

IMPLEMENTASI MICROKONTROLLER UNTUK SISTEM KENDALI KECEPATAN BRUSHLESS DC MOTOR MENGGUNAKAN ALGORITMA HYBRID PID FUZZY

Roedy Kristiyono^{1*}, Oyas Wahyunggoro^{2*}, Prapto Nugroho^{3*}

¹Jurusan Teknik Elektro, Akademi Teknologi Warga Surakarta
roedy_kristiyono@atw.ac.id

J. Raya Solo-Baki Km.2, Kwarasan Solo Baru, Sukoharjo

²Jurusan Teknik Elektro dan Teknologi Informatika, UGM Yogyakarta
oyas@ugm.ac.id

³Jurusan Teknik Elektro dan Teknologi Informasi, UGM Yogyakarta
pnugroho@jteti.gadjahmada.edu

Jl. Grafika No.2 Kampus UGM, Yogyakarta 552813

Abstrak

Motor BLDC secara luas digunakan pada banyak aplikasi industri karena mempunyai efisiensi tinggi, torsi tinggi dan volume noise yang rendah. Pengendalian kecepatan Motor BLDC merupakan suatu proses yang rumit. Tetapi kerumitan yang dilakukan sebanding dengan unjuk kerja dari Motor BLDC yang tinggi. Kendali PID konvensional terbukti mampu menunjukkan kinerja yang baik pada pengendalian dengan pembebanan tertentu pada plant. Tetapi setiap terjadi perubahan beban pada plant, kendali PID konvensional harus dilakukan set ulang parameter k_p , k_i dan k_d untuk dapat menuju ke keadaan steady state sesuai dengan set point yang diinginkan. Tujuan penelitian ini adalah merancang peralatan kendali untuk Motor BLDC yang dapat menala parameter PID secara otomatis oleh logika fuzzy. Dalam penelitian digunakan mikrokontroler AVR RISC sebagai pusat kendali. Sedangkan perangkat lunak digunakan untuk algoritma pemrograman kendali PID dan kendali hybrid PID fuzzy dengan bahasa C. Agar dapat menala parameter PID yang tepat secara waktu nyata, maka dibuat sistem kendali dua aras. Aras pertama menentukan parameter PID dengan mencari batas minimal dan maksimal nilai k_p , k_i dan k_d dengan metode kurva reaksi. Aras kedua merancang sistem fuzzy agar dapat menala penguatan PID secara otomatis, kemudian memformulasikan kedalam kombinasi 49 aturan if-then fuzzy untuk mendapatkan nilai k_p , k_i dan k_d yang tepat dari perubahan nilai error dan delta error. Pengujian perubahan set point dan perubahan beban dihasilkan karakteristik respon sistem kendali PID konvensional dengan nilai rata-rata yaitu waktu kenaikan (t_r) 0.025 detik, waktu penetapan (t_s) 0.1625 detik, overshoot sebesar 15.98%. Sedangkan kendali Hybrid PID Fuzzy dihasilkan nilai rata-rata waktu kenaikan (t_r) 0.0025 detik, waktu penetapan (t_s) 0.057 detik, overshoot sebesar 5.42%.

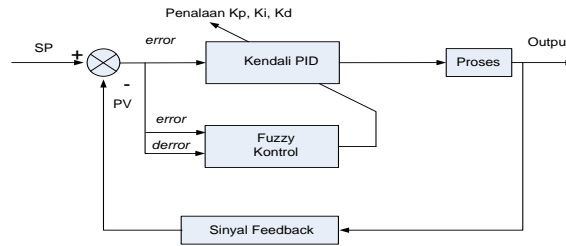
Kata kunci : Auto-tuning, Brushless DC Motor, Fuzzy, Pengendalian, PID

