

Perancangan Sistem *Active Fault Tolerant Control* (AFTC) untuk Pengendalian Posisi dengan Kontrol *Cascade* pada Sistem Servo Modular MS150 DC

Tulus Indra Hermawan, Katherin Indriawati

Departemen Teknik Fisika, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS)

e-mail : katherin@ep.its.ac.id

Abstrak—Kesalahan pada komponen seperti aktuator dan sensor dapat mengakibatkan kegagalan sistem salah satunya pada pengendalian posisi motor DC. Pada studi ini dilakukan perancangan sistem *Active Fault Tolerant Control* (AFTC) yang mampu bekerja ketika terdapat kesalahan pada komponen seperti aktuator dan sensor pada pengendalian posisi sistem servo modular MS150 DC sehingga performansi sistem tetap terjaga. Langkah pertama yang dilakukan adalah membuat pemodelan servo modular MS150 DC. Langkah kedua adalah merancang sistem kontrol *cascade* PI-P. Langkah ketiga adalah merancang sistem AFTC secara simulasi dan langkah terakhir adalah merancang sistem AFTC untuk aplikasi *real time*. Sistem tanpa AFTC pada kesalahan sensor secara *real time* mengalami *error steady state* sebesar 11% dan 91% sedangkan sistem dengan AFTC tidak mengalami *error steady state*. Sistem tanpa AFTC pada kesalahan aktuator secara *real time* mengalami *overshoot* sebesar 12,7% dan 28,6%, sedangkan sistem dengan AFTC mengalami *overshoot* lebih kecil sebesar 1,59% dan 11,15%. Hal tersebut terjadi karena sistem AFTC bisa menolerir kesalahan dari sensor dan aktuator sehingga respon sistem menjadi lebih baik.

Kata kunci—Aktuator, *Cascade* PI-P, Sensor, Sistem AFTC, Sistem MS150DC.

I. PENDAHULUAN

MOTOR DC (Direct Current) merupakan perangkat elektromagnetis yang berfungsi mengubah energi listrik menjadi energi mekanik. Motor DC memerlukan suplai tegangan yang searah pada kumparan medan untuk diubah menjadi energi mekanik. Motor arus searah tidak biasa digunakan di aplikasi industri karena semua sistem utilitas listrik menyediakan arus bolak-balik. Namun, untuk aplikasi khusus seperti di pabrik baja, tambang dan kereta listrik terkadang menguntungkan untuk mengubah arus bolak-balik menjadi arus searah dalam penggunaan motor DC. Alasannya adalah karakteristik torsi kecepatan motor dapat divariasikan dengan efisiensi yang tinggi [1].

Salah satu subjek penelitian motor DC yang sering ditemui adalah pengendalian posisi yang diaplikasikan dalam berbagai bidang seperti transportasi, peralatan elektronik, industri dan robotika. Desain pengendalian posisi yang sering digunakan adalah *closed loop* yang merupakan pengendalian paling umum dan sederhana [2].

Dalam pengendalian posisi motor DC satu loop, saat gain ditingkatkan maka input tegangan yang dibutuhkan untuk menghasilkan output akan berkurang. Namun hal tersebut berdampak dengan meningkatnya *overshoot* sehingga sistem menjadi tidak stabil. Untuk mengatasi hal tersebut, maka digunakan sistem pengendalian tertutup dengan kontrol *cascade* [3].

Suatu sistem pengendalian memiliki kinerja yang buruk ketika terjadi kesalahan pada komponen, misalnya pada sensor dan aktuator. Untuk mengatasi kelemahan tersebut maka dikembangkan pendekatan baru untuk menoleransi kerusakan komponen sehingga performansi dan kestabilan sistem tetap terjaga serta dapat mencapai kondisi yang optimal. Sistem pengendalian yang memiliki kemampuan untuk mempertahankan kestabilan dan performansi sistem pada saat terjadi kesalahan pada komponen sensor dan aktuator disebut *Fault Tolerant Control System* (FTCS). *Fault Tolerant Control System* (FTCS) merupakan suatu pengembangan teknologi pengendalian untuk kebutuhan keamanan (*safety*) dan perbaikan unjuk kerja. FTCS dapat dibagi menjadi dua tipe, yaitu *Active Fault Tolerant Control System* (AFTCS) dan *Passive Fault Tolerant Control System* (PFTCS) [4]. Dalam penelitian sebelumnya, telah dilakukan perancangan sistem AFTC pada pengendalian posisi servo modular MS150 DC menggunakan sistem kendali satu *loop* dengan kesalahan pada sensor dan aktuator [5].

