena itu akan tetap pada *linear range* untuk semua frekuensi, dan frekuensi yang digunakan divariasikan dari 0.01 Hz

sampai 2 Hz. Setiap hasil pengukuran frekuensi kemudian digrafikkan.



Gambar 8 Pengukuran data dari sistem dengan frekuensi 0.5 KHz.

Untuk memudahkan pengidentifikasian, hasil respon kemudian diatur atau digeser pada axis yang sama, menjadi seperti pada gambar 9:



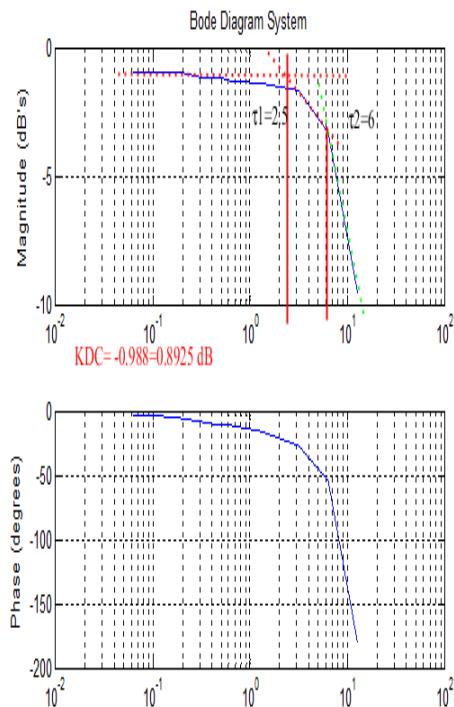
Gambar 9. Perbedaan antara sinyal input dan output.

Hasil dari *sinusoidal test identification* kemudian ditabulasikan kedalam tabel 2. Dan untuk semua grafik, dihitung *gain* dan *phasenya*.

Table 2 Hasil Pengukuran pada percobaan *Bode Diagram*

Frequency	Amplitude		Cycle Time	Shift	True Freq	Gain	Gain K	Phase
[Hz]	Ain [-]	Aout [-]	tcycle [s]	dt [s]	[Hz]	K(w)[-]	K(w) [dB]	[degrees]
0.01	1	0.8925	100	1	0.06283	0.8925	-0.9878355	-3.6
0.02	1	0.8915	50	0.5	0.12566	0.8915	-0.987573	-3.6
0.03	1	0.8915	33.333333	0.48	0.18849	0.8915	-0.987573	-5.184
0.05	1	0.8715	20	0.45	0.31415	0.8715	-1.1946522	-8.1
0.08	1	0.871	12.5	0.35	0.50264	0.871	-1.1996369	-10.08
0.1	1	0.862	10	0.3	0.6283	0.862	-1.2090547	-10.8
0.2	1	0.85	5	0.2	1.2566	0.85	-1.4116215	-14.4
0.5	1	0.826	2	0.15	3.1415	0.826	-1.6603991	-27
1	1	0.8005	1	0.15	6.283	0.8005	-3.3434374	-54
2	1	0.33	0.5	0.25	12.566	0.33	-9.8287212	-100

Untuk mendapatkan fungsi transfer dari sistem, diidentifikasi berdasarkan *bodeplot*:

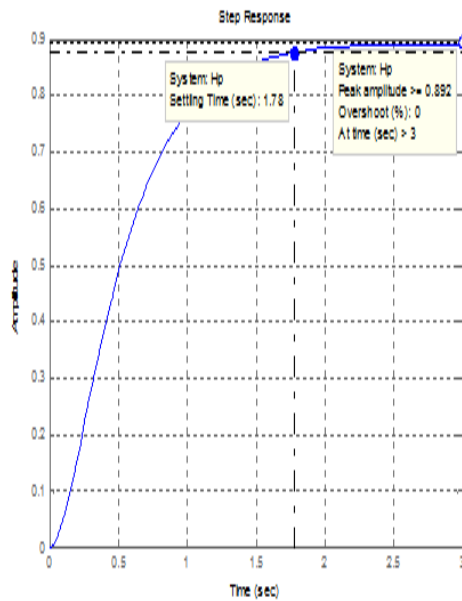


Gambar 10. *Magnitude and Phase plots of the process*

Magnitude plot (logarithmic dependency) digunakan untuk mendapatkan model dari sistem. Dari hasil *magnitude response*, dapat diperkirakan bahwa sistem adalah sistem orde dua dengan $\tau_1 = 2.5$, $\tau_2 = 6$. Dan $K_{DC} = 0.8925$. oleh karena itu, tranfer function adalah:

