

PERANCANGAN *TRAINER PID ANALOG* UNTUK MENGATUR KECEPATAN PUTARAN MOTOR DC

M. Subchan Mauludin^{1*}, Andi Kurniawan²

¹Jurusan Teknik Informatika, Fakultas Teknik, Universitas Wahid Hasyim
Jl. Menoreh Tengah X/22, Sampangan, Semarang 50236.

²Jurusan Teknik Elektronika, Fakultas Teknik Universitas Semarang
Jl. Soekarno-Hatta Semarang

*Email: aan.subhan18@gmail.com

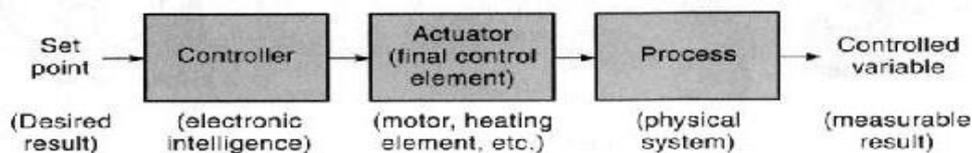
Abstrak

PID (Proportional Integral Derivative) merupakan kontroler untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Didalam suatu sistem kontrol mengenal adanya beberapa macam aksi kontrol, diantaranya yaitu aksi kontrol proporsional, aksi kontrol integral dan aksi kontrol derivative. Masing-masing aksi kontrol ini mempunyai keunggulan-keunggulan tertentu, aksi kontrol proporsional mempunyai keunggulan risetime yang cepat, aksi kontrol integral mempunyai keunggulan untuk memperkecil error, dan aksi kontrol derivative mempunyai keunggulan untuk memperkecil derror atau meredam overshoot/undershoot. Untuk itu agar dapat menghasilkan output dengan risetime yang tinggi dan error yang kecil kita dapat menggabungkan ketiga aksi kontrol ini menjadi aksi kontrol PID. *Trainer Kendali PID Analog* menggunakan IC 741 sebagai perangkat op Amp. Tujuan dari penelitian ini adalah untuk mendapatkan nilai K_p , K_i , dan K_d yang ideal sehingga dapat mengendalikan plant motor DC.

Kata Kunci: Risetime, error, overshoot, PID Analog, IC 741, motor DC

Pendahuluan

Sistem kontrol adalah suatu sistem yang berfungsi untuk mengatur pengontrolan suatu *plant* dengan cara mengatur *input*-nya. Semua sistem kontrol memiliki kontroler dan *actuator*. *Input* pada kontroler biasanya disebut sebagai *set point*. *Actuator* pada alat elektromekanik mengambil sinyal dari kontroler dan diubah dalam bentuk mekanik. Diagram blok sistem kontrol secara umum terdapat pada gambar di bawah ini.



Gambar 1. Diagram Blok Sistem Kontrol

Sumber : Kilian, Christopher T. 2001. *Modern Kontrol Technology Components And System*. St. Paul: West Publishing Company

Sistem kontrol dapat digolongkan menjadi dua bagian, yaitu sistem kontrol *loop* terbuka (*open loop*) dan sistem kontrol tertutup (*closed loop*). Perbedaan sistem kontrol *loop* terbuka dan sistem kontrol *loop* tertutup adalah pada ada tidaknya *feedback* pada sistem.

Jika suatu sistem memiliki *feedback*, maka *output*-nya akan berpengaruh pada proses kontrol. Kebanyakan sistem kontrol menggunakan istilah *error* sebagai *feedback* bagi sistem. *Error* adalah perbedaan dari nilai SP (*Set Point*) dan PV (*Present Value*).

Pada industri-industri yang membutuhkan suatu sistem kontrol dengan kecepatan tinggi dan keakuratan data output, maka pemakaian aksi kontrol PID mungkin masih dianggap kurang memuaskan. Sebab jika menggunakan aksi kendali PID didapatkan jika suatu kontroler di set sangat sensitif, maka *overshoot/undershoot* yang dihasilkan akan semakin peka, sehingga osilasi yang ditimbulkan akan lebih tinggi, sedangkan bila kontroler di set kurang peka maka terjadinya *overshoot/undershoot* dapat diperkecil, tetapi waktu yang dibutuhkan akan semakin lama, dan ini akan menjadikan suatu masalah dalam suatu proses industri.

