

Kendali Motor Sangkar Tupai 3 Phasa berbasis *Indirect Field Oriented Control (IFOC)* dengan *PID Controller*

ERA PURWANTO¹, ABDILLAH AZIZ MUNTASHIR², MUHAMMAD RIZANI
RUSLI³

^{1,3}Departemen Teknik Elektro, Politeknik Elektronika Negeri Surabaya, Indonesia

²Departemen Teknik Mekatronika, Politeknik Astra, Indonesia

Email: era@pens.ac.id

Received 4 Mei 2023 | Revised 24 Mei 2023 | Accepted 10 Juni 2023

ABSTRAK

Dunia industri sekarang banyak menggunakan motor sangkar tupai 3 phasa, karena memiliki kehandalan yang tinggi, tetapi motor sangkar tupai 3 phasa mempunyai beberapa kekurangan yaitu sukar untuk menjaga pada kecepatan konstan pada saat adanya perubahan beban, serta pengaturan kecepatan yang sulit dilakukan karena sistemnya yang coupled. Dengan melakukan metode vector control yang berbasis pada Indirect Field Oriented Control (IFOC), sehingga pengendalian motor sangkar tupai 3 phasa dapat dilakukan seperti pada DC motor penguat terpisah. PID controller digunakan untuk pengaturan kecepatan, dimana parameter PID didapatkan menggunakan Ziegler Nichols, serta hasil yang didapat adalah luaran kecepatan dengan risetime cepat dan error steady state kecil. Simulasi yang dilakukan telah dilakukan dengan menggunakan LabView, didapatkan respons waktu naik 1.132 ms, waktu penyelesaian 9,91 ms dan error steady 0,4% dengan set poin 500 Rpm. Serta waktu recovery sebesar 4.9 ms pada set poin 300 Rpm.

Kata kunci: *vector control, motor sangkar tupai 3 phasa, motor DC ,coupled, decoupled*

ABSTRACT

Three-phase squirrel cage motors are used today in many industries because of their high reliability. The downside is that speeds are difficult to adjust when connecting to a system. Three-phase cage motor control can be implemented as a discrete DC motor amplifier by using a vector control scheme based on Indirect Field Oriented Control (IFOC), for the speed control the PID controller is used in this system, PID parameters are obtained using Ziegler Nichols, and the result is an output speed, the fast rise time and low stability error. The simulation was run using LabView and had a time response 1.132 ms, a time steady state is 9.91 ms, and a error flat is 0.4% at a setpoint of 500rpm. So does the time recovery of 4.9ms in speed 300 rpm setpoint.

Keywords: *vector control, 3-phase squirrel cage, DC motor, coupled, decoupled*

1. PENDAHULUAN

Motor listrik adalah mesin listrik yang banyak digunakan di berbagai bidang seperti industri, bisnis, pelayanan publik serta perabotan listrik rumah tangga, berfungsi untuk mengoperasikan peralatan yang bervariasi termasuk blower angin, pompa air, kompresor serta perkakas mesin. Pada pengembangan industri Nasional, motor listrik memiliki peran tinggi pada total konsumsi daya nasional. Statistik menunjukkan bahwa motor listrik secara umum memaknai 2/3 konsumsi daya industri di setiap negara atau kurang lebih 40% konsumsi daya keseluruhan. Mesin induksi 3 fasa adalah motor yang masih dominan dipakai oleh dunia perindustrian, karena ada beberapa faktor yang menjadi pilihan utama karena mempunyai beberapa keunggulan antara lain: sederhana, kecil, murah, torsi awal besar, *range* torsi yang besar serta daya kontinu yang luas, memiliki respons torsi yang handal.

Motor sangkar tupai juga memiliki kelemahan yaitu rugi – ruginya besar, mempunyai *power factor* dan efisiensi yang rendah, di dalam penelitian kelemahan motor sangkar tupai yang harus diperhatikan adalah *power losses* atau rugi-rugi motor, untuk mengurangi kelemahan motor sangkar tupai tersebut maka harus dikembangkan untuk meningkatkan performa dari motor sangkar tupai **(Riba, dkk, 2016)**.

Menurut Angga Wahyu Aditya dan kawan-kawan yang dipublikasikan dalam *Jurnal Riset Saint Teknologi* 2019 yang melakukan evaluasi performa beberapa mesin listrik yang berfungsi sebagai *prime mover* adalah motor DC, PMSM maupun SRM, dari pembahasan tersebut dapat diketahui bahwa *induction motor* mempunyai keandalan yang lebih baik serta perawatannya murah, sehingga motor sangkar tupai menjadi pilihan utama bagi para penggunanya **(Aditya, dkk, 2019)**.

Motor sangkar tupai dapat diatur kecepatannya dengan menggunakan metode *scalar control* dan *vector control*. Sistem pengaturan dengan menggunakan metode *scalar control* lebih sederhana untuk penerapannya, kelemahan dari metode *scalar control* adalah adanya efek kopling yang mengakibatkan adanya pengaruh pada variabel yang lain bila variabel yang satu sedang dikontrol **(Iqbal, 2015)**, serta apabila dioperasikan pada kecepatan rendah motor sangkar tupai akan mempunyai drop tegangan yang lebih besar, sehingga mempengaruhi torsi, pada pengaturan dengan menggunakan metode *vector control* maka pengaturannya dapat dikontrol seperti mesin DC, yaitu fluksi serta torsi dapat diatur terpisah. Pengaturan fluksi, arus, tegangan, dapat meningkat performa yang baik begitu pula pada kecepatan rendah juga dapat menghasilkan performa yang baik. Metode *Vector control* dapat menghasilkan respons yang bagus pada berbagai kondisi, berbeda dengan *scalar control*, jika diimplementasikan pada kecepatan rendah dengan beban penuh, *scalar control* memiliki respons yang kurang baik **(Mukti, 2012)**. Selain itu, menurut Jisha L.K dan Powly Thomas yang dipublikasikan dalam *International Conference on Circuit, Control and Communications* 2013, bahwa metode *vector control* pada pengaturan motor sangkar tupai memiliki *tracking* kecepatan yang cepat dan *smooth* dibanding dengan *scalar control* **(Jisha & Thomas, 2013)**.

Pengaturan *kontrol berorientasi medan (FOC)* dan *kontrol torsi langsung (DTC)* adalah 2 macam metode dari *vector control*, metode FOC sendiri dikenal dengan *kontrol berorientasi medan (FOC)* dan *kontrol torsi langsung (DTC)*, perbedaan antara keduanya adalah, adanya sensor *hall effect* pada DFOC untuk pengukuran fluksi pada celah udara **(Hannan, dkk, 2018)**. Dengan adanya sensor tambahan pada metode DFOC menambah kesulitan dan biaya, karena pemasangan sensor tersebut pada celah udara motor, pada kecepatan rendah maka hasil pembacaannya tidak akurat, berbeda dengan metode IFOC metode ini mengembangkan medan motor sangkar tupai yang diatur, yaitu dengan mengubah motor sangkar tupai menjadi sistem *decupled*. Dengan ini motor sangkar tupai dapat diatur seperti motor DC, di mana arus

medan dan arus-arus torsi dapat diatur sendiri-sendiri, sehingga fluksi dan torsi juga dapat diatur secara sendiri-sendiri.

