

Kendali Kecepatan Motor Induksi 3 Fase Berbasis *Particle Swarm Optimization (PSO)*

HANIF HASYIER FAKHRUDDIN¹, HANDRI TOAR², ERA PURWANTO³, HARY
OKTAVIANTO⁴, RADEN AKBAR NUR APRIYANTO⁵, ANGGA WAHYU ADITYA⁶

^{1,3,4,5}Teknik Elektro Politeknik Elektronika Negeri Surabaya, Indonesia

²Teknik Elektro Politeknik Negeri Batam, Indonesia

⁶Teknik Elektro Politeknik Negeri Balikpapan, Indonesia

Email: hanifhasyier@gmail.com

Received 3 Februari 2020 | Revised 23 Maret 2020 | Accepted 25 Mei 2020

ABSTRAK

Motor induksi secara struktur dan kendali standarnya dirancang untuk bekerja pada kecepatan nominal, sehingga sulit mengendalikan kecepatan sesuai kebutuhan karena akan mengubah konstruksi motor. Penelitian tentang pengendalian motor induksi agar semudah mengendalikan motor DC sudah banyak dilakukan oleh peneliti, salah satunya adalah dengan kendali skalar. Kendali skalar banyak digunakan karena memiliki keunggulan sederhana, biaya murah, mudah didesain dan diimplementasikan, serta yang paling penting tidak memerlukan parameter dari motor induksi. Penggunaan kendali skalar yang telah dilengkapi pengendali PID penalaan otomatis, dengan parameter yang telah dioptimalkan algoritma Particle Swarm Optimization (PSO), akan memudahkan pengendalian kecepatan motor induksi tiga fase pada kecepatan beragam. Simulasi penalaan otomatis PID menggunakan PSO telah dilakukan dengan LabView, dengan karakteristik maksimal 10% overshoot, 1% error steady state dan rise time kurang dari 2 milidetik. Sementara dalam pengujian real time dengan MyRIO hasilnya tanpa overshoot, 5.5% error steady state maksimal dan rise time maksimal 5 detik.

Kata kunci: Kendali skalar, PID, Particle Swarm Optimization, LabView

ABSTRACT

Induction motor is designed at nominal speed as default, we have to change its structure to obtain desired speed. Many researchers develop method how to control induction motor as simple as DC motor, one of the methods is scalar control. Scalar control has several benefits, such as simply, low cost, easily designed and implemented, and the main benefit is no necessary motor parameters. Using scalar control with PID controller that optimized Particle Swarm Optimization (PSO) algorithm, will ease to control 3 phase induction motor variant speed. Simulation auto tuning using PSO has done on LabView, it has some characteristic, they are 10% overshoot, 1% steady state error, and rise time within 2ms. In other hand, real time test using MyRIO got no overshoot, 5.5% steady state error maximal, and rise time maximal 5 s characteristic.

Keywords: Scalar control, PID, Particle Swarm Optimization, LabView

1. PENDAHULUAN

Isu kebutuhan energi global menjadi faktor utama penggunaan motor listrik untuk menggantikan motor berbahan bakar minyak. Menurut BP *Energy Outlook* saat ini, konsumsi energi akhir pada bidang transportasi masih didominasi oleh bahan bakar minyak. Sejak berlakunya protokol Kyoto pada tahun 2004 dan Persetujuan Paris pada tahun 2016 dengan berkomitmen mengurangi emisi gas karbon dan mengurangi ancaman perubahan iklim dengan menjaga kenaikan suhu global agar tidak melebihi 2 derajat celsius serta berusaha untuk meningkatkan penggunaan energi yang ramah lingkungan. Hal ini berhasil meningkatkan pemasaran mobil listrik hingga 45% di beberapa negara. Dalam hal ini, motor induksi berhasil menggeser pemasaran motor berbahan bakar minyak (**International Energy Agency, 2017**).

Di lain topik, menurut *International Energy Agency*, sektor industri merupakan konsumen terbesar dari total energi listrik dengan persentase 42% dan akan terus meningkat seiring berkembangnya dunia industri. Di tahun yang berbeda, 2011, *International Energy Agency* menyatakan bahwa 64% persen energi listrik yang diserap industri digunakan untuk menggerakkan motor listrik atau disebut dengan istilah EMDS (*Electric Motor Driven System*). Kemudian pada sektor komersial sebesar 20% merupakan motor listrik, untuk sektor residensial sebesar 13%, dan sektor transportasi dan pertanian sebesar 3% (**International Energy Agency, 2017**).

Menurut Saqib Jamshed Rind yang dipublikasikan dalam *Chinese Journal of Electrical Engineering* 2017 membahas tentang evaluasi perbandingan performa beberapa jenis motor penggerak kendaraan listrik dapat diketahui bahwa motor induksi merupakan pilihan dengan penilaian yang paling tinggi untuk digunakan sebagai penggerak (**Rind, Ren, Hu, Wang, & Jiang, 2017**). Dibandingkan dengan motor jenis lain dengan ukuran dan daya yang sama, motor induksi lebih unggul dalam hal bertenaga besar, lebih murah, tidak ada emisi, tidak menggunakan magnet atau logam langka lainnya, tidak memerlukan bunga api, dan bebas perawatan menjadi faktor pilihan utama para pengguna motor tersebut.