Mempertimbangkan penelitian-penelitian yang sudah dilakukan sebelumnya, maka akan dilakukan perancangan sistem *Active Fault Tolerant Control* (AFTC) untuk pengendalian posisi dengan kontrol *cascade* pada sistem servo modular MS150 DC dengan kesalahan pada sensor dan aktuator. Dengan adanya kontrol *cascade*, diharapkan *overshoot* dan osilasi akan berkurang. Selain itu, AFTC diterapkan untuk menolerir kesalahan yang terjadi pada sistem servo modular MS150 DC sehingga sistem dapat bekerja dengan stabil dan performansi tetap terjaga.

II. TEORI PENELITIAN

A. Sistem Servo Modular MS150 DC

Sistem servo modular MS150 DC merupakan blok rangkaian elektronik yang digunakan untuk pengendalian kecepatan dan posisi dari motor DC.



Gambar 1. Sistem Servo Modular MSI50 DC

Sistem servo modular MSI50 DC secara umum terdiri dari beberapa bagian yaitu :

a. Motor DC

Motor DC yang digunakan merupakan motor DC tipe DCM150F yang terdiri dari tiga bagian yaitu motor DC magnet permanen dengan poros yang diperpanjang, *tachogenerator* dengan output diatas unit, dan poros berkecepatan rendah.

b. Servo amplifier

Modular *feedback servo amplifier* SA150D terdiri dari rangkaian transistor yang dapat menggerakkan motor DC dengan dua arah putaran.

c. Attenuator

Attenuator AU150B berfungsi sebagai pemberi hambatan yang terdiri dari dua potensio dengan kapasitas masing-masing sebesar 10 KΩ yang dapat digunakan secara bersamaan.

d. Output Potentiometer

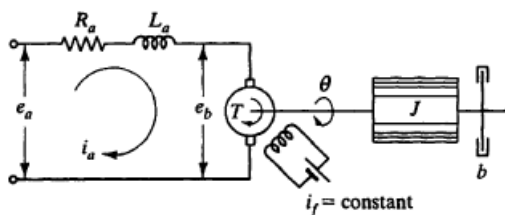
Potensiometer OP150K berfungsi sebagai sensor posisi. Potensiometer jenis ini merupakan potensiometer jenis rotasi (*rotary*) yang dapat diputar 360 derajat.

e. Tachogenerator

Tachogenerator GT150X berfungsi untuk mengkonversi besaran mekanik menjadi besaran listrik sehingga putaran motor dapat diukur.

B. Model Matematik Motor DC

Tegangan (V) untuk sistem motor DC seperti pada gambar 2 adalah masukan yang dipasang pada armatur (R, L, dan θ) motor. Tegangan tersebut oleh motor diubah menjadi gerak putar yang merupakan keluaran sistem sebagai kecepatan rotasi dari rotor motor.



Gambar 2. Rangkaian motor DC

Persamaan diferensial rangkaian kumparan magnet adalah

$$L \frac{di}{dt} + Ri = V - K\dot{\theta} \tag{1}$$

Dengan merubah persamaan diatas serta mengeliminasi I(s) maka didapatkan fungsi alih, dimana posisi sudut sebagai keluaran dan tegangan sebagai masukan.

$$\frac{\theta}{V} = \frac{K}{s((Js+b)(Ls+R) + K^2)} \tag{2}$$

dimana:

θ(s) = posisi motor (rad)

V(s) = tegangan masukan servo (V)

B = koefisien gesekan viskos ekuivalen dari motor dan beban pada poros motor, (N.m/rad.dt)

K = konstanta torsi motor

R = tahanan kumparan jangkar (Ω)

J = momen inersia ekuivalen dari motor (kg.m²)

L = induktansi kumparan jangkar (H)

Dalam bentuk persamaan ruang keadaan, persamaan diatas dapat dituliskan dengan memilih posisi motor, kecepatan rotasi motor, sebagai variabel *state* dan tegangan sebagai masukan serta posisi motor sebagai keluaran.