Untuk meningkatkan performa motor sangkar tupai digunakan beberapa skema teknik pengaturan atau pengendali. Pada penelitian ini digunakanlah PID sebagai *control* untuk perbaikan respons terhadap perubahan kecepatan yang ditentukan, untuk menghasilkan luaran yang konstan maka PID kontroler menjadi pilihan dengan kondisi kelebihan dan kekurangan dari parameter Kp, Ki dan Kd yang dapat saling memperkuat bila digabungkan menjadi satu dan dipasang sejajar. Dengan adanya PID *controller* maka luaran yang dihasilkan berupa *rise time* yang cepat, *error steady state* yang kecil serta tidak menghasilkan *overshoot* (Ferdiansyah, dkk, 2019). Metode *Ziegler Nicholz* 1 pada penelitian digunakan untuk menentukan parameter PID, karena lebih mudah untuk melakukan *tuning* dari parameter-parameter tersebut.

LabView digunakan untuk simulasi yang dilakukan dalam penelitian ini. LabView adalah instrumen virtual yang dapat digunakan untuk memantau akuisisi data secara *real time*. LabView unggul dalam membuat antarmuka pengguna melalui pemrograman grafis sederhana (Fakhrudin, dkk, 2020). Dengan LabView, perilaku kecepatan atau performa mesin dapat dipantau dengan waktu nyata.

Tujuan penelitian ini untuk memperdalam penggunaan LabView serta memperdalam penelitian LabView untuk mengamati perilaku kecepatan motor sangkar tupai 3 phasa dengan metode IFOC (Kontrol Berorientasi medan Tidak Langsung) dan PID. Pada penelitian ini menggunakan metode IFOC dengan inverter yang digunakan adalah inverter dengan rangkaian SPWM. Tujuan dari penelitian ini adalah mengimplementasikan IFOC dengan kontroler PID pada konfigurasi motor sangkar tupai 3 phasa, yang dimonitor kinerja dinamisnya serta dianalisis menggunakan LabView.

2. METODOLOGI PENELITIAN

2.1 *Field Oriented Control* (FOC)

Kontrol motor sangkar tupai, yang dikenal sebagai *kontrol berorientasi medan* (FOC), adalah metode pengaturan motor sangkar tupai yang dilakukan dengan memisahkan fluks medan dan arus torsi. Metode ini digunakan karena dapat memisahkan arus medan dan pengaturan arus torsi serta pengaturan motor DC. Penelitian ini menggunakan metode IFOC, di mana metode tersebut memperoleh besaran arus dengan perkiraan menggunakan persamaan FOC (*current model*), untuk sistem ini diperlukan sensor kecepatan sebagai umpan balik.

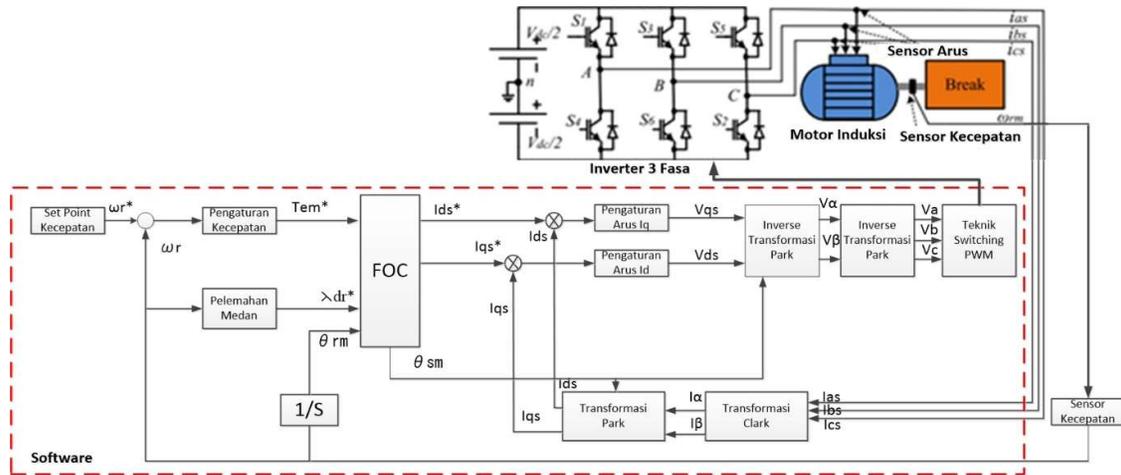
FOC membutuhkan dua konstanta sebagai referensi *input*. Komponen torsi (sejajar dengan koordinat q) dan komponen fluks (sejajar dengan koordinat d). Tegangan 3 phasa, arus, dan fluks dapat dianalisis ke dalam *complex space vector*. Jika diasumsikan i_a , i_b , dan i_c adalah arus sesaat pada fasa stator, maka vektor arus didefinisikan sebagai berikut:

$$\vec{i}_s = i_a + i_b e^{j2\pi/3} + i_c e^{j4\pi/3} \quad (1)$$

Di mana, (a, b, c) adalah sumbu sistem tiga phasa.

Arus *space vektor* ini merupakan sistem sinusoidal 3 phasa. Maka perlu ditransformasikan menjadi sistem koordinat dua *time invariant*. Transformasi ini terbagi ke dalam dua bagian yaitu transformasi *Clark* (a,b,c ke α,β) yang menghasilkan sistem koordinat dua *time variant*

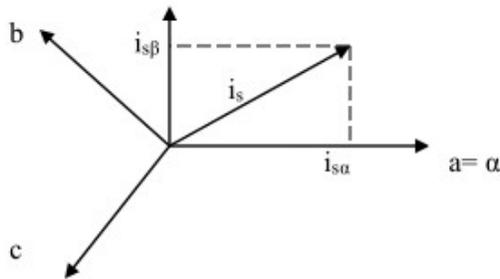
dan transformasi *Park* (α, β ke dq) yang menghasilkan sistem koordinat dua *time invariant* (Hannan, dkk, 2018).



Gambar 1. Diagram *Block* IFOC

2.2 Transformasi Clarke (ABC ke $\alpha\beta$)

Salah satu komponen terpenting dalam pengembangan metode IFOC adalah metode transformasi Clarke dan Park, yang mengubah besaran 3 fasa menjadi besaran 2 fasa, yaitu besaran arus pada stator 3 fasa (i_a , i_b dan i_c) pada medan tetap (kerangka acuan stasioner) terhadap arus stator ortogonal dua fasa (i_α dan i_β) di bidang vertikal (kerangka acuan ortogonal), sedangkan arus stator (i_α dan i_β) diubah menjadi arus stator dua fasa (i_{ds} dan i_{qs}) pada bidang putar (rangka acuan putar) dengan menggunakan transformasi parkir seperti pada Gambar 2 menunjukkan hubungan antara vektor arus stator bidang putar (komponen d , bidang tetap antara q) dan arus stator (komponen a, b, c) (Mohan, 2014).