Dasar Teori

Kontrol Proporsional

Pada sistem kontrol *proporsional* ini hubungan antara *output* dan *error* adalah sebagai berikut :

$$M(t) = K_p \cdot e(t) \quad (1)$$

Fungsi alihnya dalam transformasi *Laplace* adalah sebagai berikut :

$$\frac{M(s)}{E(s)} = K_p \quad (2)$$

Salah satu dari ketiga mode unit kontrol yang paling populer dan paling banyak dipakai adalah *unit* kontrol P. Seperti yang tercermin dari namanya, besar *output unit* kontrol P selalu sebanding dengan besarnya *input*. Bentuk dari *transfer function* sangat sederhana dan bentuk diagram bloknnya juga sederhana. Seperti yang tergambar dalam gambar 2. *Unit* kontrol P adalah *unit* kontrol yang paling banyak dipakai, baik dalam bentuk pengendali tunggal maupun dengan kombinasi mode *integral* (I) dan *differensial* (D).



Gambar 2. Diagram Blok Pengendali Proporsional

Sumber : Gunterus, Frans. 1994. *Falsafah Dasar : Sistem Pengendalian Proses*. PT Elex Media Komputindo

$$O = G_c \cdot i \quad (3)$$

Gain (G_c) *unit* kontrol *Proporsional* bisa berupa bilangan pecahan. Besarnya tetap, *linier* pada semua daerah kerja dan tidak tergantung pada fungsi waktu. Sepintas istilah *gain* memberikan kesan bahwa ada penguatan dan pembesaran sinyal. Istilah *gain* biasanya jarang dipakai, biasanya menggunakan istilah *Proportional Band* (PB), dimana :

$$G_c = \frac{100\%}{PB} \quad (4)$$

Gain atau disebut dengan PB, dapat diatur besarnya sesuai dengan kebutuhan. Dari persamaan 4 diketahui bahwa *gain* (G_c) berbanding terbalik dengan PB, kalau *gain* semakin kecil maka PB semakin besar.

Kekurangan dari pengendalian *proporsional* adalah timbulnya *offset*, dimana besarnya *offset* tergantung pada besarnya *gain* elemen-elemen di dalam *loop* serta *Proportional Band* (PB). Hal ini disebabkan oleh sifat dasar pengendali *proporsional* yang membutuhkan *error* yang menghasilkan *output*.

Kontrol Pengendali Integral

Kalau diteliti lebih seksama, *offset* dapat terjadi di pengendali *proporsional* (karena selalu membutuhkan *error*). Jadi untuk menghilangkan *offset*, diperlukan sebuah pengendali yang lain (yang dapat menghasilkan *output* walaupun padanya tidak diberikan *input*). Dengan kata lain, diperlukan pengendali yang menghasilkan *output* lebih besar atau lebih kecil dari bilangan tetap (B) pada saat *input* (*error*) sama dengan nol. Pengendali yang memenuhi kriteria ini adalah pengendali *integral*, disingkat I.

Sifat dasar pengendali *integral*, yang dapat mengeluarkan *output* pada saat *input* sama dengan nol, adalah sifat dari *unit integrator* yang prinsipnya sama dengan sifat elemen proses orde satu.

Transfer function dari *unit* kontrol *integral* adalah sebagai berikut

$$O = \frac{1}{T_r} G_c \int e dt + B \quad (5)$$

Dimana :

O = *output*

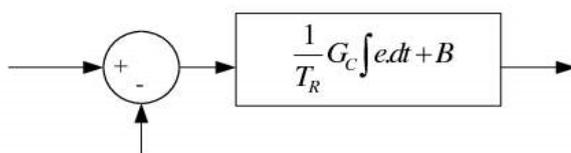
e = *error* (*input* dari *unit* kontrol)

T_r = *integral time*

B = Bilangan tetap (yang merupakan *bias* atau hasil dari *integral* sebelumnya)