I. PENDAHULUAN

Sistem kendali PID paling banyak digunakan dalam pengendalian di industri. Keberhasilan pengendali PID tergantung ketepatan dalam menentukan konstanta (penguatan) PID (Gunterus, Frans. 1994). Dalam suatu sistem kendali, kita mengenal adanya beberapa macam aksi kontrol, yaitu aksi kontrol *proporsional*, aksi kontrol *integral*, dan aksi kontrol *derivatif*. Masing-masing aksi kontrol ini mempunyai keunggulan tertentu, dimana aksi kontrol *proporsional* mempunyai keunggulan *risetime* yang cepat, aksi kontrol *integral* mempunyai keunggulan untuk memperkecil *error*, dan aksi kontrol *derivatif* keunggulannya untuk memperkecil *derror* atau meredam *overshoot/undershoot*. Agar mendapatkan suatu keluaran yang tinggi dan *error* yang kecil, maka kita dapat menggabungkan kontrol-kontrol tersebut menjadi aksi PID. Pada makalah ini sistem kendali yang digunakan adalah sistem kendali PID digital.

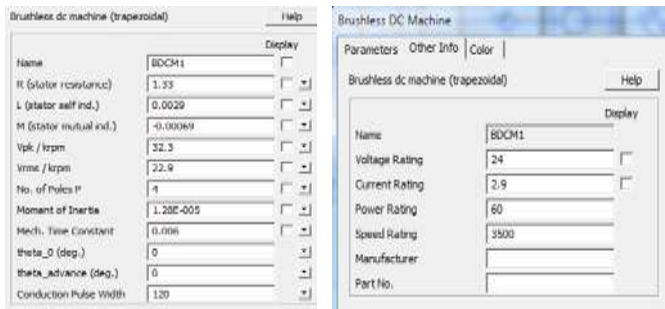
Seiring dengan perkembangan sistem berbasis pengetahuan, penalaan konstanta PID ini dapat ditentukan dengan menganalisis tanggapan suatu sistem, hasil analisis ini dibentuk dalam sejumlah aturan. Dengan mengkombinasikan aturan, pengaturan ini menjadi sebuah sistem *fuzzy* sebagai salah satu sistem berbasis pengetahuan sehingga konstanta PID dapat ditala secara waktu nyata. Untuk mencapai hal ini sistem kendali dibentuk menjadi sistem kendali dua aras (Wang, L. X. 1997). Aras pertama adalah sistem kendali PID konvensional. Aras kedua adalah sistem *fuzzy* yang menala konstanta PID secara waktu nyata.

2. METODOLOGI

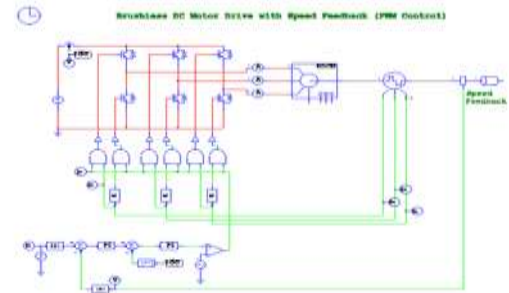


Gambar 1. Kendali PID tuning Fuzzy

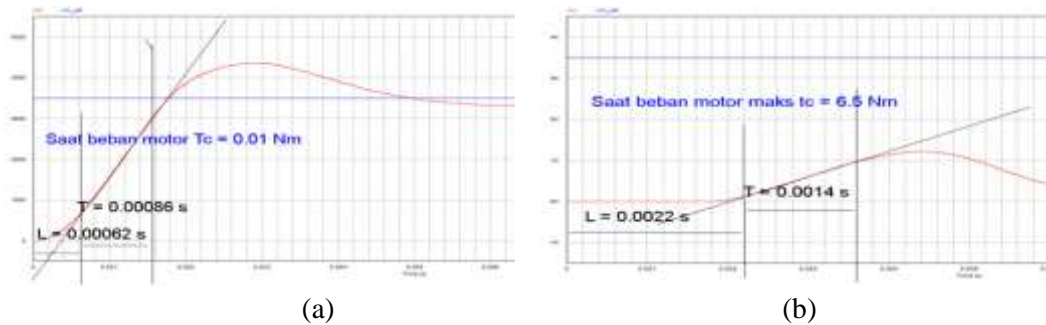
Langkah awal yang dilakukan adalah mempelajari *plant* yang akan digunakan dalam sistem kendali ini. Pada gbr.2, merupakan spesifikasi Motor BLDC yang digunakan sebagai *plant*, dengan daya 60 Watt, tegangan 24 Volt, arus 2.9 Ampere, 3500 rpm, serta beberapa data parameter motor diperoleh melalui datasheet *plant* yang digunakan. Uji rangkaian percobaan dengan menggunakan program simulator PSIM, pada motor BLDC dilakukan percobaan pembebanan untuk mendapatkan hasil *dead time* dan *delay time* dengan metode kurva proses reaksi. Dimana dari hasil kurva reaksi ini nanti akan diperoleh nilai-nilai k_p , k_i dan k_d maksimal dan minimal dengan metode Ziegler Nichols.