Gambar 2. Transformasi Arus Stator Menjadi Komponen $\alpha\beta$

Secara matematis persamaan-persamaan transformasi dapat ditunjukkan pada persamaan sebagai berikut :

$$\begin{aligned} i_{s\alpha} &= i_a \\ i_{s\beta} &= \frac{1}{\sqrt{3}}i_a + \frac{2}{\sqrt{3}}i_b \end{aligned} \quad (2)$$

atau dapat ditulis sebagai Persamaan (3).

$$\begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{\sqrt{3}} & \frac{2}{\sqrt{3}} & 0 \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} \quad (3)$$

2.3 Transformasi Park ($\alpha\beta$ ke dq)

Transformasi ini bertujuan untuk mengubah komponen dua fasa ortogonal (α, β) pada kerangka stator ke kerangka acuan rotor dq dengan kecepatan sudut ω . Jika dipertimbangkan sumbu d sejajar dengan fluksi rotor pada sumbu d, kemudian kita akan berbicara tentang sistem orientasi fluksi. Pada Gambar 3 menunjukkan *space vektor* arus stator dan komponennya pada ($\alpha\beta$) dan pada (dq). Vektor arus pada kerangka dq dirumuskan (Mohan, 2014).

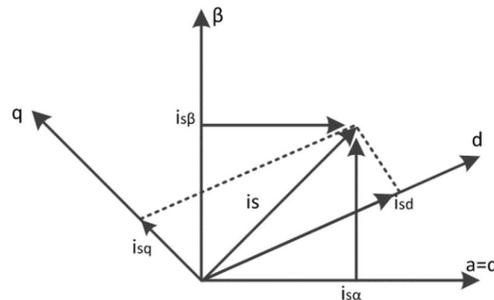
$$i_s = i_{ds} + i_{qs} \quad (4)$$

Dimana ekivalennya adalah.

$$\begin{aligned} i_{ds} &= i_\alpha \cos(\theta) + i_\beta \sin(\theta) \\ i_{qs} &= -i_\alpha \sin(\theta) + i_\beta \cos(\theta) \end{aligned} \quad (5)$$

Atau dapat dinyatakan dalam bentuk matriks sebagai berikut.

$$\begin{bmatrix} i_{ds} \\ i_{qs} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} \quad (6)$$



Gambar 3. Proyeksi Arus Stator dari Koordinat $\alpha\beta$ Menjadi Koordinat Dq yang Berorientasi pada Fluks Rotor.

2.4 PID Controller

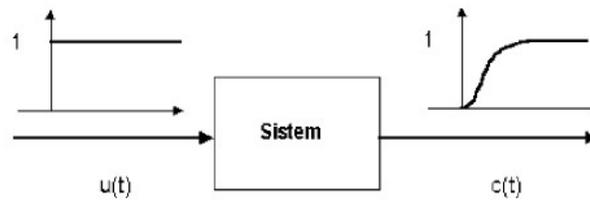
PID controller adalah controller yang digunakan pada penelitian ini, kontroler ini merupakan gabungan dari tiga kontroler yang disusun secara paralel dan diharapkan dapat memberikan kontribusi keunggulan masing-masing kontroler K_p , K_i , dan K_d (Ardana, 2013). Persamaan kontroler PID dapat dituliskan sebagai Persamaan (7) seperti berikut.

$$M(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p T_d \frac{de(t)}{dt} \quad (7)$$

Untuk mendapat luaran yang diinginkan maka diperlukan proses *tuning* untuk konstanta proposional, integral dan deferensial yang ditentukan dengan baik, terdapat metode untuk menentukan K_p, K_i dan K_d tersebut yaitu metode coba-coba (*cut and try method*), metode *Ziegler-Nichols* dan metode tanggapan step.

2.5 Ziegler Nichols

Metode *Ziegler-Nichols* adalah metode yang dipakai pada penelitian ini untuk menghitung konstanta dari PID kontroler metode ini pertama kali diperkenalkan tahun 1942 (Juharsyah, dkk, 2011).



Gambar 4. Step Response

Terdapat dua metode pada metode analisis penentuan parameter K_p , K_i , K_d pada PID *controller* dengan menggunakan metode *Ziegler_Nichols* yaitu kurva respons dan osilasi yang digunakan pada kondisi respons sistem di mana *overshoot* maksimum 25%, namun pada sistem *open loop* metode kurva respons dapat digunakan pada saat sistem *loop* terbuka menerima *input* sinyal *step* serta sistem tidak mengandung elemen integrasi atau kutub kompleks, *output* sinyal adalah kurva-S dalam bentuk seperti yang ditunjukkan pada Gambar 4 (Zahrudin Arif, dkk, 2020). Metode kurva reaksi terdapat beberapa kendala yaitu tidak mampu untuk sistem yang terdapat komponen integrator dan yang yang memiliki *pole* kompleks.

Terdapat dua konstanta dalam proses *tuning* parameter PID menggunakan S-curve, yaitu L (waktu mati) serta T (waktu tunda). Seperti terlihat pada Gambar 4, kurva respons secara bertahap berubah dari L ke atas seiring perubahan yang terjadi pada waktu tunda, saat kurva mencapai keadaan tunak 66%. Untuk menghitung parameter L (*dead time*), dibuat garis singgung yang memotong sumbu horizontal dan garis maksimum, perpotongan garis singgung dan sumbu vertikal adalah ukuran L (*dead time*), serta perpotongan garis maksimum adalah T (waktu tunda) yang diukur dengan L (*dead time*). Proses *Tuning* untuk konstanta PID dengan menggunakan kurva S terdapat dua parameter, seperti ditunjukkan pada Gambar 4 kurva reaksi berubah naik, pada selang waktu L, sedangkan perubahan waktu tunda akan terdapat perubahan, bila kurva telah bernilai 66% dari keadaan stabilnya. Untuk menghitung parameter L (*dead time*), maka pada kurva dibuat garis singgung yang bersinggungan dengan garis kurva. Garis singgung memotong sumbu horizontal dan garis maksimum, perpotongan garis singgung dengan sumbu vertikal adalah ukuran waktu mati, dan perpotongan garis maksimum adalah waktu tunda yang diukur dengan L (*dead time*).

Proses *tuning* konstanta PID dihitung menggunakan kedua parameter tersebut (Kartika & Muhammad, 2016). Proses *tuning* konstanta K_p , T_i , dan T_d pada PID kontroler dengan menggunakan kedua parameter tersebut, atas dasar eksperimen dan saran dari *Zeigler* dan *Nichols*. Tabel 1 merupakan rumusan penalaan parameter PID berdasarkan cara kurva reaksi.