G_c = *gain* dari *controller*

Besarnya *integral time* (T_r) dinyatakan dalam satuan *minute / repeat*. Artinya, sebuah pengendali *integral* dengan $G_c = 1$, dikatakan mempunyai *integral time* 2 *minute / repeate* apabila pengendali memerlukan waktu 2 menit untuk mencapai *output* sama dengan *input*. Jadi unsur yang diperhatikan dalam hal ini adalah unsur waktu (*time*). Sebaliknya jika nilai *minute / repeat* kecil, reaksi pengendali akan semakin cepat atau semakin sensitif. Kalau sebaliknya nilai *time / repeat* besar, pengendali akan semakin lambat, atau pengendali akan kurang sensitif. *Integral time* yang kecil pada dasarnya lebih menguntungkan. Walaupun, seperti PB yang kecil, *integral time* yang kecil akan membuat *loop* lebih mudah berisolasi. Itu berarti bahwa keadaan tidak stabil akan menjadi lebih mudah terjadi apabila *integral time* dibuat terlalu kecil.



Gambar 3. Diagram Blok Pengendali Integral

Sumber : Gunterus, Frans. 1994. *Falsafah Dasar : Sistem Pengendalian Proses*. PT Elex Media Komputindo

Kontrol Pengendali *Differential*

Karena lambatnya kontrol pengendali PI dalam pengendalian sistem. Upaya memperbaiki *respon* didapat dengan menggunakan *unit* kontrol *differential* atau *derivative*, disingkat D. *Output* pengendali D merupakan *differential* dari fungsi *input*. Sayangnya unsur D tidak dapat mengeluarkan *output* (bila tidak ada perubahan *input*). Karena sifat ini, pengendali D tidak pernah dipakai sendirian. *Unit* pengendali D selalu dipakai dalam kombinasinya dengan P dan I, menjadi pengendali PD atau PID. Selain itu, pengendali D tidak dapat dipakai untuk proses variable yang mengandung *noise*. Karena banyaknya kendala dalam pengendali D, populasi pengendali PID dan PD menjadi tidak sebanyak pengendali P atau PI, (Gunterus 1997, p, 8-3) menyajikan persamaan *transfer function* (O) lengkap dari pengendali *differential* sebagai berikut :

$$O = G_c \cdot T_D \frac{ds}{dt} + B \quad (6)$$

Dimana :

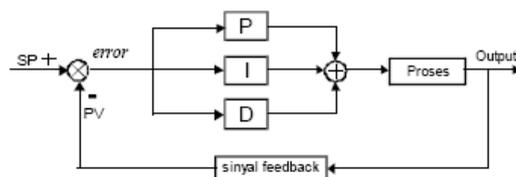
G_c = *gain*

E = *Error*

T_D = *derivative time*

B = *bias*

Unsur *derivatif* yang ada pada *unit* kontrol ini, jika diberi *input* yang naik secara perlahan – lahan dalam bentuk fungsi *ramp*, maka *output* berfungsi *step*. Besarnya *output* tersebut tergantung pada kecepatan naiknya *input* dan T_D . Oleh karena itu pengendali ini juga disebut *rate controller*.

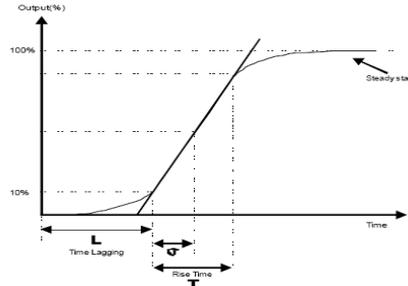


Gambar 4. Blok Diagram Kendali PID

Untuk mendapatkan harga-harga K_p , $T_i (=1/K_i)$, $T_d (=K_d)$ menurut *Ziegler nichols* ditentukan oleh kurva proses reaksi, dimana sistem dijalankan secara *open loop*.