Motor induksi secara struktur dan kendali standarnya memang dirancang untuk bekerja pada kecepatan nominal sehingga sulit untuk mengendalikan kecepatan sesuai yang dibutuhkan karena akan mengubah konstruksi motor. Pengendalian motor induksi terbagi menjadi dua metode, yaitu kendali skalar dan kendali vektor. Kendali skalar banyak digunakan dalam beberapa literatur karena memiliki keunggulan yang sederhana, biaya yang murah, lebih mudah didesain dan diimplementasikan, membutuhkan arus transien yang rendah, dan *error steady state* yang rendah. Keuntungan terbesar metode ini adalah tidak memerlukan parameter dari motor induksi (**Reza, Islam, & Mekhilef, 2014**) (**Suetake, Da Silva, & Goedel, 2011**).

Kendali skalar ditemukan oleh Bimal K. Bose pada tahun 1960. Kendali skalar (V/f) sangat baik diterapkan dalam pengendalian kecepatan yang beragam atau torsi yang beragam. Prinsip dasar kendali ini adalah menjaga perbandingan tegangan dengan frekuensi (V/f) selalu konstan, sehingga mempertahankan medan magnet di celah udara (*air gap*) motor induksi (**Hannan, Ali, Mohamed, & Hussain, 2018**).

Beberapa penelitian tentang metode untuk meningkatkan kendali V/f dilakukan seperti pada rujukan (**Hannan, Ali, Mohamed, & Hussain, 2018**) (**Suetake, Da Silva, & Goedel, 2011**). Metode paling mendasar adalah kendali PID, penentuan parameter PID yang paling diminati adalah metode Ziegler Nichols karena lebih mudah dalam mencari parameternya.

Beberapa metode juga ditambahkan untuk penala otomatis parameter PID, contohnya Fuzzy-PID, GA-PID, NN-PID, dan lain-lain (**Hannan, Ali, Mohamed, & Hussain, 2018**).

Particle Swarm Optimization (PSO) merupakan metode terbaru yang ditemukan pada tahun 1995 oleh Eberhart dan Kennedy. PSO terinspirasi dari cara burung berkelompok. Metode ini menjanjikan implementasi yang tidak rumit tetapi memiliki keandalan yang sangat baik (**Ramli, dkk, 2015**). Prinsip kerja dari algoritma ini adalah mencari parameter PID yang lolos dari kelompok lokal. Selanjutnya dari yang lolos di kelompok tersebut dicari yang paling optimal untuk menjadi lokal optimal. Kemudian lokal optimal dari banyak kelompok akan disaring menjadi global optimal (**Salem, Awadallah, & Bayoumi, 2015**).

LabView merupakan instrumen virtual yang mendukung *monitoring* secara *real time* dalam akuisisi data. LabView unggul dalam membangun antar muka pengguna dengan pemrograman grafik yang mudah (**Bhavin, dkk, 2016**) (**Nhizanth & Gopalakrishnan, 2015**). Dalam implementasinya, LabView mendukung perangkat keras seperti produk National Instrument bernama MyRIO. Dengan menggunakan MyRIO, akan mudah menerapkan akuisisi data kecepatan motor secara *real time* karena keandalan dan kemampuan pemrosesan sinyal berbasis FPGA yang dimiliki MyRIO.

Secara umum, penelitian ini bertujuan untuk memperkaya penelitian LabView dan implementasinya di Indonesia. Secara khusus, penelitian ini melakukan pengamatan terhadap respon kecepatan motor induksi 3 fase dengan optimasi parameter PID yang diperoleh dari metode *Particle Swarm Optimization* (PSO). Kendali motor induksi yang digunakan adalah kendali skalar. Sedangkan untuk kendali *inverter* menggunakan metode SPWM yang tertanam pada MyRIO. Tujuan dari penelitian ini adalah mengimplementasikan kendali cerdas PSO pada pengendalian motor induksi 3 fase secara *real time* dan menganalisis performa dinamikanya menggunakan LabView.