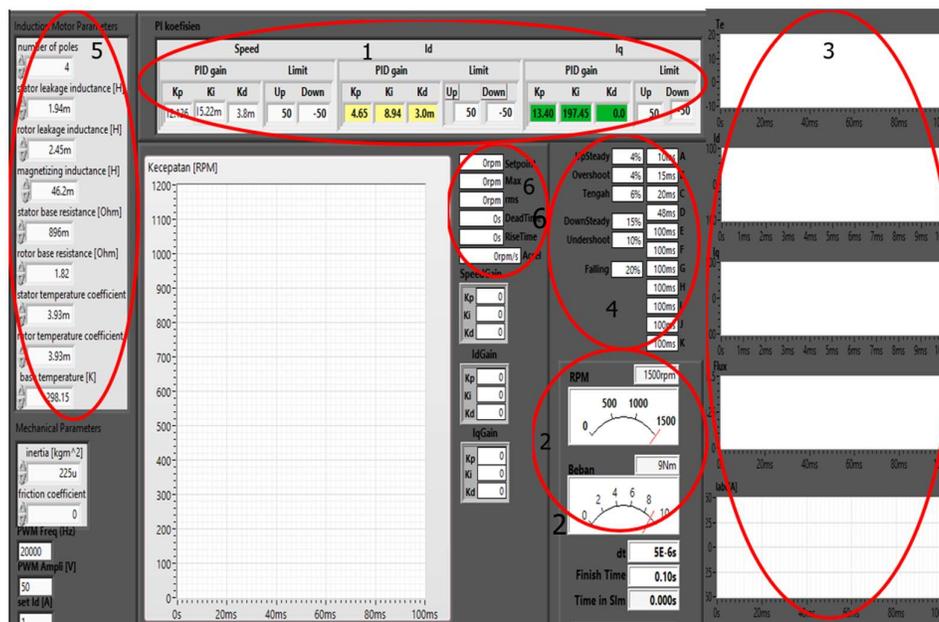
Tabel 1. Parameter PID dengan Metode Kurva Reaksi

Time Pengontrol	K_P	T_i	T_d
P	T/L	\sim	0
I	$0.9 T/L$	$L/0.3$	0
PID	$1.2 T/L$	$2L$	$0.5 L$

3. ANALISIS DAN UJI SIMULASI

3.1 Perancangan Simulasi LabView

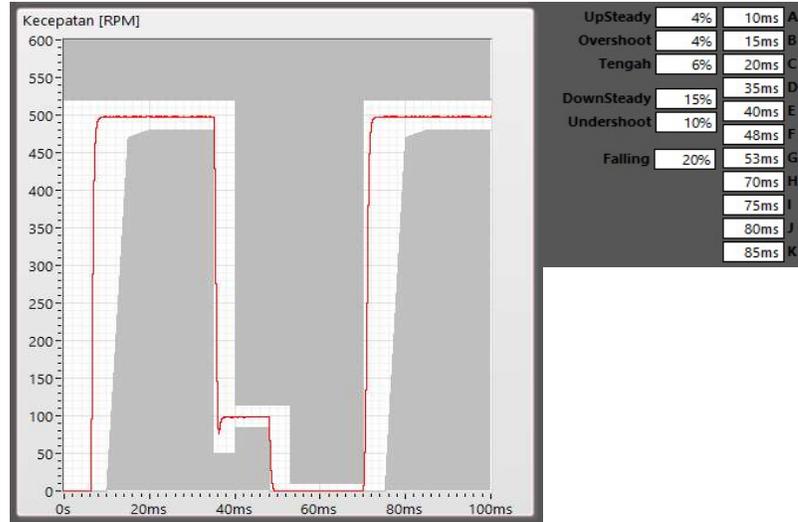
Pada penelitian ini, LabView digunakan sebagai media untuk menyimulasikan, dengan antarmuka pengguna yang mirip dengan yang ditunjukkan pada Gambar 5, dengan masing-masing panel memiliki tujuan yang berbeda.



Gambar 5. Tampilan Antarmuka Labview

Gambar 5 adalah tampilan antarmuka LabView, dimana terdapat beberapa fungsi yang beragam. Pada bagian 1 merupakan block dalam mengatur/memasukkan nilai parameter dari *PID controller*, yaitu terdapat PID untuk *speed controller*, PID untuk pengaturan arus fluksi (*Id Gain*) dan PID untuk pengaturan arus torsi (*Iq Gain*). Pada pelaksanaannya pengaturan dilakukan untuk kecepatan *saja*, dengan membuat parameter *Kd*, *Ki* dan *Kp* dibuat dengan *default* pada *Id* dan *Iq* dibuat *default*. Pada bagian 2 merupakan *block* untuk mengatur/set RPM sebagai set point masukan (dalam satuan Rpm) yang nantinya sinyal *error* kecepatan digunakan sebagai masukan dari *PID controller*. Selain itu pada bagian 2 juga terdapat set nilai dari beban (satuan *Newton meter (Nm)*), pada penelitian ini nilai dari beban akan diubah ketika sudah mencapai kondisi *steady state* pada kecepatan untuk mengamati dan menganalisis kondisi dinamisnya (perubahan beban).

Pada bagian 3 merupakan *block* tampilan dari beberapa gelombang, di antaranya yaitu gelombang torsi motor, arus fluksi (*Id*), arus torsi (*Iq*), Fluks dan arus 3 phasa (*Iabc*). Bagian 4 merupakan *block* untuk mengamati performa sistem. Dalam menganalisis respons dari suatu sistem kontrol terdapat beberapa parameter, di antaranya adalah *risetime*, *settling time*, *dead time*, *steady time*, *falling time*, *overshoot* dan *undershoot*. Sehingga terlebih dahulu harus menentukan batas maksimal dari beberapa parameter tersebut, agar dapat dikatakan sistem memiliki respons yang bagus. Sehingga dari skema pewaktu yang telah dibuat, sistem dapat dikatakan memiliki respons kecepatan yang bagus ketika berada di antara batas minimal dan *maximal* (di daerah berwarna putih) dari skema pewaktu. Skema waktu yang didesain pada simulasi seperti pada Gambar 6.



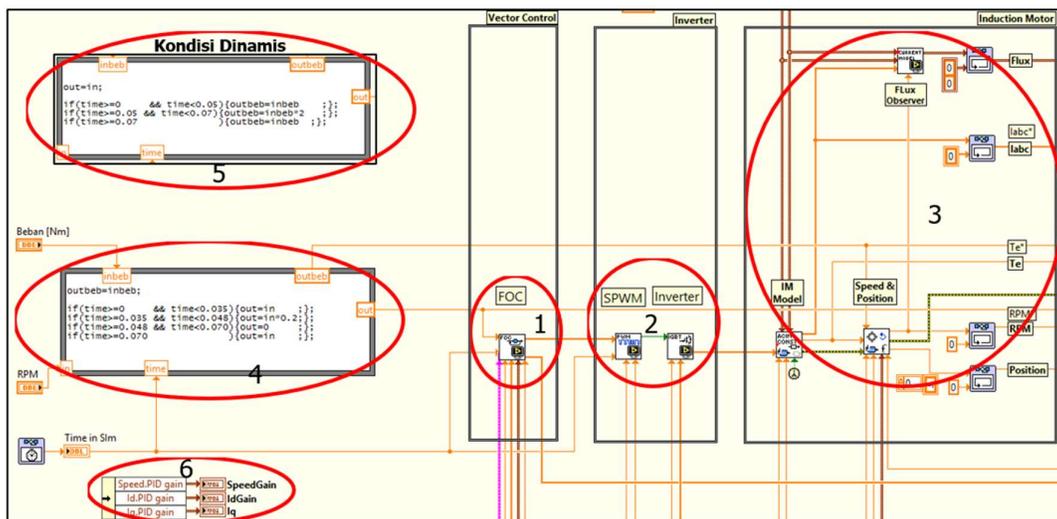
Gambar 6. Desain Skema Pewaktu pada Simulasi

Pada bagian 5 merupakan *block* untuk memasukkan nilai spesifikasi dari motor sangkar tupai yang digunakan. Spesifikasi dari motor sangkar tupai seperti ditunjukkan Tabel 2.