$$\frac{d}{dt} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{b}{J} \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} V \tag{3}$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} \tag{4}$$

C. Sistem Pengendalian Cascade

Sistem kontrol *cascade* adalah salah satu jenis strategi kontrol yang digunakan untuk mengoptimasi suatu sistem kontrol. Ciri khas sistem pengendalian *cascade* (bertingkat) adalah adanya *manipulated variable* (variabel yang dimanipulasi) sebuah pengendali yang menjadi *set point* dari pengendali lain. ada dua jalur umpan balik pada sistem pengendalian *cascade*, sehingga terbentuk dua mata rantai pengendalian. Mata rantai bagian luar (*outer loop*) disebut *primary loop*, dan mata rantai bagian dalam (*inner loop*) disebut *secondary loop*. *Primary loop* mengendalikan proses variabel primer sedangkan *slave* atau *secondary loop* mengendalikan proses variabel sekunder. keefektifan sistem kontrol *cascade* ini karena fakta bahwa gangguan yang mempengaruhi loop sekunder secara efektif dikompensasi sebelum mempengaruhi output pada proses utama[6].

D. Sistem Active Fault Tolerant Control (AFTC)

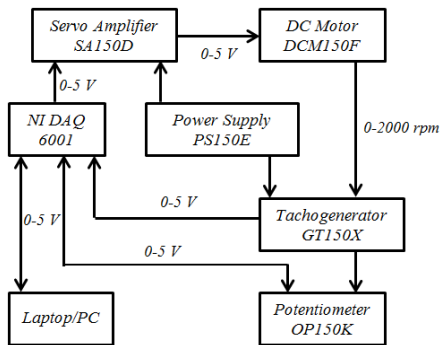
Sistem AFTC bereaksi terhadap kesalahan yang terjadi pada suatu komponen seperti sensor dan aktuator dengan merekonfigurasi aksi pengendali sehingga kestabilan dan kinerja pada sistem dapat dijaga. Dengan demikian, tujuan utama dari sistem FTC adalah merancang pengendali dengan struktur yang cocok untuk mencapai kestabilan dan kinerja yang diinginkan. Tidak hanya ketika komponen pengendali berfungsi secara normal tetapi ketika terjadi kesalahan pada suatu komponen tersebut.

Active Fault Tolerance Control (AFTC) secara umum :

- a. *Fault Detection Diagnosis (FDD) atau Fault Detection and Identification (FDI)*
- b. *Reconfigurable Control (RC)*

III. METODOLOGI PENELITIAN

A. *Set Eksperimen Sistem Servo Modular MS150 DC*



Gambar 3. Diagram Blok Sistem Pengendalian Posisi Motor DC

Pengendalian posisi dapat dilakukan dengan mengatur sinyal masukan tegangan pada motor DC, karena posisi motor dipengaruhi oleh besarnya tegangan masukan yang diberikan. Perintah dari PC diberikan kepada DAQ sebagai sinyal kendali untuk servo dimana tegangan keluaran DAQ diatur dengan rentang 0-5 volt. Tegangan terminal yang berupa sinyal kendali merupakan masukan bagi *servoamplifier*. Masukan tersebut akan menggerakkan motor DC, dimana tegangan terminal akan proporsional dengan kecepatan. Hasil pergerakan dari motor DC akan dikopel dengan *tachogenerator*. Pergerakan pada bagian kecepatan rendah akan dibaca oleh potensiometer yang kemudian dikonversi ke besaran tegangan dan masuk dalam PC/laptop.

B. *Pemodelan Sistem Servo Modular MS150 DC*

Pemodelan dilakukan dengan pemilihan model berupa persamaan fungsi transfer. Pemilihan fungsi transfer model harus mendekati grafik pengukuran dari *real plant*. Setelah dilakukan pemodelan dengan menggunakan perangkat lunak *labVIEW* 2013, diperoleh nilai fungsi transfer dalam domain *z*. Persamaan (5) kemudian dirubah kedalam persamaan dengan domain *k* yang selanjutnya dirubah ke dalam bentuk persamaan ruang keadaan (*state space*), yang dituliskan :