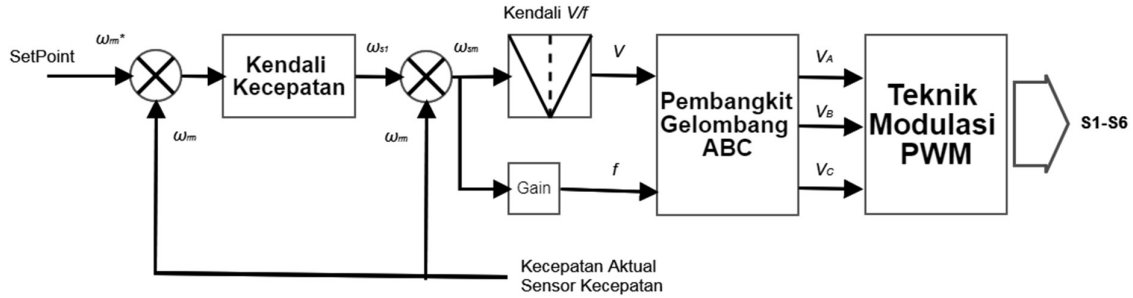
2. METODOLOGI PENELITIAN

2.1 Kendali Skalar

Kendali skalar pada motor induksi pertama kali dikenalkan pada tahun 1960 oleh Bimal K. Bose. Kendali skalar dibagi menjadi sistem kendali *open loop* dan *closed loop*. Pada *open loop* tidak dibutuhkan respon kecepatan yang akurat karena sistem ini tanpa sensor umpan balik. Contoh penerapan *open loop* seperti pada ventilasi, AC, kipas, dan *blower* (**Saidur, Mekhilef, Ali, Safari, & Mohammed, 2012**). Sedangkan untuk *closed loop*, dibutuhkan sensor kecepatan untuk mengubah tegangan dan frekuensi. Prinsip dasar kendali ini adalah menjaga perbandingan tegangan dengan frekuensi (V/f) selalu konstan, sehingga mempertahankan medan magnet di celah udara (*air gap*) motor induksi (**Reza, Islam, & Mekhilef, 2014**). Perbandingan ini bisa dituliskan menjadi Persamaan (1) berikut:

$$\Psi_m \cong \frac{V_p}{f} \cong K_v \quad (1)$$

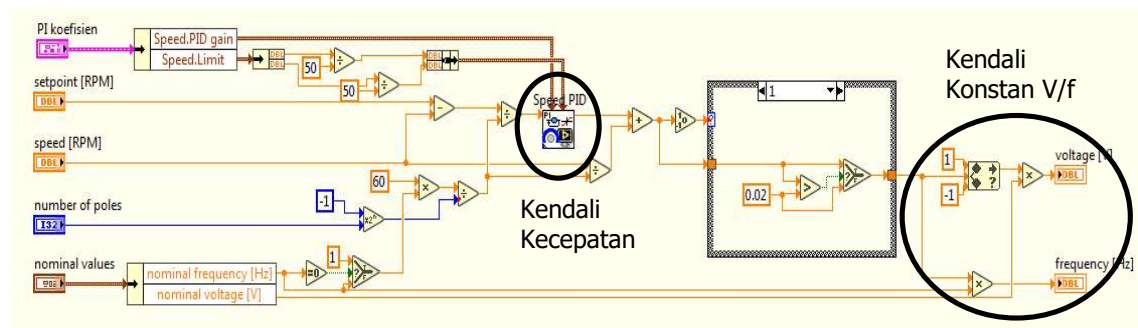
Dimana Ψ_m adalah medan magnet maksimal di celah udara, V_p adalah tegangan maksimal, dan K_v adalah rasio dari V_p dengan f . Diagram blok dari metode kendali V/f dan slip kecepatan (ω_{s1}) ditunjukkan pada Gambar 1.



Gambar 1. Diagram Blok Kendali Skalar

Slip kecepatan (ω_{s1}) adalah variabel dari kecepatan yang dihitung pada karakteristik dasar dari motor induksi. Kecepatan slip motor ditambah dengan kecepatan rotor (ω_{rm}) untuk mendapat (ω_{sm}). Kecepatan sinkron tersebut kemudian dikonversi menjadi frekuensi sinkron untuk mendapat tegangan maksimal pada pengendali V/f. Batasan utama dari kendali ini adalah performa yang kurang baik pada kecepatan rendah (**Alsofyani & Idris, 2013**) (**Reza, Islam, & Mekhilef, 2014**) (**Suetake, dkk, 2011**).

Pada penelitian ini, kendali kecepatan merupakan PID hasil penalaan otomatis menggunakan PSO. Semua blok pada Gambar 1 terletak pada hardware MyRIO seperti pada Gambar 12. Adapun isi dari blok kendali skalar (V/f) pada LabView seperti pada Gambar 2.



Gambar 2. Kendali Skalar (V/f) pada LabView

2.2 Particle Swarm Optimization (PSO) - PID

PSO merupakan teknik komputasi yang terbaru, ditemukan oleh Ebehart dan Kennedy pada tahun 1995. PSO terinspirasi dari cara burung berkelompok (**Ramli, dkk, 2015**). Posisi dari tiap partikel dapat dianggap sebagai calon solusi (*candidate solution*) bagi suatu masalah optimisasi. Setiap partikel diberi suatu fungsi *fitness* yang dirancang sesuai dengan menunjuk masalah yang bersesuaian.