Tabel 2. Data Spesifikasi Motor Sangkar Tupai 3 Phasa

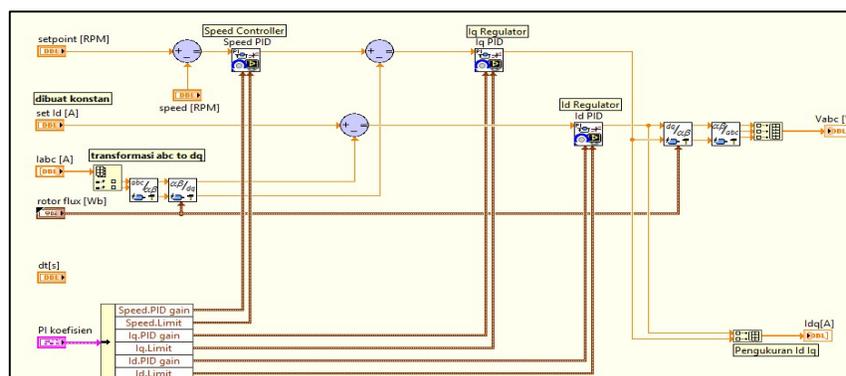
No	Parameter	Nilai	Satuan
1	R_{stator}	0.896	Ω
2	R_{rotor}	1.82	Ω
3	L_{stator}	1.94	mH
4	L_{rotor}	2.45	mH
5	$L_{magnetization}$	46.2	mH
6	frekuensi	50	Hz
7	pole	4	unit

Bagian 6 merupakan *block* nilai respons kecepatan yang dihasilkan sistem, diantaranya setpoint, kecepatan maksimal dan minimal yang dihasilkan sistem. Diagram blok program dari sistem pada LabView ditunjukkan pada Gambar 7.



Gambar 7. Diagram Block Program Sistem pada Labview

Pada Gambar 7 merupakan diagram *block* program sistem simulasi dengan menggunakan LabView. Pada blok diagram di atas terdiri dari beberapa blok, yaitu pada bagian 1 merupakan *block* dari *vector control* yaitu *Field Oriented Control (FOC)*. Bagian 2 merupakan *block* dari inverter 3 phasa dengan teknik *switching* (sinyal pembangkitan) SPWM. Bagian 3 merupakan *block* dari pemodelan motor sangkar tupai. Bagian 4 merupakan *block* dari pengaturan dari *set point* kecepatan. Bagian 5 merupakan *block* saat pengamatan dalam kondisi keadaan dinamis yaitu pengaturan dari perubahan nilai beban. Perubahan beban dilakukan ketika sistem sudah mencapai *steady state* sehingga dapat diamati dan dianalisis dari *recovery time* dalam menuju ke *steady state* kembali. Bagian 6 merupakan *block* dari *setting* nilai parameter PID pada pengaturan *speed*, pengaturan *id* dan *iq*. Diagram blok dari *PID controller* pada pengaturan *speed Indirect Field Oriented Control (IFOC)* pada LabView ditunjukkan pada Gambar 8.



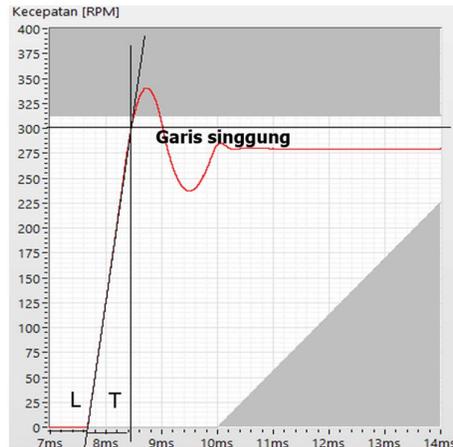
Gambar 8. PID Controller pada Pengaturan Speed IFOC Labview

Pada Gambar 8 besarnya kecepatan *actual* yang didapatkan akan dibandingkan dengan nilai *set point* kecepatan sehingga akan menghasilkan sinyal error yang digunakan sebagai input dari *PID controller* pada pengaturan kecepatan, pengaturan yang dilakukan pada penelitian ini hanya untuk kecepatan dan nilai *id* dibuat konstan sebesar 1 Pu.

3.2 Penentuan Parameter PID Controller

PID Controller adalah *controller* yang terdiri dari *proportional*, *integral* dan *derivative*, bila ketiga kontroler tersebut dikombinasikan maka mempunyai keunggulan dalam memperbaiki respons sistem, karena kelebihan dari masing-masing kontrol tersebut. Ketiga parameter K_p , K_i , dan K_d harus ditentukan secara optimal untuk mendapatkan performa sistem yang telah ditentukan. Dalam menentukan nilai parameter PID ada beberapa metode, sering disebut sebagai metode kurva reaksi. Kurva reaksi didasarkan pada respons sistem *open loop*.

Kurva S dari kurva respons memiliki dua parameter, yaitu waktu_tunda (T) serta waktu_mati (L). Ketika keadaan kurva respons mulai naik, waktu intervalnya adalah L , T adalah kondisi perubahan kurva setelah mencapai kondisi tunak 66%. L dihitung dengan menarik garis dari respons *loop* terbuka yang dihasilkan, dan menyinggung garis kurva saat mulai naik, serta memotong absis dan garis maksimum (nilai nominal), sedangkan titik potong dengan garis maksimum (*set value*) disebut T , yang diukur dari L . Untuk informasi lebih lanjut, dapat dijelaskan seperti pada Gambar 11, yang menunjukkan contoh cara menentukan konstanta L dan T kondisi *loop* terbuka sistem saat kecepatan target 500 rpm.



Gambar 9. Penentuan Waktu Mati (L) dan Waktu Tunda (T) Set Point 500 rpm

Dengan menggunakan teorema *Ziegler Nichloz* 1 yaitu dengan menggunakan kurva S, dapat dipakai untuk menghitung parameter K_p , K_i dan K_d . Nilai K_p , K_i dan K_d dihitung dengan Persamaan (8) sampai (10) (Ogata, 2010).

$$K_p = 1,2 \left(\frac{T}{L} \right) \tag{8}$$

$$T_i = 2 L \tag{9}$$

$$T_d = 0,5 L \tag{10}$$

3.3 Hasil Simulasi

Pada penelitian ini dilakukan pengujian simulasi dengan dua kondisi, yaitu kondisi statis dan kondisi dinamis untuk kondisi pengujian masing-masing, kemudian dievaluasi kinerja dari respons sistem kontrol seperti *steady_state_error*, *rise_time*, *overshoot*, *setting_time* dan *undershoot*. Sistem pengaturan yang dilakukan pada penelitian ini adalah pengaturan kecepatan, nilai parameter PID pada i_d dan i_q dibuat konstan, seperti pada Tabel 3 merupakan nilai parameter PID pada i_d dan i_q konstan.