- a. Kecepatan

$$\omega(z) = \frac{0.91}{z - 0.5948} u(z) \tag{5}$$

persamaan (8) di transformasi balik, menjadi persamaan

$$\omega(k + 1) = 0.5948\omega(k) + 0.91u(k) \tag{6}$$

- b. Posisi

$$\frac{\theta(z)}{\omega(z)} = \frac{K_x T_s}{z - 1} \tag{7}$$

kemudian persamaan (12) diinvers *Z* menjadi persamaan

$$\theta(k + 1) = \theta(k) + K_x T_s \omega(k) \tag{8}$$

diketahui $K_x = 0,6$ dan $T_s = 0,1$ detik, maka didapatkan,

$$\theta(k + 1) = \theta(k) + 0.06\omega(k) \tag{9}$$

Selanjutnya diperoleh bentuk persamaan ruang keadaan untuk sistem servo modular MS150 DC dalam domain *k*,

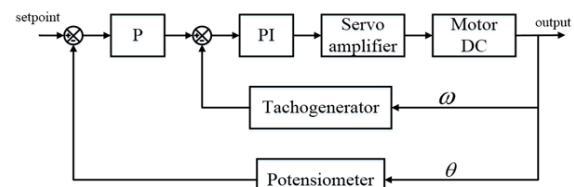
$$\begin{bmatrix} \theta(k + 1) \\ \omega(k + 1) \end{bmatrix} = \begin{bmatrix} 1 & 0,06 \\ 0 & 0,5948 \end{bmatrix} \begin{bmatrix} \theta(k) \\ \omega(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0,91 \end{bmatrix} u(k) \tag{10}$$

$$y(k) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \theta(k) \\ \omega(k) \end{bmatrix} \tag{11}$$

Validasi kemudian dilakukan dengan cara membandingkan respon antara sistem riil dengan model fungsi transfer dari sistem. Batas *error* validasi maksimal yang digunakan sebesar 5%.

C. *Perancangan Sistem Kendali Cascade PI-P*

Nilai K_p dan K_i didapatkan secara eksperimen dengan metode *trial and error* untuk perancangan secara simulasi dengan *simulink* pada perangkat lunak *matlab* R2103a, kemudian setelah sistem kendali dapat bekerja dengan baik, dilakukan perancangan sistem kendali *cascade* P-PI untuk diterapkan secara *realtime* pada sistem servo modular MS150 DC dengan menggunakan perangkat lunak *labVIEW* 2013. Nilai *gain* K_p dan K_i untuk simulasi terkadang tidak tepat untuk digunakan secara *real time* jika perangkat lunak yang digunakan berbeda, sehingga diperlukan perubahan nilai *gain* yang sesuai untuk diterapkan secara *realtime*.



Gambar 4. Diagram pengendalian cascade PI-P

D. *Perancangan Sistem Active Fault Tolerant Control (AFTC)*

- Perancangan *Observer*

Perancangan *observer* digunakan untuk mengestimasi kesalahan yang ada dengan cara mengembangkan persamaan ruang keadaan sistem. Nilai matriks ruang keadaan untuk perancangan *observer* diperoleh dari pemodelan sistem servo modular MS150 DC pada persamaan 3.6 dan 3.7 adalah:

$$A = \begin{bmatrix} 1 & 0,06 \\ 0 & 0,5948 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0,91 \end{bmatrix} \quad F_a = \begin{bmatrix} 0 \\ 0,91 \end{bmatrix}$$

Persamaan ruang keadaan yang mengandung kesalahan aktuator dituliskan sebagai berikut :

$$x(k+1) = Ax(k) + Bu(k) + F_a f_a(k) \quad (12)$$

$$y(k) = Cx(k) \quad (13)$$

dengan

$$x(t) \in R^n, u(t) \in R^m, y(t) \in R^p, d(t) \in R^n, \omega(t) \in R^q, f_a(t) \in R^r, f_s(t) \in R^q$$

berturut-turut adalah keadaan (*state*), masukan, keluaran, gangguan, *noise*, kesalahan aktuator dan kesalahan sensor. Sedangkan A, B, C, dan F_a adalah matriks keadaan, masukan, keluaran, dan kesalahan aktuator.