Gbr 2. Data Parameter Motor



Gbr 3. Rangkaian Percobaan Simulator PSIM



Gambar 4. Motor BLDC dengan pembebanan minimal dan maksimal

Untuk mendapatkan harga K_p , $T_i = 1/K_i$, T_d (K_d) maka ditentukan dengan kurva proses reaksi (Gbr.4.), dimana sistem dijalankan secara open loop (Y. S. Lai, 1999). Perhitungan *auto tuning* PID menurut Ziegler Nichols dapat dicari dengan persamaan :

$$K_p = 1,2 \frac{T}{L} \tag{1}$$

$$K_i = K_p \cdot \frac{1}{T_i} = K_p \cdot 2L \tag{2}$$

$$K_d = K_p \cdot T_d = K_p \cdot \frac{L}{2} \tag{3}$$

Dengan hasil pada gbr 4(a), dan 4(b), dimasukan pada persamaan :

$$K'_p = \left[\frac{K_p - K_{p \min}}{K_{p \max} - K_{p \min}} \right] \quad (4)$$

$$K'_d = \left[\frac{K_d - K_{d \min}}{K_{d \max} - K_{d \min}} \right] \quad (5)$$

$$K_i = \left[\frac{K_p}{\alpha \cdot T_d} \right] \quad (6)$$

Sehingga diperoleh persamaan :

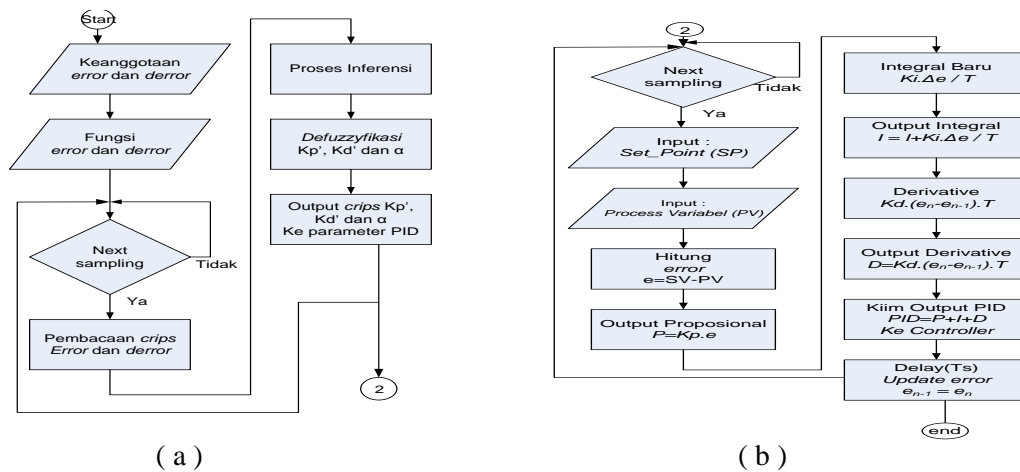
$$K_p = K'_p \cdot (0.368) + 1.1316 \quad (7)$$

$$K_d = K'_d (0.00048) + 0.00035 \quad (8)$$

$$K_i = \left[\frac{1.1316}{\alpha \cdot (0.00031)} \right] \quad (9)$$

Dari persamaan diatas nilai K_p , K_d dan K_i akan berubah-ubah sesuai dengan pembacaan hasil defuzzyfikasi ketiga parameter K'_p , K'_d dan α .