Tabel 3. Nilai Parameter PID pada i_d dan i_q

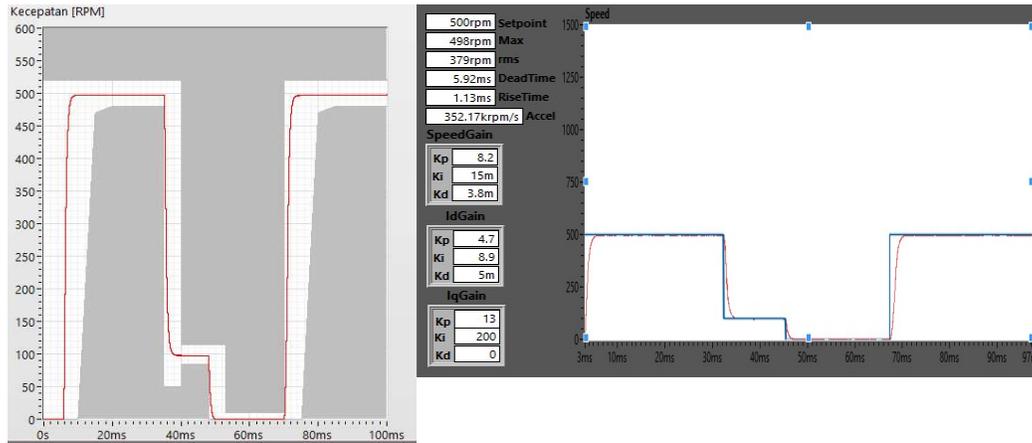
Kp	Ki	Kd	Keterangan
3,6	8,01	0,006	i_d
12,9	198,2	0	i_q

Pada pengujian simulasi dengan kondisi statis dilakukan dengan nilai torsi beban yang konstan yaitu sebesar 9 Nm, untuk kondisi dinamis dilakukan dengan beberapa set poin kecepatan yaitu 500 rpm, 800 rpm, 1000 rpm dan 1200 rpm. Dari respons kecepatan *open loop* sistem yang didapatkan dengan nilai variasi *set point* maka dapat ditentukan nilai parameter PID menggunakan metode *Ziegler Nicholz* 1 seperti yang sudah dijelaskan pada pembahasan penentuan parameter PID *controller*. Nilai parameter PID *controller* pada pengaturan kecepatan yang didapatkan dengan variasi nilai *set_point* seperti pada Tabel 4.

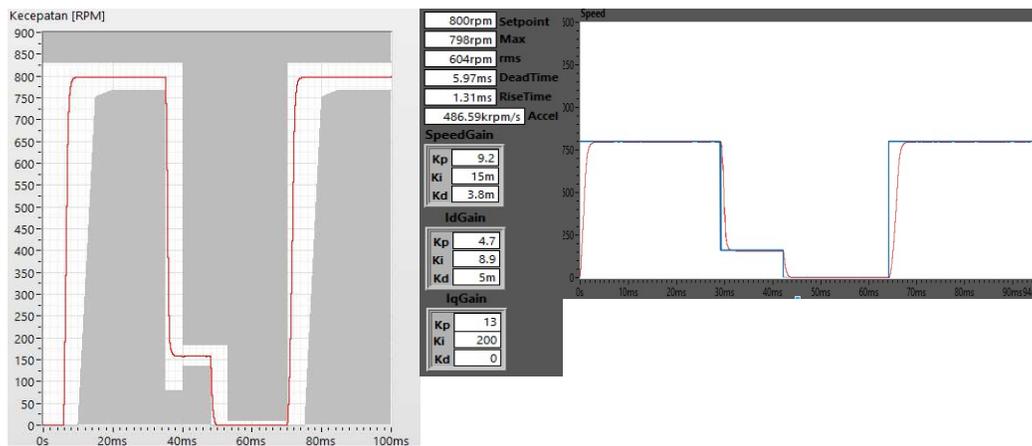
Sehingga dengan menentukan nilai parameter PID pada Tabel 5 maka didapatkan respons kondisi *close loop* sistem pengaturan kecepatan motor sangkar tupai 3 fasa dengan metode IFOC menggunakan PID *controller* dengan berbagai *set point* kecepatan seperti pada Gambar 10 sampai Gambar 13.

Tabel 4. Nilai Kp, Ki Dan Kd pada Pengaturan Kecepatan

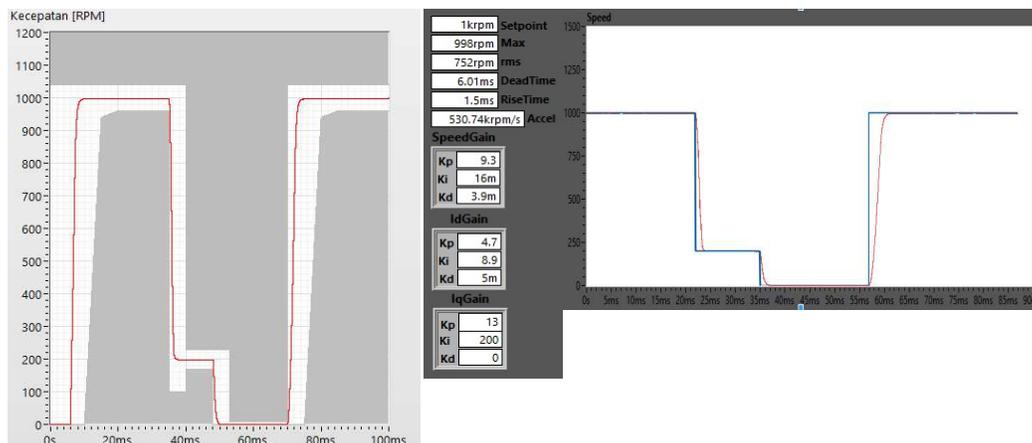
SET POINT (rpm)	KP	KI	KD
500	8,2010	0,015240	0,003810
800	9,3380	0,015260	0,0038150
1000	9,3420	0,015560	0,003890
1200	8,3980	0,01560	0,00390



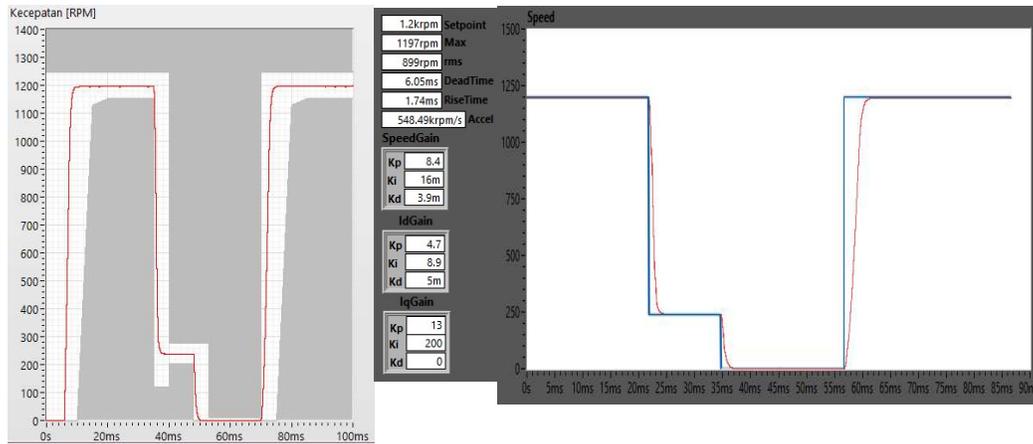
Gambar 10. Close Loop Sistem dengan Set Point 500 rpm.