Kemudian *state* baru z_{k+1} ditambahkan untuk memindahkan kesalahan sensor dari persamaan keluaran ke persamaan keadaan sehingga diperoleh persamaan :

$$z(k+1) = -A_z T_s z(k) + A_z T_s Cx(k) + z(k) \quad (14)$$

persamaan dirubah ke dalam bentuk persamaan yang lebih ringkas menjadi

$$\begin{bmatrix} x(k+1) \\ z(k+1) \end{bmatrix} = \begin{bmatrix} A & 0 \\ A_z C & -A_z T_s + I \end{bmatrix} \begin{bmatrix} x(k) \\ z(k) \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u(k) + \begin{bmatrix} F_a & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} f_a(k) \\ f_s(k) \end{bmatrix} \quad (15)$$

$$y(k) = \begin{bmatrix} 0 & I \end{bmatrix} \begin{bmatrix} x(k) \\ z(k) \end{bmatrix} \quad (16)$$

dengan nilai :

$$A_z = 1000, T_s = 0.1$$

Persamaan keadaan dan persamaan keluaran yang mengandung kesalahan aktuator dan sensor digunakan untuk merancang persamaan *observer* yang dituliskan:

$$\hat{x}(k+1) = A.\hat{x} + B.u(k) + F_a.\hat{f}_a(k) + K_e(y(k) - \hat{y}(k)) \quad (17)$$

$$\hat{y}(k) = C\hat{x}(k) \quad (18)$$

Kemudian ditambahkan *state* yang mengandung gain untuk mengestimasi kesalahan aktuator dan sensor (L),

$$\hat{f}_a(k+1) = T_s L_e(k) + \hat{f}_a(k) \quad (19)$$

sehingga dari persamaan diatas jika dirubah kedalam bentuk yang lebih ringkas, menjadi

$$\begin{bmatrix} \hat{x}(k+1) \\ \hat{f}_a(k+1) \end{bmatrix} = \begin{bmatrix} A & F_a \\ 0 & I \end{bmatrix} \begin{bmatrix} \hat{x}(k) \\ \hat{f}_a(k) \end{bmatrix} + \begin{bmatrix} \tilde{B} \\ 0 \end{bmatrix} u(k) + \begin{bmatrix} K_e \\ L_e T_s \end{bmatrix} e(k) \quad (20)$$

$$\hat{y}(k) = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} \hat{x}(k) \\ \hat{f}_a(k) \end{bmatrix} \quad (21)$$

dari bentuk (19) dan (20) dapat dirubah menjadi persamaan keadaan dan persamaan keluaran *observer*, sebagai berikut :

$$\hat{\tilde{x}}(k+1) = \tilde{A}.\hat{\tilde{x}}(k) + \tilde{B}.u(k) + \tilde{K}_e(\tilde{y}(k) - \hat{\tilde{y}}(k)) \quad (21)$$

$$\hat{\tilde{y}}(k) = \tilde{C}.\hat{\tilde{x}}(k) \quad (22)$$

dengan substitusi nilai matriks A, B, C, F, I, Ts, diperoleh nilai matriks sebagai berikut :

$$\tilde{A} = \begin{bmatrix} 1 & 0.06 & 0 \\ 0 & 0.5948 & 0.91 \\ 0 & 0 & 1 \end{bmatrix} \quad \tilde{C} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\tilde{B} = \begin{bmatrix} 0 \\ 0.91 \\ 0 \end{bmatrix}$$

Untuk memperoleh nilai gain observer (\tilde{K}_e) digunakan metode *pole-placement*. Nilai pole yang diinginkan ditentukan dengan metode *trial-and-error*, dengan berdasarkan ilmu kestabilan berdasarkan letak *pole*, diketahui bahwa sebuah sistem diskrit didefinisikan stabil jika seluruh akar-akar karakteristik bagian riil atau imajiner bernilai tidak lebih dari 1 atau -1. Sehingga dengan metode *trial-and-error* diperoleh nilai 3 pole yaitu 0,41; 0,41; 0,42.