Partikel dari PSO menempati 2 tempat, tempat pertama adalah titik terbaik dalam suatu kelompok atau iterasi (*local best*), sedangkan tempat kedua adalah titik terbaik dari semua iterasi (*global best*). Algoritma PSO bergantung pada 2 faktor, kecepatan (*velocity*) dan posisi dari partikel-partikel. Faktor-faktor tersebut dapat diperbarui menggunakan Persamaan (2) dan Persamaan (3) (**Salem, dkk, 2015**).

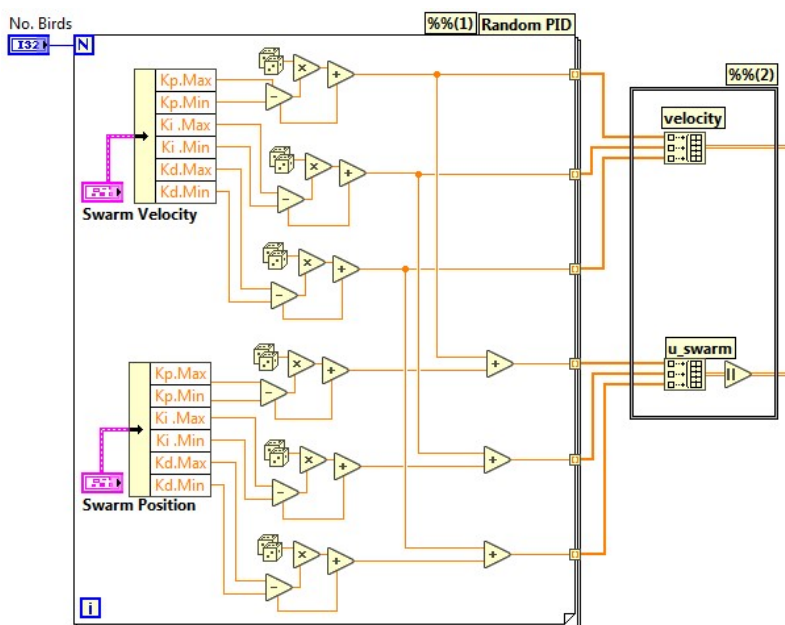
$$V_i^d(t + 1) = wV_i^d(t) + c_1r_1 (P_i^d(t) - X_i^d(t)) + c_2r_2 (P_i^d(t) - X_i^d(t)) \quad (2)$$

$$X_i^d(t + 1) = X_i^d(t) + V_i^d(t + 1) \quad (3)$$

Dimana c_1 dan c_2 masing-masing adalah laju perubahan sosial; r_1 dan r_2 masing-masing adalah random antara 0-1; V adalah faktor kecepatan dari individu i pada iterasi d ; t adalah iterasi saat ini; w adalah bobot inersia; dan X adalah faktor posisi.

PSO memiliki keunggulan dalam mencapai pola memusat dan kemampuan untuk memecahkan masalah optimasi yang kompleks dalam berbagai macam domain. Batasan dari PSO adalah mudah terjebak pada lokal optimal, dan jika ada kesalahan dalam memilih parameter maka hasilnya akan tidak baik (**Chao, Lin, & Lai, 2015**).

Metode PID merupakan metode konvensional yang hasilnya sangat baik. PID merupakan gabungan dari penguatan *Proportional* (P), *Integral* (I), dan *Derivative* (D). Perpaduan ini berfungsi untuk menutupi kekurangan dan mempertahankan keunggulan karakter dari masing-masing. Metode PID merupakan persamaan yang sensitif terhadap perubahan fungsi alih dari sistem, oleh sebab itu metode PID sering digabung dengan metode lain agar bisa menala secara otomatis (**Ali, Hannan, & Mohamed, 2014**). Pada penelitian ini parameter PID akan ditala secara offline menggunakan PSO. Sehingga Persamaan (2) dan Persamaan (3) berlaku untuk tiap nilai Kp, Ki, dan Kd seperti pada Gambar 3 pada LabView. Setelah itu dilanjutkan proses optimasi parameter Kp, Ki, dan Kd seperti pada Gambar 10 untuk mencari *Global Best*.