Gambar 11. Close Loop Sistem dengan Set Point 800 rpm.



Gambar 12. Close Loop Sistem dengan Set Point 1000 rpm



Gambar 13. Close Loop Sistem dengan Set Point 1200 rpm

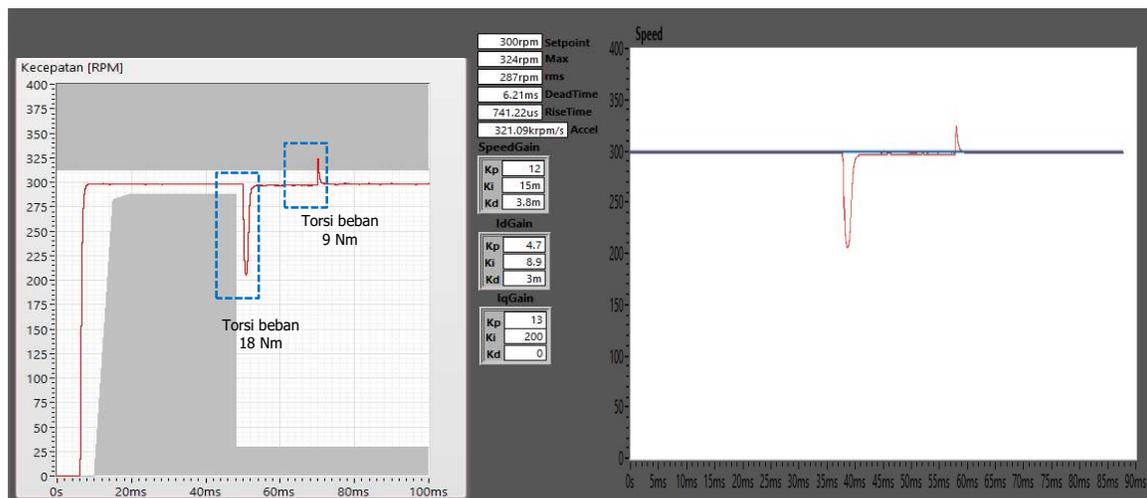
Pada pengujian kondisi statis, dengan waktu simulasi sebesar 0.1 sekon, didapatkan respons kecepatan *close loop* sistem seperti pada Gambar 8 sampai dengan Gambar 9. Dari respons yang didapatkan dengan mengatur nilai parameter PID yang didapatkan, dengan variasi nilai *set point* dapat diketahui bahwa respons kecepatan yang didapatkan menjadi lebih baik, karena dari hasil simulasi didapat peningkatan dari nilai *rise_time* serta *settling_time*, *settling_time* adalah waktu yang dipakai oleh motor untuk menuju ke keadaan *steady state*, *rise_time* adalah waktu yang menunjukkan peningkatan respons sistem dari 10% sampai 90% dari nilai respons *steady state*. Respons kecepatan lebih baik ketika nilai *settling_time* dan *rise_time* diperoleh dengan menggunakan kontrol PID dalam pengaturan kecepatan kontrol medan tidak langsung (IFOC). Selain itu, dari respons *close loop* yang didapat, tidak adanya respons sistem yang berada di luar batas maksimal dan minimal dari skema respons sewaktu atau respons sistem berada pada daerah putih, sehingga dapat dikatakan sistem mempunyai respons yang bagus. Jika dibandingkan dengan kondisi sistem *open loop* (tidak ada pengontrol), respons *close loop* sistem memiliki respons yang bagus yaitu dengan waktu *rise_time* dan *settling_time* yang cepat, serta adanya penurunan pada error *steady state*. Selain itu jika dilihat dari respons yang didapatkan, ketika dioperasikan pada kondisi *low speed* yaitu pada *set point* 500 rpm respons yang didapatkan lebih stabil, hal ini merupakan kelebihan dari *Indirect Field Oriented Control* (IFOC) yang bisa dioperasikan pada kecepatan dengan rendah dengan respons yang stabil dibanding dengan *scalar control*. Sehingga pengamatan respons dapat ditabelkan seperti pada Tabel 5.

Tabel 5. Perbandingan Karakteristik Kecepatan Motor Hasil Simulasi dengan Variasi Nilai Set Point

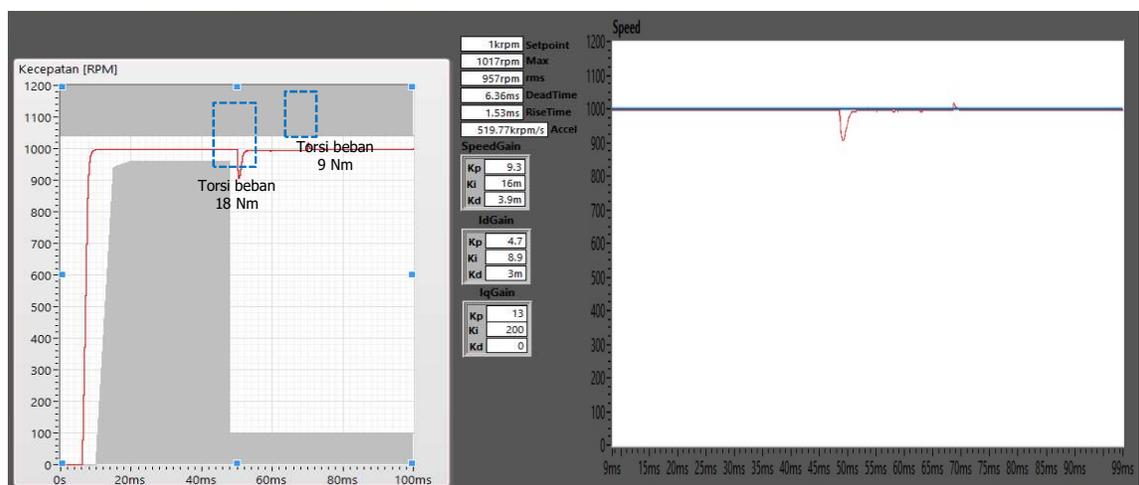
Set Point (Rpm)	500	800	1000	1200
Kp	8,2	9,3	9,3	8,3
Ki	0,01	0,01	0,01	0,01
Kd	0,003	0,003	0,003	0,003
Dead Time (ms)	5,9	5,9	6,0	6,0
Rise Time (ms)	1,1	1,3	1,5	1,7
Settling time (ms)	9,7	10,2	10,3	11
Error Steady (%)	0,4	0,2	0,2	0,2

Pada pengujian simulasi dengan kondisi dinamis dilakukan dengan nilai torsi beban yang diubah ketika sistem sudah mencapai kondisi *steady state*. Perubahan nilai torsi beban dilakukan pada waktu simulasi ke 50 ms – 70 ms dengan melakukan penambahan dua kali

dari nilai torsi beban awal. Hasil responsnya seperti ditunjukkan pada Gambar 14 dan Gambar 15.