Kemudian nilai matriks gain observer didapatkan dengan menggunakan program perangkat lunak *matlab* R2013a sehingga diperoleh matriks \tilde{K}_e sebagai berikut :

$$K_e = \begin{bmatrix} 0.59 & 0 & 0 \\ 0.06 & 0.7648 & 0.3760 \end{bmatrix}$$

Selanjutnya, estimasi kesalahan sensor diperoleh dengan persamaan berikut:

$$\hat{f}_s = \hat{\theta} - \theta \quad (23)$$

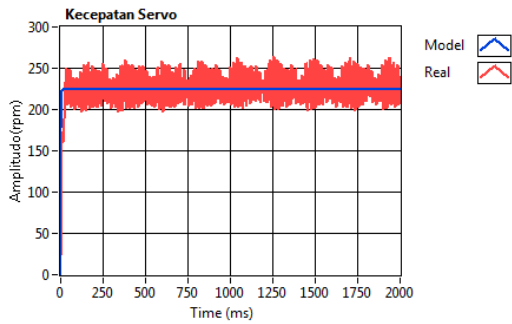
Setelah dilakukan perancangan *observer* secara simulasi menggunakan perangkat lunak *matlab* R2013a dan *observer* dapat mengestimasi kesalahan dengan benar, maka selanjutnya dilakukan penerapan *observer* secara *realtime* dengan menggunakan perangkat lunak *labVIEW* 2013.

IV. HASIL DAN PEMBAHASAN

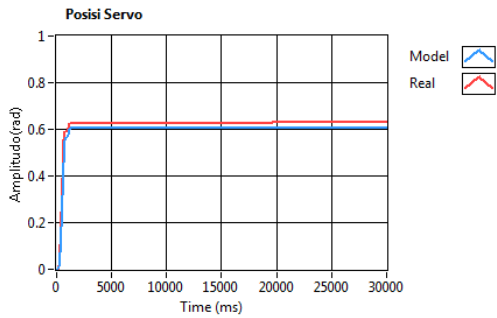
A. Validasi

Validasi dilakukan dengan membandingkan grafik model dan *real* dari kecepatan dan posisi sistem servo modular MS150 DC.

Gambar 7 menunjukkan nilai *error* validasi kecepatan sebesar 2,4%, sedangkan gambar 8 menunjukkan nilai *error* validasi posisi sebesar 3,7%. Grafik respon kecepatan dan posisi antara model dan sistem secara *realtime* sudah saling mendekati sehingga model yang diperoleh sudah dapat merepresentasikan sistem secara *realtime*.



Gambar 5. Respon sistem validasi kecepatan motor DC



Gambar 6. Respon sistem validasi posisi motor DC

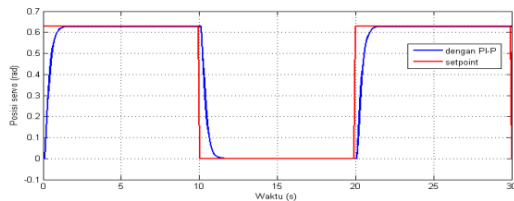
B. Sistem Kendali Cascade PI-P

Tahapan perancangan sistem kendali *Cascade* PI-P dilakukan untuk mendapatkan nilai *gain* pengendali yang selanjutnya hasil respon dapat diperoleh sehingga dapat diketahui kinerja dari sistem kendali yang dirancang.

1) Pengendalian Cascade PI-P secara Simulasi

Dengan menggunakan *simulink* pada perangkat lunak *Matlab* R2013a, diperoleh nilai pengendalian sebagai berikut :
 -*Inner loop*: $K_p = 1, K_i = 4$
 -*Outer loop*: $K_p = 4$

Grafik respon sistem kendali PI-P pada motor DC secara simulasi ditunjukkan pada gambar 7.



Gambar 7. Respon sistem perancangan sistem kendali PI-P secara simulasi

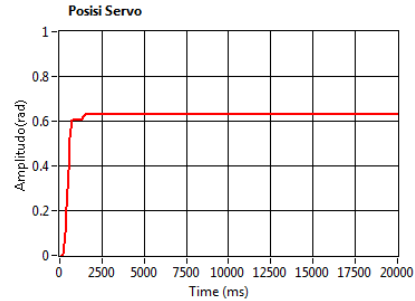
Gambar 7 menunjukkan 2 grafik, yaitu grafik berwarna merah yang menunjukkan nilai *set point* dan grafik berwarna biru yang menunjukkan respon hasil pengendalian *cascade*. Nilai *set point* sebesar 0,628 radian diberikan sehingga dari kedua grafik tersebut dapat diketahui bahwa sistem dengan PI-P yang dirancang dapat bekerja dengan baik karena mampu mencapai nilai *set point* dengan karakteristik respon nilai *maximum overshoot* sebesar 0% , *settling time* sebesar 2,1 detik dan *error steady state* sebesar 0%.