L = *Time lagging*

$T = \text{Time konstan}$



Gambar 5. Kurva Proses Respon Motor

Metodologi Penelitian

Menentukan Plant Model Motor DC

Motor DC mempunyai prinsip kerja berdasarkan percobaan *lorentz* yaitu sebatang penghantar listrik yang berarus berada didalam medan magnet, maka pada penghantar tersebut akan menimbulkan torsi yang menimbulkan gerakan. pada motor DC bagian yang bergerak (*rotor*) diubah menjadi gerakan yang berputar. Sehingga dihasilkan persamaan :

$$\frac{\omega(s)}{V(s)} = \frac{K}{(Js + \beta)(R + Ls) + K^2} \quad (7)$$

Menentukan pengendali

PID pada dasarnya merupakan suatu proses dari suatu program yang dijalankan / diexecute dengan menggunakan komputer, dimana kita memasukkan nilai *Setting Point* (SP) dan *Present Value* (PV), yang kemudian data yang didapatkan diproses sehingga *error* yang didapatkan sama dengan 0, atau nilai *Setting Point* = *Present Value*. Untuk dapat mengimplementasikan sistem kendali PID pada komputer, PID harus diubah ke dalam persamaan diskrit
Hasil akhir dari persamaan PID yaitu :

$$V_o - V_{on-1} + Kp(e_n - e_{n-1}) + Ki e_n Ts + \frac{Kd}{Ts} (e_n - 2e_{n-1} + e_{n-2}) \quad (8)$$

Keterangan :

- V_o = Output
- V_{on-1} = Output sebelumnya
- Kp = Konstanta *Proporsional*
- Ki = Konstanta *Integral*
- Kd = Konstanta *derivative*
- e_n = *Error* sekarang
- e_{n-1} = *Error* 1 kali sebelumnya
- e_{n-2} = *Error* 2 kali sebelumnya
- Ts = *Time Samplin*

Hasil dan Pembahasan

Pengujian Kendali *Proportional*

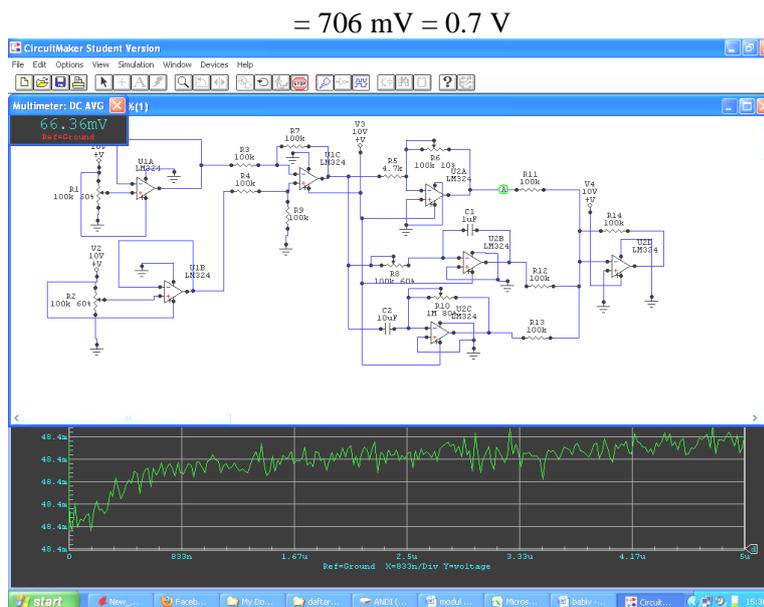
Rangkaian *proportional* menghasilkan tegangan keluaran sebesar 66.36 mV dengan gain penguatan sebesar :