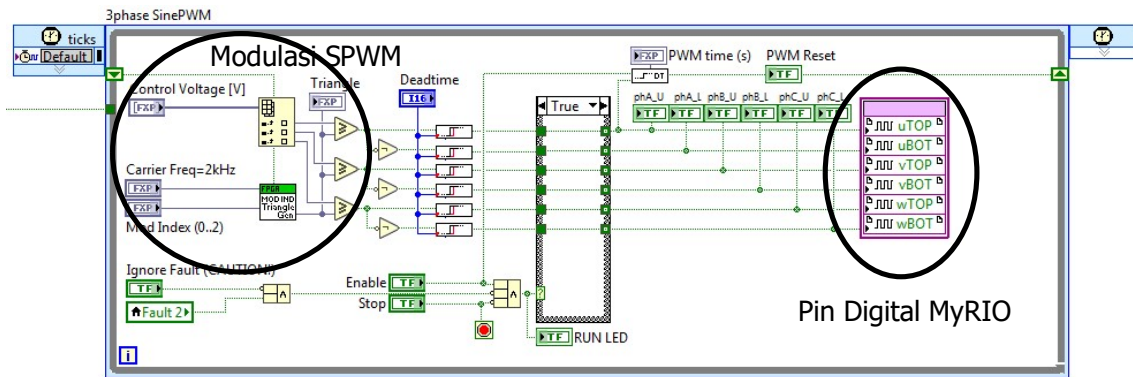


Gambar 3. Penalaan Parameter PID dengan PSO pada LabView

2.3 SPWM Inverter

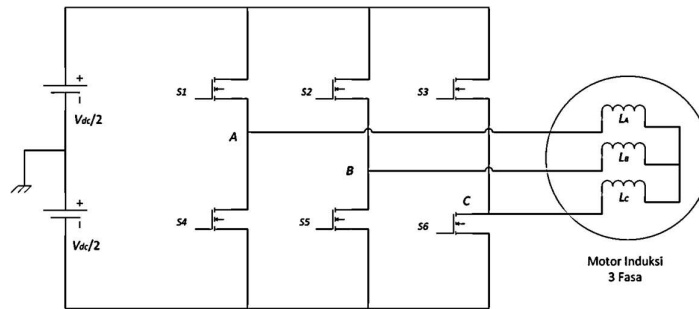
Inverter merupakan rangkaian yang dapat merubah tegangan DC menjadi tegangan AC dengan frekuensi sesuai kebutuhan. Apabila tegangan masukan konstan, tegangan keluaran dapat diatur melalui pengaturan *gain* PWM. *Gain inverter* merupakan rasio tegangan keluaran AC terhadap tegangan masukan DC. *Inverter* pada kendali motor 3 fase merupakan jenis VSI (*Voltage Source Inverter*). Sedangkan untuk kendali *inverter* 3 fase umumnya menggunakan SPWM (Sinusoidal PWM), yaitu teknik memodulasi sinyal sinus 50Hz dengan sinyal gergaji berfrekuensi 2kHz-15kHz. Proses modulasi sinyal pada MyRIO dilakukan pada chip FPGA agar frekuensi sinyal yang dihasilkan akurat. Gambar 4 merupakan contoh modulasi yang digunakan pada penelitian dengan frekuensi sinyal pembawa sebesar 2kHz. Jika dilakukan filter 500Hz

pada sinyal modulasi maka akan menghasilkan sinyal sinus 50Hz (**Ferdiansyah, Sutedjo, Yanaratri, & Raharja, 2019**).



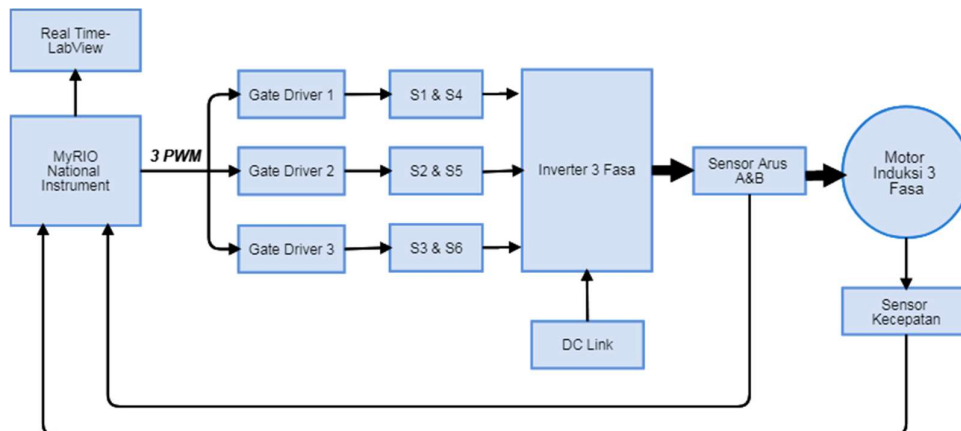
Gambar 4. Sinyal Modulasi SPWM MyRIO

Pada desain perangkat keras, MyRIO menghasilkan 3 sinyal modulasi pada pin digitalnya dengan tegangan 0 Volt dan 3.3 Volt. Kemudian 3 sinyal tersebut dikuatkan oleh FOD8412 dan dijadikan menjadi 6 sinyal oleh rangkaian *gate driver* menggunakan IC IR2111. Keluaran IR2111 yang berupa 6 sinyal selanjutnya dimasukkan pada kaki gate MOSFET (S1-S6) rangkaian *inverter* 3 fase untuk menggerakkan motor induksi 3 fase seperti Gambar 5.