$$H_p(s) = \frac{0.8925}{\left(1 + \frac{s}{2.5}\right) \cdot \left(1 + \frac{s}{6}\right)} = \frac{0.8925}{(1 + 0.4 \cdot s) \cdot (1 + 0.167s)} \quad (7)$$

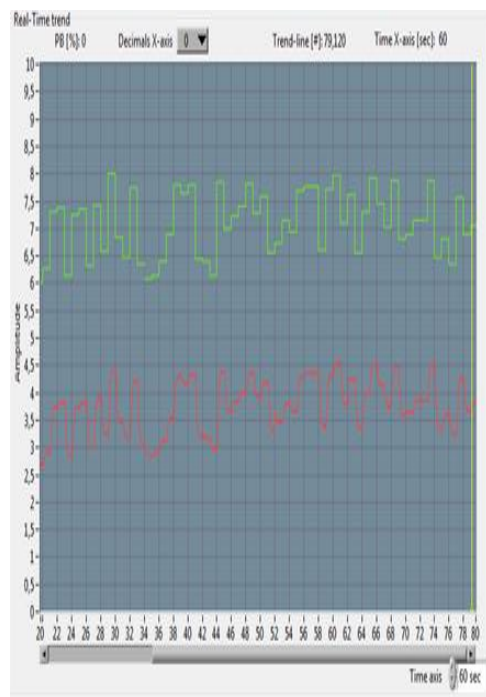
Untuk melihat respon dari model, maka model tersebut disimulasikan dan hasil responnya dapat dilihat pada gambar 11 berikut.



Gambar11. Respon dari model

Noise Measurements

Dari hasil pengukuran *noise measurement*, data dapat direpresentasikan pada gambar 12 berikut:

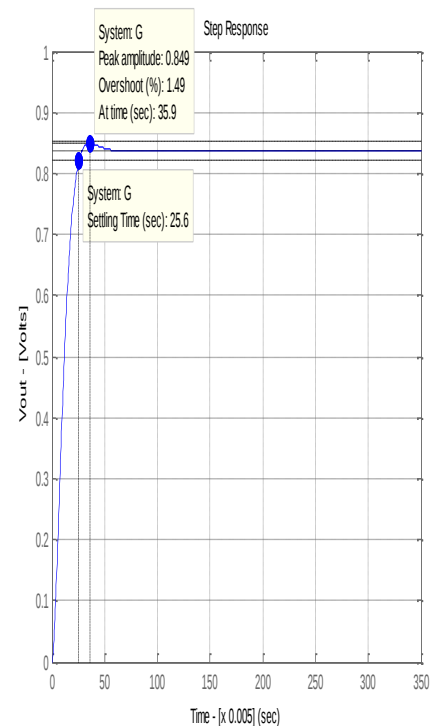


Gambar 12. Noise Measurement data

Dengan menggunakan By using Output error identification system, maka fungsi transfer dapat diperoleh seperti terlihat pada persamaan berikut:

$$H_p = \frac{0.008377s + 0.0174}{s^2 + 0.2311s + 0.02079} \quad (8)$$

Dan respon step yang diperlihatkan pada gambar 13 dapat diobservasi



Gambar 13. Step response

Perancangan Kontroler

Langkah selanjutnya adalah mendesain kontroler yang cocok. Kontroler didesain berdasarkan fungsi transfer dari sistem sebagaimana langkah-langkah yang dilakukan sebelumnya. Adapun kebutuhan spesifikasi dari sistem adalah:

1. *Stable*
2. *Robust : Phase Margin*
(PM) > 45° *C and Gain Margin*
(GM) > 6dB
3. *Steady state error : zero*
4. *Overshoot : less than 4.3%*
5. *Settling time for 10% : as fast as possible.*