2) Pengendalian Cascade PI-P Secara Real

Untuk melihat respon sistem pengendalian posisi motor DC secara langsung (*real time*) maka dilakukan pengendalian dalam perangkat lunak *LabVIEW* 2013. Nilai pengendali yang didapat secara eksperimental adalah

- Inner loop*: $K_p = 0,7, \tau_i = 0,001$
- Outer loop*: $K_p = 0,34$

Grafik respon sistem posisi motor DC setelah diberi pengendalian *cascade* secara *real time* ditunjukkan pada gambar 10 berikut



Gambar 8. Respon sistem penerapan sistem kendali PI-P secara *realtime*

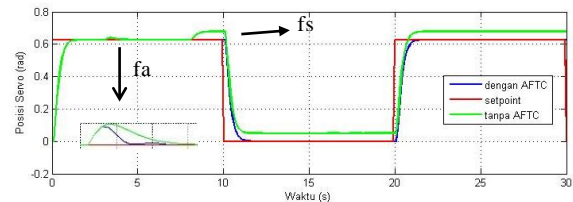
Gambar 8 menunjukkan grafik respon sistem secara *real* untuk kendali PI-P dengan *set point* yang diberikan sebesar 0,628 radian. Dari grafik tersebut dapat diketahui bahwa sistem kendali PI-P dapat bekerja dengan baik karena mampu mencapai nilai *set point* dengan *settling time* sebesar 2,2 detik serta tidak ada *maximum overshoot* dan *error steady-state* .

C. Uji Performansi

Uji performansi dilakukan untuk menguji algoritma *active fault tolerant control* yang telah dibuat dapat memperbaiki kesalahan yang terjadi pada aktuator dan sensor. Uji secara simulasi dilakukan dengan memberikan kesalahan pada aktuator dan sensor berupa kesalahan bias. Kesalahan bias secara simulasi pada aktuator diberikan sebesar 16%, sedangkan kesalahan bias pada sensor diberikan sebesar 8%.

1) Uji Kesalahan Bias Pada Aktuator dan Sensor Secara Simulasi

Hasil respon uji kesalahan bias pada aktuator sebesar 16% dan pada sensor sebesar 8% secara simulasi dapat dilihat pada gambar 9.



Gambar 9. Respon sistem AFTC dengan kesalahan bias 16% pada aktuator dan 8% pada sensor secara simulasi

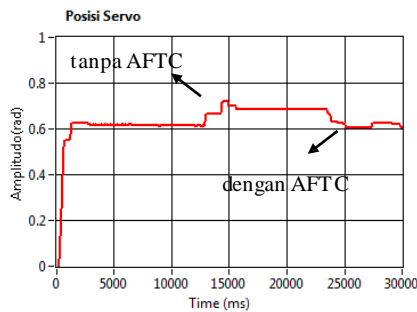
Tabel 1.

Performansi sistem dengan kesalahan bias pada aktuator 16% dan sensor 8% secara simulasi

Parameter	Kesalahan Aktuator		Kesalahan Sensor	
	Dengan AFTC	Tanpa AFTC	Dengan AFTC	Tanpa AFTC
Overshoot	1,27%	1,6%	0%	0%

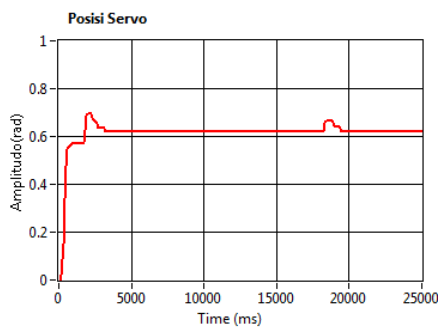
Error Steady State	0%	0%	0%	8 %
Settling Time	4,8 s	5 s	2,2s	-