Gambar 5. Rangkaian Inverter 3 Fase

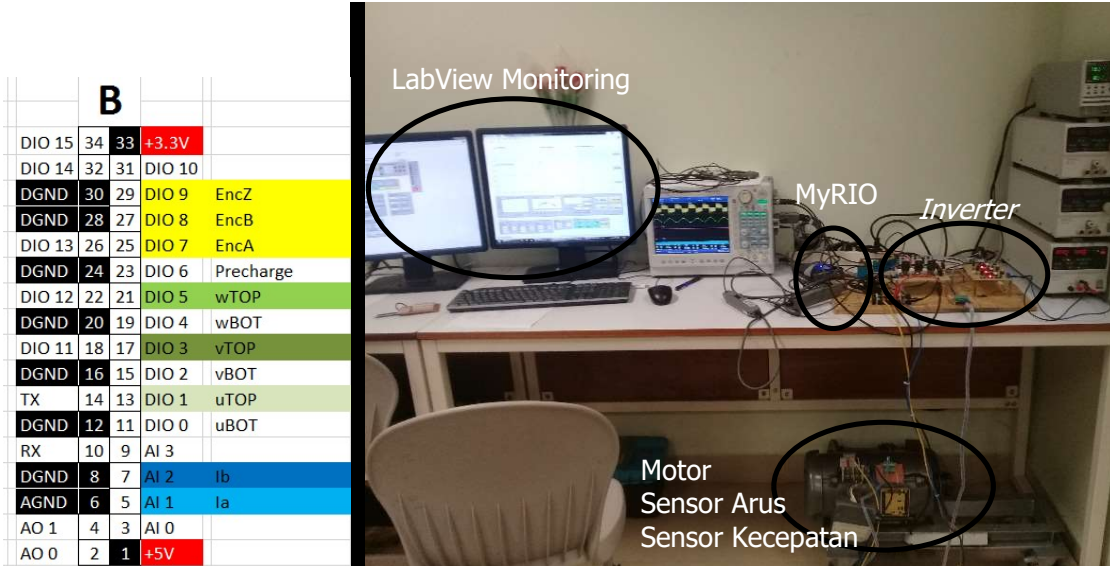
Rangkaian Pengujian Secara lengkap digambarkan dalam diagram blok Gambar 6.



Gambar 6. Digram Blok Pengujian dan Implementasi

Konfigurasi pin pada MyRIO yang digunakan pada penelitian ini dijelaskan pada Gambar 7.

Implementasi MyRIO pada Kendali Kecepatan Motor Induksi 3 Fase dengan Berbasis *Particle Swarm Optimization (PSO)*



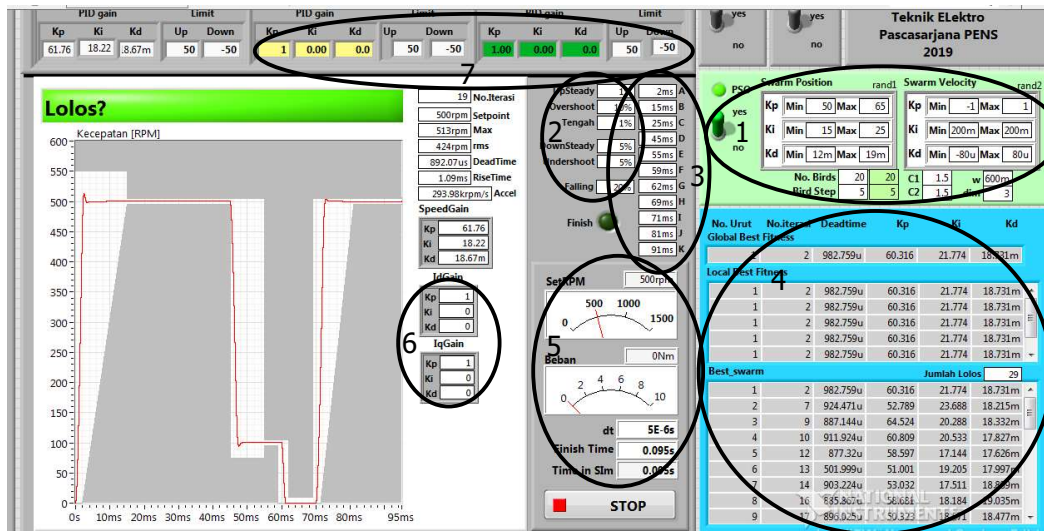
Gambar 7. Konfigurasi Pin MyRIO yang Digunakan & Rangkaian Pengujian

Port yang digunakan adalah port B dari MyRIO. EncZ, EncB, dan EncA merupakan pin-pin digital yang dipakai untuk membaca sensor kecepatan. wTop, vTop, dan uTop merupakan pin-pin digital yang menghasilkan sinyal modulasi SPWM untuk dikirimkan menuju *gate driver*. Ia dan Ib masing-masing merupakan pin ADC yang dipakai untuk membaca sensor arus pada fase A dan fase B.

3. PENGUJIAN DAN ANALISIS

3.1 Desain Simulasi pada LabView

Penggunaan LabView pada makalah ini terbagi menjadi 2 proyek, yaitu proyek simulasi untuk melakukan penalaan otomatis PID menggunakan PSO dan proyek *real time* antar muka kendali motor menggunakan metode *V/f closed loop*. Pada proyek simulasi memiliki tampilan antar muka seperti pada Gambar 8 dengan tiap panel memiliki fungsi beragam.