Analisa Model Sistem

Berdasarkan hasil pemodelan diatas, ada tiga fungsi transfer yang diperoleh yaitu fungsi transfer *step test(positive and negative)*, *bode measurement and noise measurement*. Dan semuanya adalah Sistem orde dua. Dengan membandingkan ketiga fungsi transfer antara model dan hasil pengukuran dan melihat karakteristik dari *step responnya*, maka diputuskan untuk menggunakan Fungsi transfer dari *Bode measurement test* untuk mendesain kontroler. Sebagai contoh, hal ini dapat dilihat pada *settling time of step of bode measurement* adalah lebih kecil dibandingkan dengan yang lainnya. Lihat gambar 6,7,13 dimana ini menunjukkan bahwa responnya lebih cepat dibandingkan dengan yang lainnya. Dan dari hasil *step response* juga tidak terdapat *overshoot* dibandingkan dengan yang ada pada *noise measurement*. Alasan lainnya adalah bahwa pada *bode measurement* melibatkan cukup banyak sinyal dinamik baik untuk input maupun output. Oleh karena itu, digunakan persamaan fungsi transfer dari *bode measurement* berikut untuk didesain PID controller.

$$H_p(s) = \frac{0.8925}{(1 + s/2)(1 + s/6)} \quad (9)$$

atau

$$H_p(s) = \frac{10.71}{(s + 2)(s + 6)} \quad (10)$$

Performances of uncompensated system

Sebelum melakukan perancangan kontroler, Performans dari *the uncompensated system* dievaluasi terlebih dahulu kemudian barulah ditentukan kontroler yang cocok. Pada bagian ini, kami menginvestigasi performans dari *the uncontrolled system* yang bekerja pada area 3 % overshoot (damping ratio= 0.745) dengan menggunakan root locus tool. Hasil lengkap dari performance of the uncompensated system diperlihatkan pada tabel 3.

Tabel 3. *The uncompensated system performances*

Characteristics	Values
Plant	$G(s) = \frac{K(10.71)}{(s+2)(s+6)}$
Dominant poles	$-3.99 \pm j3.57$
K	1.57
ζ	0.745
ω_n	5.36
%OS	3
T_s	1.08
T_p	0.88
Peak Value	0.601
K_p	1.4
e_{ss}	0.417

Perancangan PD Kontroler

Dari performa yang diperlihatkan oleh uncompensated system, *settling timenya* adalah 1.08 second. Adapun settling time yang diinginkan adalah 10% lebih cepat. Dalam hal ini dipilih *settling time* = 0.36. berdasarkan *overshoot %OS* = 3 pdan garis di *sisotool* diperoleh nilai *damping ratio* $\zeta = 0.745$. Berdasarkan nilai ini, dapat dihitung dominant poles yang diinginkan.

$$\omega_n = \frac{4}{\zeta \cdot T_s} = \frac{4}{0.745 \cdot 0.36} = 14.9142$$

Berikut dominant poles yang diperoleh:

$$p_{1,2} = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2}$$

$$p_{1,2} = -11.11 \pm j9.946$$

Dengan menghitung *the compensating zero's location*. Dengan menggunakan program *root locus*, diperoleh jumlah sudut dari *uncompensated system's poles* ke *desired compensated dominant pole* yaitu sebesar -249.688° . Oleh karena itu kontribusi yang dibutuhkan dari kompensator zero adalah $-249.688^\circ - 180^\circ = 69.7^\circ$ $z_c = 14.79$

Nilai tersebut kemudian disubsitusikan ke dalam fungsi transfer dari kontroler PID

$$H_{PID}(s) = K_p \cdot a \cdot \left(\frac{s + \frac{1}{\tau_i}}{s} \right) \left(\frac{s + \frac{1}{\tau_d}}{s + \frac{a}{\tau_d}} \right) \quad (11)$$

$$z_c = \frac{1}{\tau_d} \text{ and } p_c = \frac{a}{\tau_d}$$

Dengan menggunakan faktor tamed $a = 10$, maka diperoleh nilai

$$\tau_d = \frac{1}{14.79} = 0.0676$$

$$p_c = \frac{10}{0.0676} = -147.929$$

Dan fungsi transfer dari Kontroler PD adalah:

$$G_{PD}(s) = a \cdot \frac{(s + z_c)}{(s + p_c)} = 10 \cdot \frac{(s + 14.79)}{(s + 147.929)}$$