Sedangkan pembacaan nilai K'_p , K'_d dan α adalah seperti diagram alir gbr 5a. Sehingga dapat direalisasikan pada rutin *fuzzy* kedalam bahasa pemrograman mikrokontroler. Tahap pertama adalah membentuk definisi keanggotaan *error* dan *derror* kedalam struktur *array*, kemudian membentuk fungsi keanggotaan tiap variabel keanggotaan *error* dan *derror*.



Gambar 5. Pembacaan Nilai Kontanta K'_p , K'_d dan α dan implementasi program ke mikrokontroler

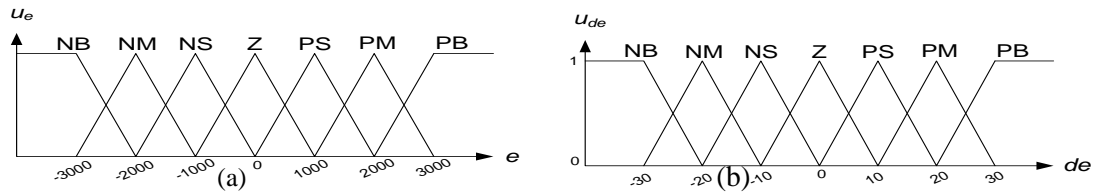
Dari diagram alir gbr 5.a, terjadi eksekusi program di setiap waktu *sampling*, maka besar nilai *crisp error* dan *derror* akan dimasukkan kedalam fungsi keanggotaannya. Hasil dari pembacaan fungsi keanggotaan dilakukan proses *inferensi*, selanjutnya dilakukan proses defuzzyfikasi untuk menentukan nilai K'_p , K'_d dan α . Nilai hasil defuzzyfikasi merupakan nilai tegas yang nantinya digunakan oleh kendali PID sebagai variabel penalaannya.

Dari diagram alir gbr.5.b dapat direalisasi kedalam bahasa pemrograman mikrokontroler sebagai berikut : Ketika terjadi periode waktu *sampling*, sistem akan menunggu apakah ada penekanan *keypad* jika ada, kemudian memasukkan nilai *set_point* (SP) melalui penekanan *keypad*. Kemudian besaran *variabel process* (PV) akan selalu dibaca nilainya dan digunakan sebagai nilai pengurang terhadap SP untuk mendapatkan harga *error*. Untuk menghasilkan keluaran *proporsional* maka nilai *error* dikalikan dengan penguatan *proporsional*. Sedangkan harga keluaran *integral* merupakan besarnya akumulasi *error* dikalikan dengan penguatan *integral*. Dan harga keluaran *derivatif* adalah harga perubahan *error* dikalikan dengan penguatan *derivatif*.

Setelah ketiga parameter PID didapatkan kemudian akan diumpankan ke kontroler. Proses ini terjadi selama satu perioda *sampling* sebesar waktu tunda (*ts*). Selanjutnya harga *error* selama satu perioda *sampling* perlu diperbarui untuk perhitungan nilai *derivatif* berikutnya.

Nilai *error* dibagi kedalam tujuh aras (NB, NM, NS, Z, PS, PM, PB), sedangkan nilai perubahan *derror* juga dibagi kedalam tujuh aras (DNB, DNM, DNS, DZ, DPS, DPM, DPB). Huruf pertama N, P dan D berarti *negative*, *positive* dan *delta*, sedangkan huruf kedua B, M, S dan Z berarti *big*, *medium*, *small* dan *zero*.