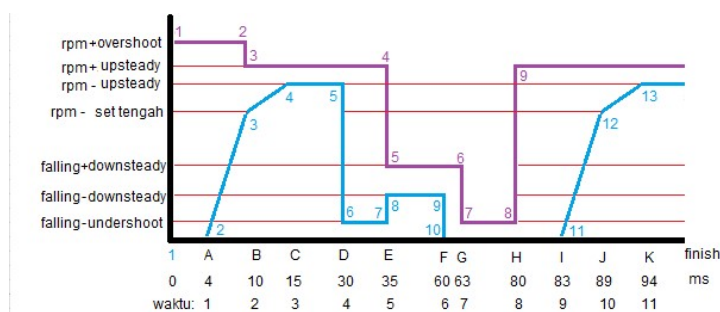


Gambar 8. Tampilan Panel Depan LabView untuk Simulasi

Panel berwarna hijau, ditunjukkan lingkaran 1, merupakan pengaturan awal dari PSO. Meliputi rentang nilai K_p , K_i , dan K_d (dinamakan *swarm position*), rentang perubahan nilai K_p , K_i , dan K_d (dinamakan *swarm velocity*), jumlah burung dalam 1 kelompok (dinamakan *number birds*), jumlah kelompok dalam 1 populasi (dinamakan *Bird Step*), C_1 dan C_2 merupakan laju perubahan sosial, nilai bobot inersia atau w , nilai dimensi atau dim . Metode penalaan otomatis dengan PSO bisa digunakan dengan cara mengaktifkan saklar PSO menjadi ON.

Lingkaran 2 berisi aturan performa dari sistem yang diinginkan. Dikatakan lolos jika parameter PID menghasilkan performa sistem (garis merah) sepenuhnya berjalan di jalur putih. Aturan performa meliputi *error steady* maksimal saat kecepatan acuan (dituliskan *UpSteady* dalam persen), *overshoot* maksimal (dituliskan *Overshoot* dalam persen), *error steady* maksimal saat kecepatan diturunkan menjadi 20% acuan (dituliskan *Tengah* dalam persen), *error steady* maksimal saat kecepatan diturunkan menjadi 0 Rpm (dituliskan *DownSteady* dalam persen), *undershoot* maksimal ketika kecepatan acuan diturunkan 20% dari nilai semula (dituliskan *Undershoot* dalam persen), dan penurunan berapa persen kecepatan acuan (dituliskan *Falling* dalam persen).

Adapun untuk huruf A-K pada lingkaran 3 merupakan aturan performa dalam waktu (dalam satuan milidetik). Pewaktuan ini berkaitan dalam menentukan beberapa aturan seperti: menentukan maksimal *deadtime*, *rise time*, *steady time* dan *falling time*. Lebih jelasnya, detail skema pewaktuan terdapat pada Gambar 9.



Gambar 9. Posisi Pewaktuan Variabel A-K pada Simulasi

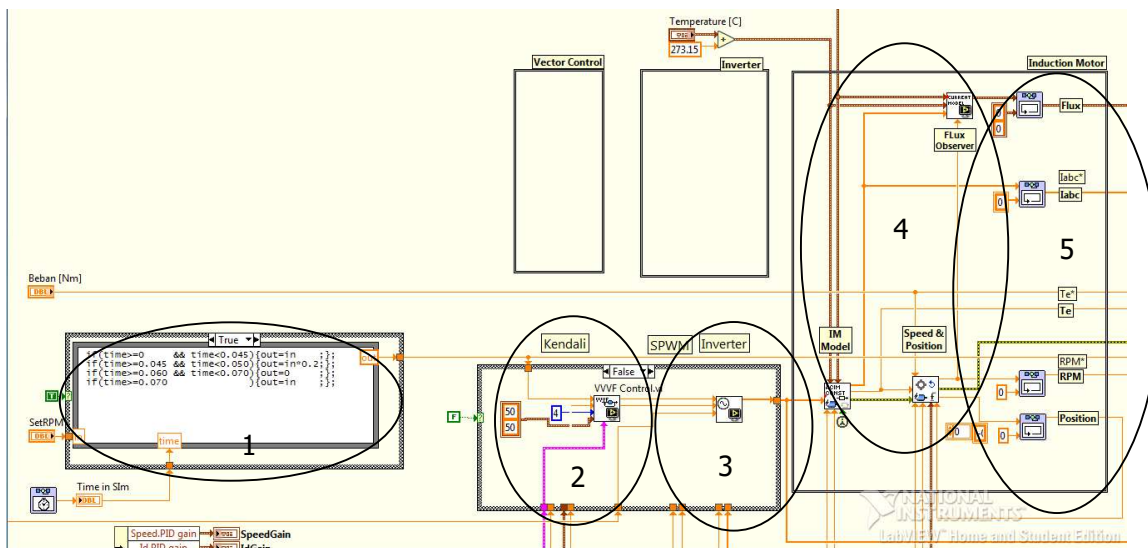
Panel berwarna biru pada lingkaran 4 merupakan tabel dari nilai-nilai partikel hasil penalaan otomatis PSO-PID yang lolos dalam pembelajaran. Partikel yang lolos dalam sebuah kelompok parameternya dicatat dalam tabel *best swarm*, parameter yang dicatat yaitu: nomor urut, iterasi, *dead time*, K_p , K_i , dan K_d . Tabel *local best fitness* berisi individu terbaik dari kelompoknya masing-masing. Tabel *global best fitness* merupakan individu yang terbaik dari seluruh kelompok dalam populasi. Parameter-parameter *global best fitness* adalah parameter PID yang sudah dioptimasi dengan algoritma PSO. Penentuan individu yang terbaik berdasarkan dari parameter *dead time* yang terbesar, hal ini berdasarkan respon ketika pembebanan. Ketika *dead time* diambil yang terkecil, maka ketika ada beban pada motor, waktu *start* motor berputar akan mengalami *delay*. *Delay* tidak konsisten ini harus dihindari ketika mendesain kendali motor yang mengutamakan kenyamanan seperti pada kendaraan listrik.