Fungsi transfer dari kontroler PD kemudian ditempatkan *original root locus the uncompensated system*, oleh karena itu fungsi transfernya menjadi sebagai berikut:

$$G(s) = \frac{K \cdot 10 \cdot (10.71)(s + 14.7929)}{(s + 2)(s + 6)(s + 147.929)} \quad (12)$$

Performa dari sistem kontroler PD dapat dilihat pada tabel 4 berikut:

Tabel 4. Performa sistem kontroler PD

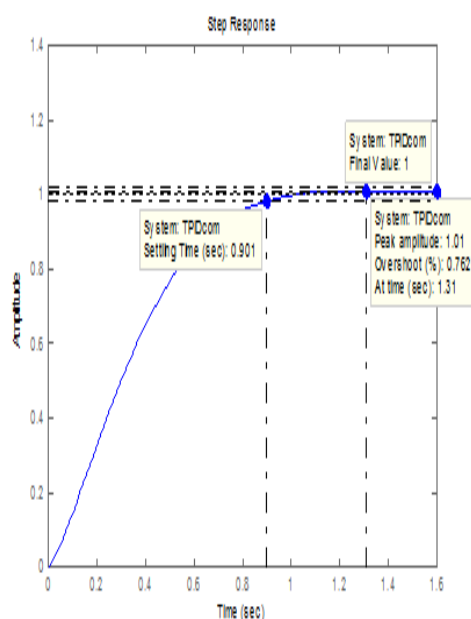
Characteristics	Values
Plant	$G(s) = \frac{K \cdot 10 \cdot (10.71)(s + 14.7929)}{(s + 2)(s + 6)(s + 147.929)}$
Dominant poles	$-5.13 \pm j4.58$
K	3.23
ζ	0.746
ω_n	6.87
%OS	3.61
T_s	0.782
T_p	0.588
Peak Value	0.772
K_p	2.877
e_{ss}	0.258
Comment	

Perancangan PID Kontroler

Setelah perancangan kontroler PD, kemudian PI kontroler didesain untuk mengurangi *error steady state* menjadi nol untuk input step. Untuk penelitian ini, untuk zero pole pada posisi -2 untuk membuat respon transien tidak begitu besar berubah. Fungsi transfer dari dimasukkan pada *root locus the uncompensated system and G_{PD} compensated.*, oleh karena itu PID kontroler menjadi:

$$G(s) = \frac{K \cdot 10 \cdot (10.71)(s + 14.7929)(s + 2)}{(s + 2)(s + 6)(s + 147.929)s} \quad (13)$$

Respon akhir dari kontroler dapat dilihat pada gambar 14 dan dapat diambil kesimpulan bahwa karakteristik dari sistem kontroler PID seperti tergambar pada tabel 5.



Gambar 14. Respon dari *PID Controlled System*

Tabel 5 Performa *PID-controlled system*

Characteristics	Values
Plant	$G(s) = \frac{K \cdot 10 \cdot (10.71)(s + 14.7929)(s + 2)}{(s + 2)(s + 6)(s + 147.929)s}$
Dominant poles	$-3.57 \pm j2.27$
K	1.66
ζ	0.843
ω_n	4.23
%OS	0.762
T_i	0.901
T_p	1.31
Peak Value	1.01
K_p	∞
e_{ss}	0

Fungsi transfer dari Kontroler PID adalah sebagai berikut:

$$G_{PID}(s) = \frac{1.66 \cdot 10(s + 2)(s + 14.7929)}{s(s + 147.929)} \quad (14)$$

Oleh karena itu diperoleh parameter PID sebagai berikut.