Pada penelitian menggunakan fungsi bentuk segitiga. Bentuk fungsi segitiga lebih mudah diterapkan dalam pembuatan program. Didalam rutin program *fuzzy* nilai *set point* dikalikan dengan 10 dikurangi nilai dari kecepatan motor melalui masukan ADC 10 bit. Nilai masukan ADC digunakan sebagai nilai pengurang terhadap *set point* untuk menghasilkan besarnya *error*. Rentang kecepatan motor yang dapat dibaca sebesar 100 sampai 3000. Nilai *error* yang terbaca dapat dipetakan menurut fungsi keanggotaan input *derror* dan perubahan *derror* seperti gbr 6.



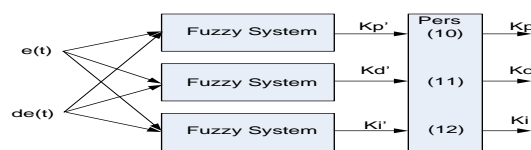
Gambar 6 (a) Keanggotaan input *error* (b)Keanggotaan Input *derror*

$$K'_p = \frac{\sum_{i=1}^{49} Y_p \cdot U_A(e(t)) \cdot U_B(d_e(t))}{\sum_{i=1}^{49} U_A(e(t)) \cdot U_B(d_e(t))} \tag{10}$$

$$K'_d = \frac{\sum_{i=1}^{49} Y_d \cdot U_A(e(t)) \cdot U_B(d_e(t))}{\sum_{i=1}^{49} U_A(e(t)) \cdot U_B(d_e(t))} \tag{11}$$

$$K'_\alpha = \frac{\sum_{i=1}^{49} Y_\alpha \cdot U_A(e(t)) \cdot U_B(d_e(t))}{\sum_{i=1}^{49} U_A(e(t)) \cdot U_B(d_e(t))} \tag{12}$$

Dari aturan pada gbr 7, maka dapat ditentukan aturan ketiga konstanta PID. Selengkapnya sesuai persamaan (10), (11), dan (12) pernyataan fuzzy tersebut dapat ditabelkan pada tabel 1, tabel 2 dan tabel 3, yang kemudian akan di masukkan sebagai data pemrograman mikrokontroler yang digunakan system kendali hybrid PID-Fuzzy untuk mengatur BLDC motor



Gbr 7. Fuzzy sistem menala kendali PID

Tabel 1 Aturan untuk K'p

	DNB	DNM	DNS	DZ	DPS	DPM	DPB
NB	B	B	B	B	B	B	B
NM	S	B	B	B	B	B	S
NS	S	S	B	B	B	S	S
Z	S	S	S	B	S	S	S
PS	S	S	B	B	B	S	S
PM	S	B	B	B	B	B	S
PB	B	B	B	B	B	B	B

Tabel 2 Aturan untuk K'd

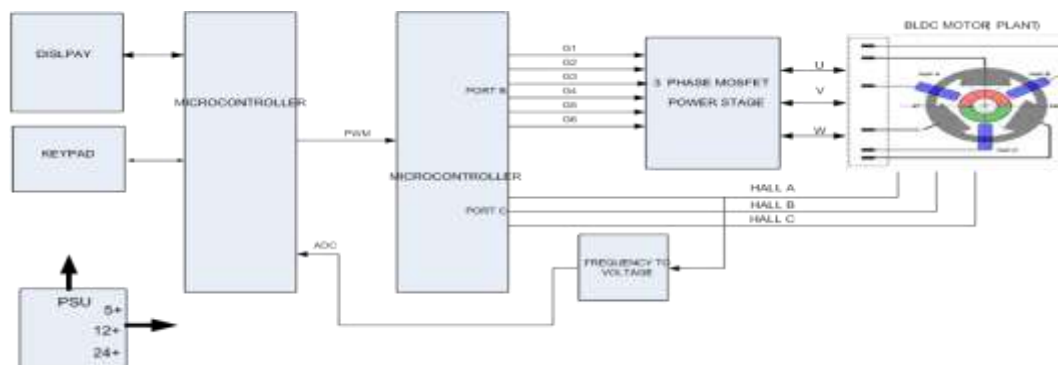
	DNB	DNM	DNS	DZ	DPS	DPM	DPB
NB	S	S	S	S	S	S	S
NM	B	B	S	S	S	B	B
NS	B	B	B	S	B	B	B
Z	B	B	S	S	S	B	B
PS	B	B	B	S	B	B	B
PM	B	B	S	S	S	B	B
PB	S	S	S	S	S	S	S