$$A_v = -\frac{R_f}{R_{in}}$$

$$= -\frac{50 \text{ K}}{4,7 \text{ K}}$$

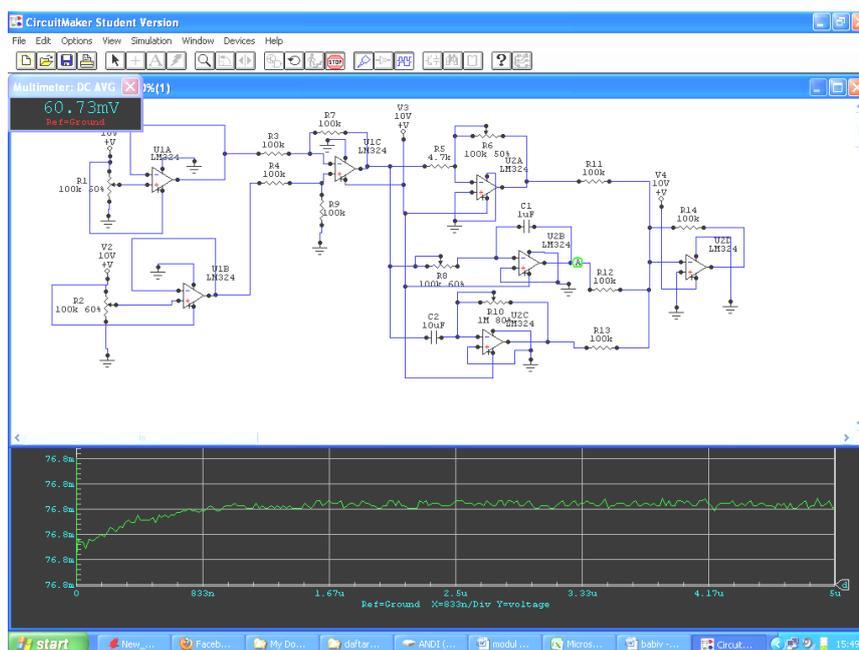
$$= 10.64$$

Maka tegangan keluaran *proportional* = 10.64 x 66,36mV



Gambar 6 . Gelombang keluaran rangkaian *Proportional*

Pengujian Kendali *Integral*



Gambar 7. Gelombang keluaran rangkaian *Integral*

Rangkaian *integral* menghasilkan tegangan keluaran sebesar 60,73 mV dengan *gain* penguatan (*transfer function*) sebesar :

$$A_v = -\frac{1}{RC}$$

$$A_v = -\frac{1}{50K(x10^{-6}F)}$$

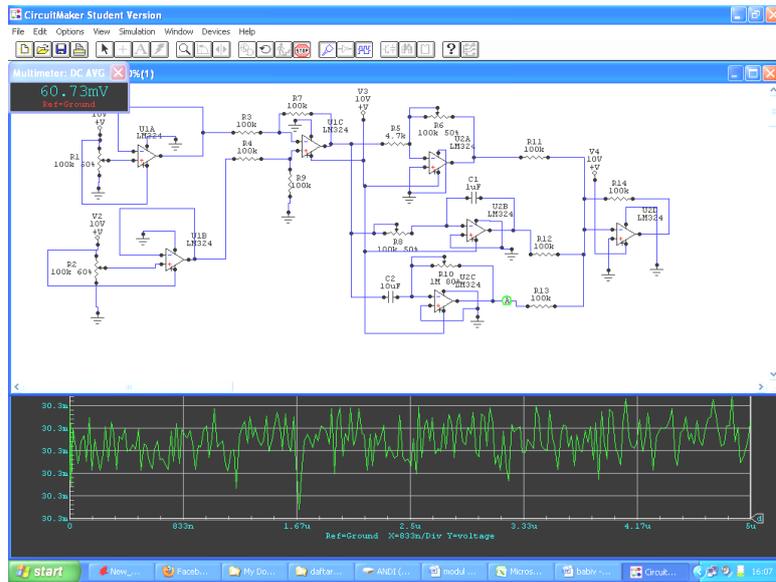
$$A_v = - \frac{20}{\Omega \times \frac{C}{V}}$$

$$A_v = - \frac{20}{\Omega \times \frac{A.s}{V}}$$

$$= - 20 / s$$

Maka tegangan keluaran $integral = -20/s \times 60,73mV$
 $= -1214 mV/s = -1,2 V/s$

Pengujian Kendali *Differential*



Gambar 8. Gelombang keluaran rangkaian *Differential*

Rangkaian *Differential* menghasilkan tegangan keluaran sebesar 66,73 mV dengan *gain* penguatan (*transfer function*) sebesar :