$$K_p = 1.66$$

$$\tau_d = 1/14.7929 = 0.0676 / 0.068$$

$$\tau_i = 1/2 = 0.5$$

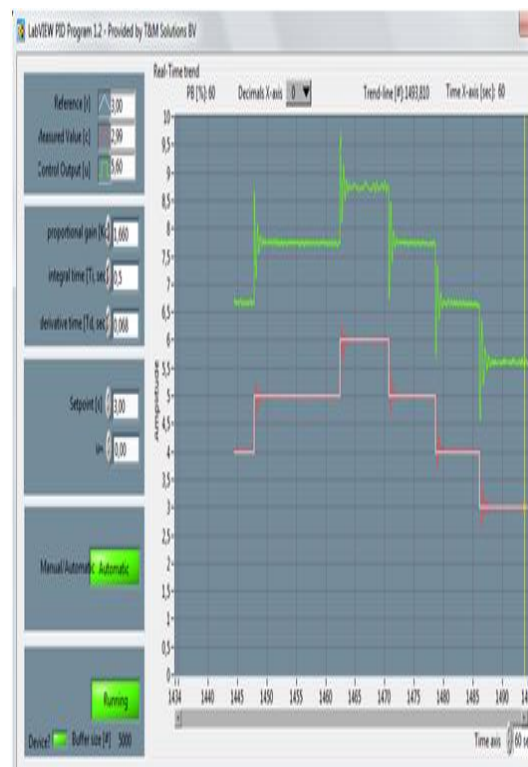
Implementasi

Setelah memperoleh parameter dari PID kontroler (parameter K_p , τ_i and τ_d) kemudian parameter ini diimplementasikan pada sistem yang riil yaitu *flow control plant*.

Kontroler dari hasil pemodelan

Implementasi pertama

Pada langkah ini, nilai dari parameter $K_p=1,66$, $\tau_i=0,5$ dan $\tau_d=0,068$ yang diperoleh dari hasil sebelumnya diimplementasikan ke riilsistem dan sistem memberikan respon seperti yang terlihat pada gambar 15.

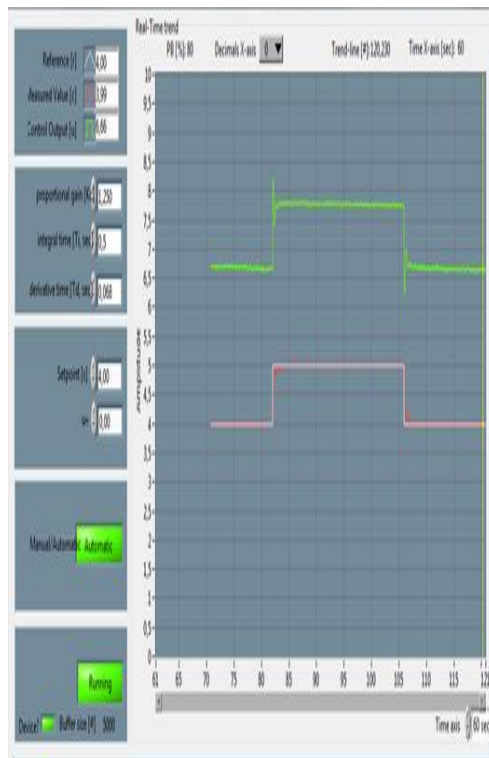


Gambar 15. Hasil Implementasi pertama dari kontroler PID

Berdasarkan gambar 15, dapat dikatakan bahwa respon sistem menunjukkan hasil yang memuaskan dimana steady state mendekati nol atau boleh dikatakan nol, *quick in settling time*, akan tetapi ada sedikit *overshoot* yang terjadi. Dalam kasus ini, diduga bahwa adanya *overshoot* karena data yang diperoleh dari hasil pengukuran tidak selalu sama pada sat sistem diuji dan disamping itu adanya perbedaan antara simulasi dan proses riil. Ini berarti bahwa karakteristik dari sebuah sistem biasanya berubah karena adanya pengaruh lingkungan. Sebagai contoh bahwa keakurasian dari sebuah sistem berkurang oleh waktu. Oleh karena itu untuk mengurangi *overshoot* pada respon sistem, maka *gain proportional* K_p dikurangi dan tetap menjaga nilai dari τ_i and τ_d . Ini berarti bahwa tidak ada perubahan pada *root locus design*, tetapi hanya untuk mendapatkan nilai *gain* K_p yang cocok..