Gambar 14. Respons Dinamis Sistem dengan *Set Point* 300 rpm



Gambar 15. Respons Dinamis Sistem dengan *Set Point* 1000 rpm

Pada pengujian kondisi dinamis, dengan waktu simulasi sebesar 0.1 sekon, didapatkan respons kecepatan *close loop* sistem seperti pada Gambar 14 dan 15. Dari respons yang didapatkan dengan perubahan torsi beban, *PID controller* mempunyai respons yang lama dalam merespons perubahan parameter, sehingga membutuhkan waktu *recovery* yang lama dalam menuju ke keadaan *steady state* kembali. Pada Gambar 14, dengan nilai set point kecepatan sebesar 300 Rpm ketika terjadi perubahan nilai torsi beban dari 9 Nm ke 18 Nm pada waktu simulasi ke 50 ms – 70 ms, membutuhkan waktu *recovery* sebesar 4.9 ms untuk kembali ke keadaan *steady state*. Tetapi setelah adanya respons dari *controller* untuk tetap menjaga ke keadaan *steady*, masih adanya osilasi sehingga menyebabkan sistem kurang stabil dan tidak mencapai set point yang diinginkan. Pada Gambar 15, dengan nilai set point 1000 rpm ketika terjadi perubahan nilai torsi beban dari 9 Nm ke 18 Nm pada waktu simulasi dari 50 ms menuju ke 70 ms, membutuhkan waktu sebesar 1.952 ms untuk kembali ke keadaan *stead state*. Tetapi *PID controller* belum merespons dengan cepat serta tidak dapat menjaga ke nilai *set*

point yang diinginkan. Sehingga PID *controller* kurang cocok jika digunakan pada pengaturan dengan kondisi dinamis (adanya perubahan parameter dari internal maupun eksternal) seperti pada simulasi ini yaitu adanya perubahan nilai torsi beban.

4. KESIMPULAN

Motor sangkar tupai 3 fasa yang dikendalikan dengan menggunakan metode *Indirect Field Oriented Control* (IFOC) serta PID *controller* pada *labview* memiliki hasil yang bagus. Dengan nilai parameter PID yang didapatkan menggunakan teorema *Ziegler Nicholz* 1 pada tiap-tiap nilai set point kecepatan, respons kecepatan yang didapatkan bagus, karena semakin cepatnya respons kecepatan motor dalam menuju keadaan *steady state* pada tiap-tiap nilai *set point*. PID *controller* dapat menekan *error steady* hingga 0.2%. Tetapi pada simulasi kondisi dinamis, penggunaan PID *controller* dinilai kurang cocok, karena respons PID *controller* kurang cepat ketika terjadi penambahan torsi beban. Sehingga membutuhkan waktu *recovery* yang lama, hingga sebesar 4.9 ms pada *set point* 300 rpm.

DAFTAR RUJUKAN

- Aditya, A. W., Ihsan, I., Utomo, R. M., & Hilmansyah, H. (2019). Evaluasi Motor Listrik Sebagai Penggerak Mobil Listrik. *JRST (Jurnal Riset Sains dan Teknologi)*, 3(2), 55-59
- Ardana, I. W. R. (2013). Simulasi sistem kontroler PID untuk motor sangkar tupai menggunakan perangkat lunak matlab/simulink. *Majalah Ilmiah Teknologi Elektro*, 7(2).
- Fakhrudin, H. H., Toar, H., Purwanto, E., Oktavianto, H., Apriyanto, R. A. N., & Aditya, A. W. (2020). Kendali Kecepatan Motor sangkar tupai 3 Fase Berbasis Particle Swarm Optimization (PSO). *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 8(3), 477.
- Ferdiansyah, I. (2019). Analisis Pengaruh Arus Medan (id current axis) Terhadap Kecepatan Motor sangkar tupai 3 fasa Pada Pengaturan Menggunakan Metoda Field Oriented Control (FOC). *JTT (Jurnal Teknologi Terpadu)*, 7(2), 95-100.
- Hannan, M. A., Ali, J. A., Mohamed, A., & Hussain, A. (2018). Optimization techniques to enhance the performance of induction motor drives: A review. *Renewable and Sustainable Energy Reviews*, 81, 1611–1626.
- Iqbal, R. (2015). *Pengaturan Kecepatan Motor sangkar tupai 3 fasa Menggunakan Metoda Vektor Kontrol Berbasis Jaringan Syaraf Tiruan Radial Basis Function–PD (JST RBF–PD) Hybrid* (Doctoral dissertation, Institut Teknologi Sepuluh Nopember).
- Jisha, L. K., & Thomas, A. P. (2013). A Comparative Study on Scalar and Vector Control of Induction Motor Drives. *International conference on Circuits, Controls and Communications (CCUBE) IEEE*, (pp. 1-5).

- Juharsyah, D., Setiawan, I., & Wahyudi, W. (2011). *Perbandingan Unjuk Kerja Controller PID Metoda Pertama Ziegler-Nichols dan CMAC (Cerebellar Model Articulation Controller) pada Pengaturan Plant Suhu* (Doctoral dissertation, Jurusan Teknik Elektro Fakultas Teknik Undip).
- Kartika, D., & Muhammad, I. S. (2016). Desain Dan Implementasi Tanggapan Sistem Kontrol Proportional, Integral, Diferensial (PID) Pada Pembebanan Kompleks Menggunakan Metoda Ziegler-Nichols.
- Mukti, K. H. (2012). Implementasi Scalar Control dan Vector Control dalam Pengaturan Kecepatan Motor sangkar tupai 3 phasa. *ELTEK*, 10(2), 11-21.
- Mohan, N. (2014). *Advanced electric drives: analysis, control, and modeling using MATLAB/Simulink*. John wiley & sons.
- Ogata, K. (2010). *Modern control engineering*. Prentice hall.
- Quang, N. P., & Dittrich, J. A. (2008). *Vector control of three-phase AC machines* (Vol. 2). Berlin, Heidelberg: springer.
- Riba, J. R., Torrez, C. L., Romeral, L., & Garcia, A. (2016). Rare Earth Free Propulsion Motors for Electric Vehicle A Technology Review. *Renewable and Sustainable Energy Reviews (ELSEVIER)*, 57, 367-379.
- Zahrudin Arif, S., Siradjuddin, I., & Budi, E.S. (2020). Kontrol Arah Gerak 4 Buah Omni Wheels Pada Penyedot Debu dengan Metoda PID Secara Wireless. *Jurnal Elektronika dan Otomasi Industri*, 3(3), 112-120.