$$A_v = (-RC) s$$

$$= 50 k\Omega \times 10^{-5}F$$

$$= (-0.5) s$$

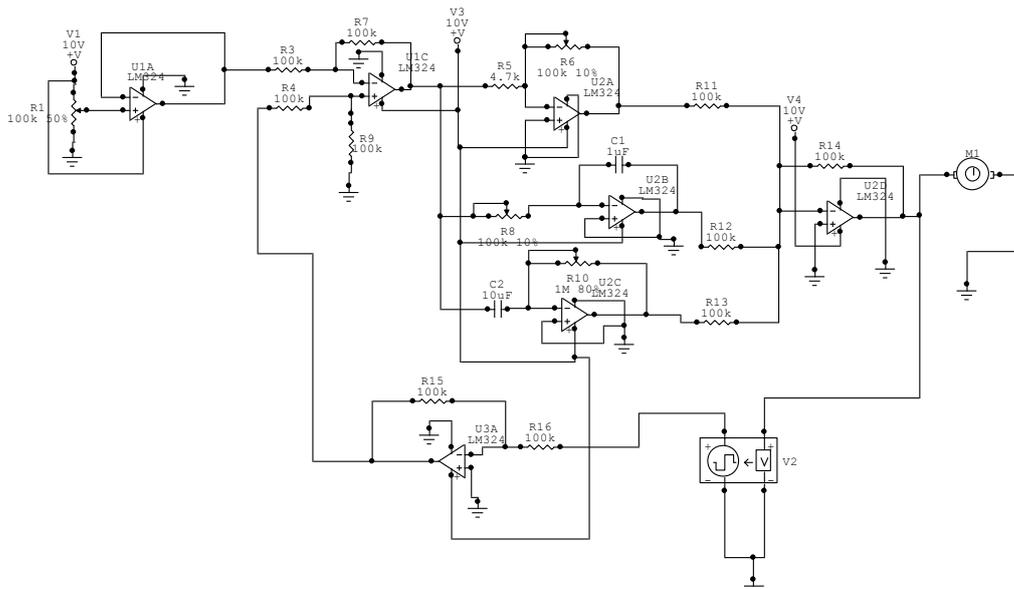
Maka tegangan keluaran $differential = - 0.5 /s \times 66,73mV$
 $= - 33,365 mV/s = -0,3 V/s$

Pengujian PID trainer secara loop tertutup

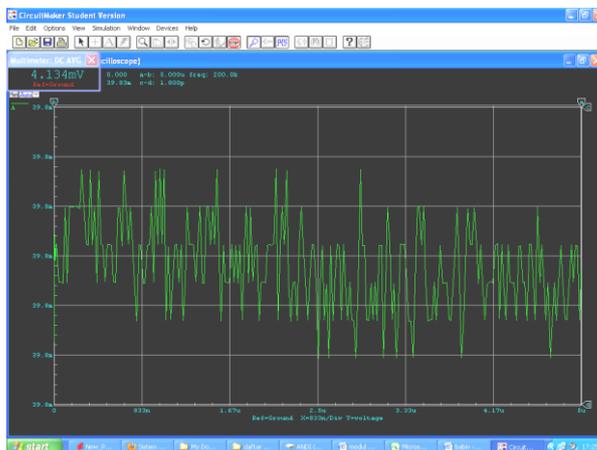
Hasil keluaran pada

1. rangkaian *inverter* = 39.83 mV
2. keluaran *subtraction* = 4.134 mV

Rangkaian VCO (*Voltage Controlled Oscilator*) berguna untuk melakukan koreksi balikan pada keluaran motor sehingga pengendalian PID sangat bergantung pada masukan rangkaian *subtraction*.