Implementasi Kedua

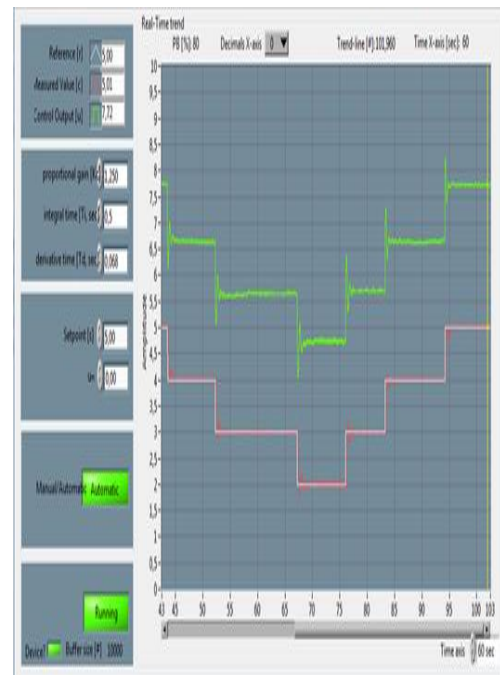
Sebagaimana yang telah dijelaskan sebelumnya bahwa untuk mengurangi *overshoot* maka *gain proportional* K_p dikurangi dan tetap menjaga nilai τ_i and τ_d . Setelah melakukan *tuning* pada nilai K_p , diperoleh nilai parameter yang cocok yaitu $K_p=1,25$, dan nilai dari $\tau_i=0,5$ and $\tau_d=0,068$ tetap dijaga. Gambar 16 memperlihatkan respon sistem dengan $K_p=1,25$, $\tau_i=0,5$ and $\tau_d=0,068$.



Gambar 16. Hasil Implementasi kedua dari kontroler PID

Berdasarkan respon sistem yang terlihat pada gambar 16, maka dapat disimpulkan bahwa respon sistem sangat baik dimana *overshoot* hampir tidak ada, *a quick in the time settling* dan *error steady state* sama dengan nol.

Gambar 16 memperlihatkan bahwa kontroler PID dapat mengontrol sistem dengan sangat baik. Garis putih adalah representasi dari nilai yang diinginkan. Garis merah berkorespondensi dengan nilai keluaran dan garis hijau menunjukkan sinyal input kontrol PID. Respon sistem terlihat stabil dan *robust* terhadap adanya *disturbance*. Nilai dari *steady state error* sama dengan nol dimana tidak ada perbedaan antara nilai yang diinginkan dan nilai output.



Gambar 18. Sistem respon dengan nilai input yang berbeda

Pada gambar 18 terlihat bahwa akan terjadi ada perubahan overshoot dan dan settling time tergantung pada nilai output yang diinginkan. Dari hasil ini dapat disimpulkan bahwa :

Robust: sistem stabil terhadap adanya disturbance terlihat pada stabilnya sistem dengan adanya perubahan input.

overshoot: berubah 0-3 % tergantung nilai output yang diinginkan

settling time untuk 10 %: biasanya (1-3 seconds) tergantung output yang diinginkan.

Flow loop control pada riil proses dibatasi oleh tegangan input. Dimana sistem beroperasi pada 0 – 10 V. Pada Gambar 19, kita dapat melihat bahwa sistem tidak dapat mengikuti nilai yang diinginkan lihat garis putih). Dimana output maksimal dibatasi pada nilai 7.5 volt.