Tabel 3 Aturan untuk α

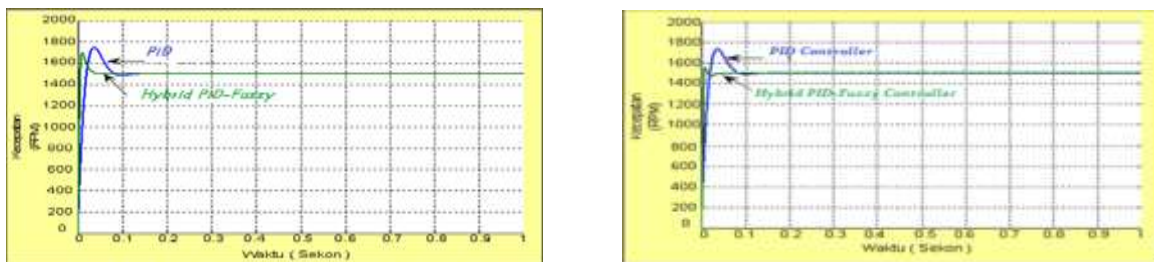
	DNB	DNM	DNS	DZ	DPS	DPM	DPB
NB	2	2	2	2	2	2	2
NM	3	3	2	2	2	3	3
NS	4	4	3	2	3	3	4
Z	5	4	3	3	3	4	5
PS	4	3	3	2	3	3	4
PM	3	3	2	2	2	3	3
PB	2	2	2	2	2	2	2

3. HASIL DAN PEMBAHASAN

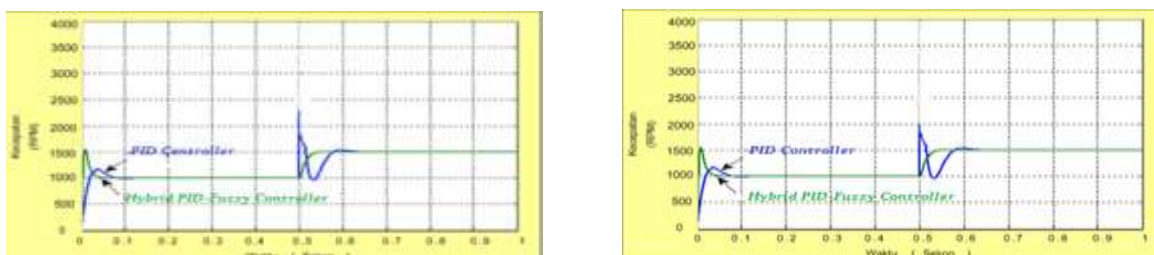
Pada gambar 8, diagram blok konstruksi hardware penelitian yang dilakukan, dan gambar 9, 10, 11, merupakan hasil dari penelitian yang dilakukan.



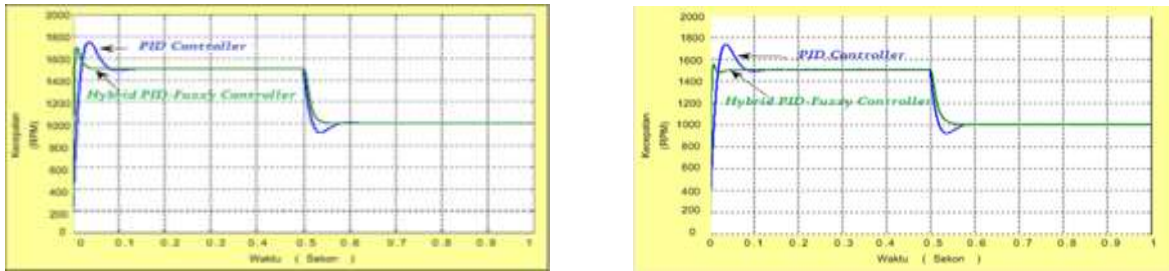
Gambar 8 Diagram Blok Konstruksi Hardware Penelitian