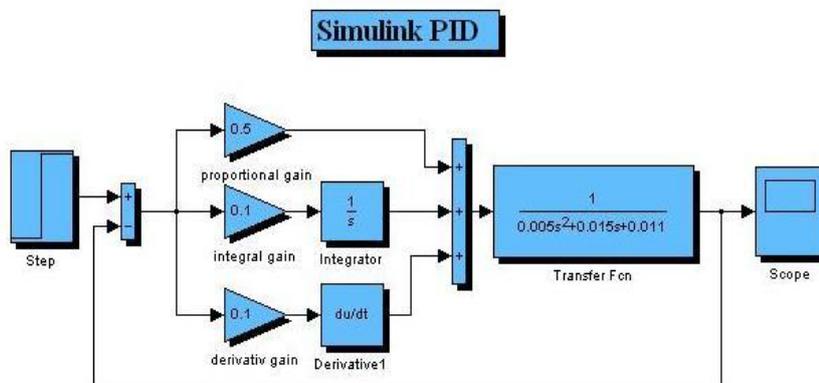


Gambar 9. Pengujian kontrol PID dengan beban motor DC dan sensor VCO

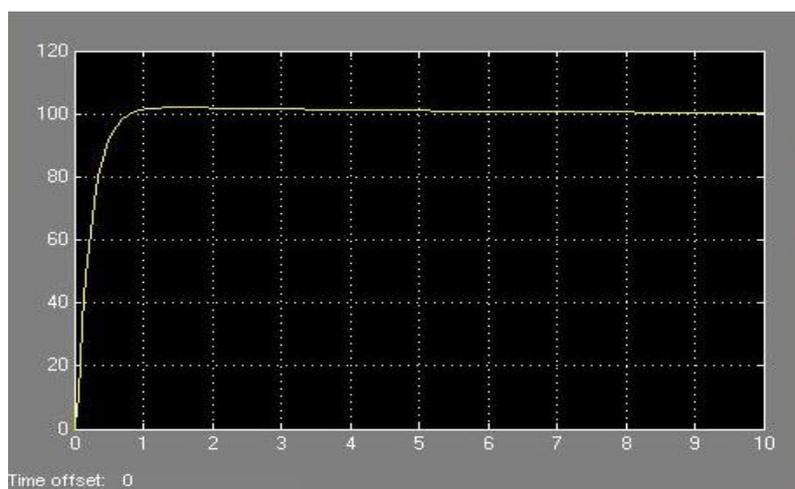


Gambar 10. Gelombang keluaran rangkaian *inverter*

Pengujian Kontrol PID Menggunakan Matlab



Gambar 11. Blok Simulink Model PID Kontrol



Gambar 12. Respon Pengendali PID dengan $K_p=0.3$, $K_i=0.1$ dan $K_d=0.1$

Dari gambar kendali PID dengan setting point 100, nilai *rise time* = 1.12 detik, *peak time* = 6.72 detik dan nilai *settling time* = 9.8 detik, *overshoot* maksimal (M_p) yang dicapai adalah 102.0092 sehingga persen *overshoot* dihitung sebagai berikut:

$$\begin{aligned} \%overshoot &= \frac{mp - sp}{sp} \times 100\% \\ \%overshoot &= \frac{102.0092 - 100}{100} \times 100\% \\ \%overshoot &= 0.020092 \end{aligned}$$

overshoot yang terjadi sekitar 0.020092 dengan demikian sistem sudah mendekati stabil.

Kesimpulan

Berdasarkan hasil penelitian di dapatkan :

1. Besar tegangan keluaran $K_p = 0.7$ V, $K_i = -1,2$ V, dan $K_d = 0.3$ V.
2. Dengan merubah nilai dari K_p , K_d , K_i maka didapatkan nilai % *overshoot* terkecil $K_p=0.3$, $K_d=0.1$ dan $K_i=0.1$ merupakan respon yang paling stabil.

Daftar Pustaka

- Guntereus, Frans, 1994, *Sistem Pengendali Proses*, PT. Elex Media Komputindo, Jakarta.
- Kontroler PID-Ziegler-Nichols pada Sistem Kontrol posisi berbasis komputer IBM-PC, <http://www.Elektro Indonesia.com/>, Maret 1998
- Mallesham, Gaddam, Rajani, Akula, *Automatic Tuning Of PID Controller Using Fuzzy Logic*, University College of Engineering, Osmania University, Hyderabad, India
- Ogata, Katsuhiko, 1997, *Modern Control Engineering*. Prentice Hall International.
- Putranto, Agus, dan kawan-kawan, 2008, *Teknik Otomasi Industri untuk SMK*, Departemen Pendidikan Nasional
- Pitowarno, Endro, 2006, *Robotika : Desain, Kontrol, dan Kecerdasan Buatan*, Andi, Yogyakarta.
- Sumathi, Sivanandam, and Deepa, 2007, *Introduction to FuzzyLogic using MATLAB*, Springer-Verlag Berlin Heidelberg