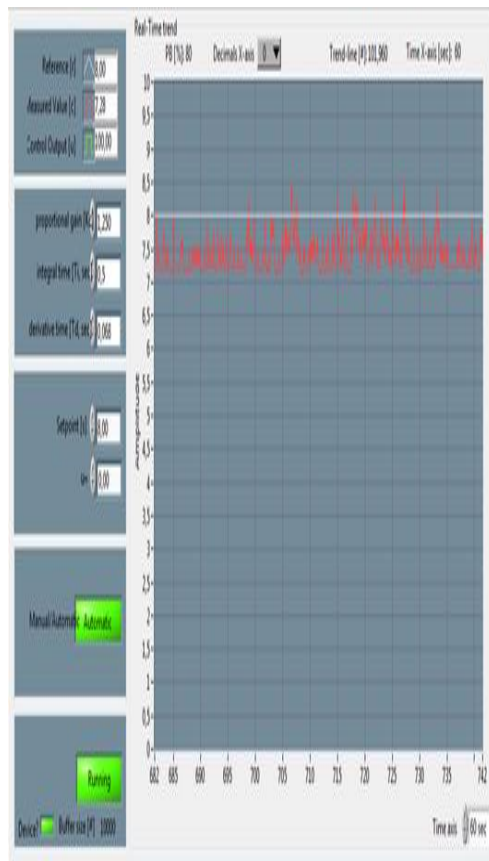


Figure 19. Respon Keterbatasan Sistem

KESIMPULAN

1. Telah digambarkan proses dalam pengontrolan keluaran sistem. Langkah pertama dengan mendapatkan informasi tentang *linear range*, *point of operation* dan juga *gain*. Pada perkiraan awal karakteristik dinamik sistem dengan cara mendapatkan data dari *step response* and informasi lebih detail diperoleh dari *bode plot*. Setelah membandingkan ketiga respon dari 3 model dari *transfer function step measurement*, *bode measurement* and *noise measurement* maka diputuskan untuk menggunakan model dari *bode measurement*.
2. Berdasarkan *transfer function of bode measurement*, didesain kontroler PID dengan menggunakan metode *root locus* dan diperoleh parameter $K_p = 1,66$, $\tau_i = 0,5$ and $\tau_d = 0,068$. Setelah mengimplementasikan nilai parameter tersebut, diperoleh bahwa respon dari sistem cukup bagus, dimana tidak ada *error steady state*, *quick in settling time*, tetapi ada sedikit *overshoot*.
3. Dengan mengurangi nilai dari gain proportional (K_p) dari 1,66 menjadi 1,25 dapat dikurangi *overshoot* dari proses atau sistem dengan menjaga nilai τ_i and τ_d .
4. Setelah mengimplementasikan nilai dari parameter $K_p = 1,25$, $\tau_i = 0,5$ and $\tau_d = 0,068$ pada sistem, terlihat bahwa kontrol PID mengontrol sistem atau proses dengan sangat baik tak ada *overshoot*, tak ada *error steady state* dan a *quick in settling time*.
5. Penggunaan parameter hasil rancangan dan melihat hasil respon dari simulasi pada software matlab terlihat bahwa kontrol PID hasil rancangan memberikan hasil yang sesuai dengan kebutuhan spesifikasi sistem dimana sistem stabil, *robust*, tak ada *overshoot*, dan memiliki *settling time* yang cepat tak ada *error steady state*.
6. Pada rill proses, sistem terlihat stabil, *robust* terhadap berbagai *disturbance*. Respon sistem juga terlihat cepat dan stabil pada area operasi. *Overshoot* pada proses berkisar antara 0-3%. Dimana

Asriyadi, Implementasi Kontroler Pid Pada Simulator Kontrol Aliran 59

hal ini sangat tergantung pada nilai input. Secara umum dapat disimpulkan bahwa sistem dengan adanya kontroler PID memenuhi spesifikasi kebutuhan.

7. Nilai maximum keluaran pada sistem adalah sebesar 7.5 volt. Diduga disebabkan oleh keterbatasan power dari pompa air dan juga keterbatasan *power supply*.

DAFTAR PUSTAKA

- Matlab, User Manual, The Mathworks,
- Nise, N.S, Control System Engineering, John Wiley and Sons, Inc, 2011
- PID Tools, Manual HAN PID tools, HAN University, 2013
- Tazelaar, E. 2004. *Manual Laboratory of Control System Engineering. HAN University of Applied Science*, 2011 Netherland.