2) Uji Kesalahan Bias Pada Aktuator dan Sensor Secara Real Time

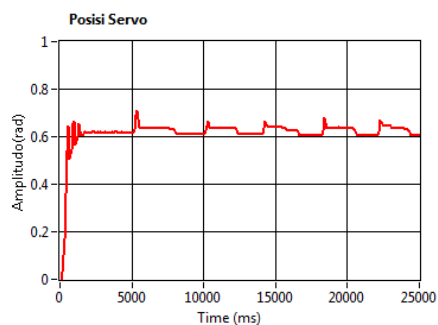


Gambar 10. Respon sistem hasil AFTC dengan kesalahan bias pada sensor 8% secara real time

Pada gambar 10 diketahui respon sistem dengan AFTC dan tanpa AFTC dengan nilai *setpoint* sebesar 0,628 radian. Ketika kesalahan bias pada sensor terjadi, sistem dengan AFTC dapat memperbaiki kesalahan sehingga dapat mencapai nilai *setpoint*. Namun pada sistem tanpa AFTC, sistem tidak dapat kembali ke nilai *setpoint* yang diinginkan dan mengalami *error steady state* sebesar 11%.



Gambar 11. Respon sistem dengan kesalahan bias pada aktuator sebesar 16% secara real time



Gambar 12. Respon sistem hasil AFTC dengan kesalahan bias pada aktuator sebesar 16% secara real time

Gambar 11 menunjukkan sistem tanpa AFTC sedangkan gambar 12 menunjukkan sistem hasil AFTC dengan nilai *setpoint* sebesar 0,628 radian. Ketika kesalahan bias pada aktuator terjadi, sistem tanpa AFTC memiliki nilai *overshoot* sebesar 12,7% sedangkan sistem dengan AFTC memiliki nilai *overshoot* lebih kecil sebesar 1,59%.

V. KESIMPULAN

Berdasarkan hasil analisis data yang telah dilakukan, didapatkan kesimpulan dari studi mengenai perancangan sistem *Active Fault Tolerant Control* pada pengendalian kecepatan sistem servo modular MS150 DC dengan kesalahan pada aktuator dan sensor adalah sebagai berikut :

- a. Telah dilakukan perancangan algoritma AFTC berupa perancangan *observer* dan *reconfigurable control* dengan kesalahan aktuator dan sensor secara simulasi serta secara *real time* pada pengendalian posisi sistem servo modular MS150 DC sehingga sistem mampu memperbaiki kesalahan bias pada aktuator dan sensor dengan sistem dapat kembali mencapai nilai *setpoint* dibandingkan dengan tanpa AFTC.
- b. Sistem tanpa AFTC pada kesalahan sensor secara simulasi mengalami *error steady state* sebesar 8% sedangkan sistem dengan AFTC tidak mengalami *error steady state*. Sistem tanpa AFTC pada kesalahan aktuator secara simulasi mengalami *overshoot* sebesar 1,6% sedangkan sistem dengan AFTC mengalami *overshoot* lebih kecil sebesar 1,27%.
- c. Sistem tanpa AFTC pada kesalahan sensor secara *real time* mengalami *error steady state* sebesar 11% sedangkan sistem dengan AFTC tidak mengalami *error steady state* . Sistem tanpa AFTC pada kesalahan aktuator secara *real time* mengalami *overshoot* sebesar 12,7%, sedangkan sistem dengan AFTC mengalami *overshoot* lebih kecil sebesar 1,59%.

DAFTAR PUSTAKA

- [1] T. Wildi, *Electrical Machines, Drives, and Power Systems*. 2002.
- [2] M. Namazov, "DC motor position control using fuzzy proportional-derivative controllers with different defuzzification methods," *Off. J. Turkish Fuzzy Syst. Assoc.*, vol. 1, no. 1, pp. 36–54, 2010.
- [3] D. Arvani, F. "Digital Control of MS-150 Modular Position Servo System," *IEEE NECEC*, 2007.
- [4] Y. Z. and J. Jiang, "Bibliographical Review on Reconfigurable Fault-Tolerant Control Systems," *Annu. Rev. Control*, vol. 32, no. 2, pp. 229–252, 2008.
- [5] I. Akbar, "Perancangan Sistem Active Fault Tolerant Control (AFTC) Pada Pengendalian Posisi Sistem Servo Modular MS150 DC dengan Kesalahan Pada Aktuator Dan Sensor," *ITS*, 2017.
- [6] A. Visioli, "Practical PID Control," *Springer*, 2006.