Gambar 9 Plant BLDC pada 1500 rpm tanpa beban dan dengan pembebanan



Gambar 10 Plant BLDC dari 1000rpm dinaikan ke 1500rpm tanpa beban dan dengan beban



Gambar 11 Plant BLDC dari 1500rpm turun ke 1000rpm tanpa beban dan dengan beban

Tabel 5 menunjukkan hasil uji percobaan yang dilakukan dengan mengubah-ubah set point (kecepatan). Juga dilakukan pula uji percobaan dengan mengubah set point secara tiba-tiba dalam selang waktu 0,5 detik dari set point 1000rpm menuju ke 1500 rpm tanpa beban dan dengan beban 0,5 kg, juga dilakukan uji perubahan set point dari 1500 rpm menuju 1000 rpm dengan seang waktu 0,5 detik.

Tabel 4. Data hasil uji percobaan

Kecepatan Set Point	Kontrol PID			Kontrol <i>hybrid PID-Fuzzy</i>		
	Rise Time (sec)	Over- shoot (%)	Settling Time (sec)	Rise Time (sec)	Over- shoot (%)	Settling Time (sec)
1500 tanpa beban	0.025	17.50	0.175	0.0025	17.00	0.05
1500 dengan beban	0.025	17.50	0.25	0.0025	15.50	0.075
1000 – 1500 tanpa beban	0.025	20.00	0.1375	0.0025	0	0.05
1000 – 1500 dengan beban	0.025	22.50	0.150	0.0025	0	0.044
1500 – 1000 tanpa beban	0.025	9.20	0.1875	0.0025	0	0.05
1500 – 1000 dengan beban	0.025	9.20	0.075	0.0025	0	0.075

4. KESIMPULAN

Dari hasil penelitian dan pembahasan, dapat disimpulkan bahwa proses kendali dengan sistem penalaan parameter kendali PID dengan logika *Fuzzy* yang diaplikasikan untuk mengatur Motor BLDC dapat memperbaiki kinerja kendali PID konvensional. Pengujian perubahan *set point* dan perubahan beban, dihasilkan karakteristik tanggapan sistem kendali PID konvensional dengan nilai rata-rata yaitu waktu kenaikan (t_r) 0.025 detik, waktu penetapan (t_s) 0.1625 detik, *overshoot* sebesar 15.98%. Sedangkan kendali *hybrid PID Fuzzy* dihasilkan nilai rata-rata waktu kenaikan (t_r) 0.0025 detik, waktu penetapan (t_s) 0.057 detik, *overshoot* sebesar 5.42%.

Dapat disimpulkan bahwa kendali *hybrid PID Fuzzy* mampu meningkatkan kinerja dari kendali PID konvensional.

UCAPAN TERIMAKASIH

Terimakasih kami sampaikan kepada Jurusan Teknik Elektro Akademi Teknologi Warga Surakarta yang telah memberi fasilitas dalam penelitian ini dan kepada Wiyono dan Budi Nugroho yang telah banyak membantu dan memberi dukungan.

REFERENSI

- Gunterus, Frans. 1994, Falsafah Dasar: Sistem Pengendalian Proses, Jakarta. PT. Elex Media Komputindo..
- Wang, L. X. 1997, A Course in Fuzzy Systems and Control, New Jersey: Prentice-Hall International. Inc: pp. 257-263.
- Y. S. Lai, Proceedings of the IEEE PES Winter Meeting, 1999, p.47