Pada lingkaran 5, SetRPM merupakan masukan kecepatan acuan (dalam satuan Rpm) yang diinginkan untuk dinala parameter PID-nya. Pada pengujian ini beban diatur konstan 0 Nm (tanpa beban) dan panjang simulasi yang dianalisis penalaannya selama 95 milidetik (*Finish Time*).

Implementasi MyRIO pada Kendali Kecepatan Motor Induksi 3 Fase dengan Berbasis *Particle Swarm Optimization* (PSO)

Beberapa hal yang tidak dipakai pada panel depan ini, ditunjukkan pada lingkaran 6 dan 7, adalah nilai parameter PID pada I_d *Gain* dan I_q *Gain*. Hal tersebut dikarenakan pada kendali skalar tidak bisa melakukan kendali terhadap I_d dan I_q . I_d dan I_q merupakan parameter dalam kendali vektor, sehingga pada makalah ini tidak akan dibahas.

Panel depan program simulasi memiliki diagram blok seperti pada Gambar 10.



Gambar 10. Diagram Blok LabView untuk Simulasi

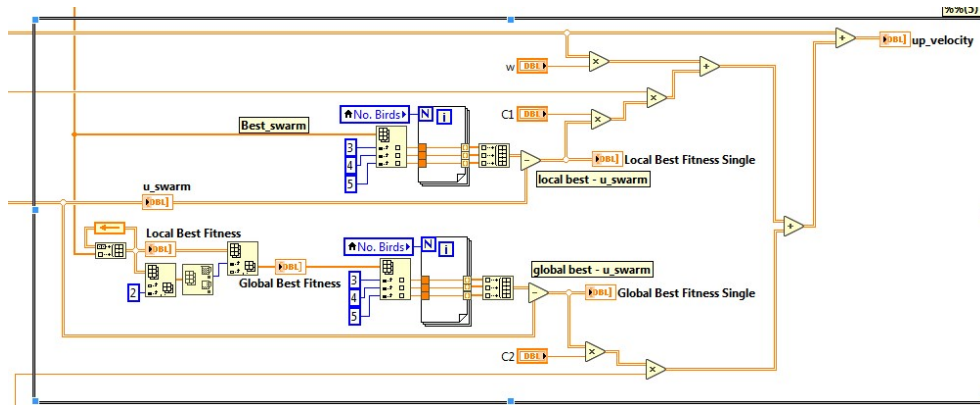
Gambar 9 merupakan diagram blok dari simulasi pada LabView. Pada digram di atas terdapat beberapa bagian, yaitu: 1. Blok pengaturan kecepatan acuan, 2. Blok kendali V/f, 3. *Inverter* 3 fase, 4. Permodelan motor, 5. Tampilan menuju panel depan.

LabView mode simulasi memungkinkan penggunaanya untuk memodelkan motor AC sebagai sebuah diagram blok. Parameter-parameter yang dimasukkan pada permodelan motor induksi (IM Model) adalah parameter asli motor 3 fase yang akan digunakan. Pada penelitian ini digunakan motor induksi 3 fase dengan kapasitas daya 1.5 KW, dengan kecepatan sinkron 1435 Rpm, tegangan 220 V dan arus 4.3 A. Motor tersebut diuji dengan menggunakan beberapa *standart test*, diantaranya *DC test*, *AC test*, *no load test*, dan *full load test*. Tabel 1 menunjukkan hasil parameter motor induksi 3 fase yang didapat.

Tabel 1. Data Parameter Motor Induksi 3 Fase

No	Parameter	Nilai	Satuan
1	R_{stator}	5.27	Ω
2	R_{rotor}	3.40	Ω
3	L_{stator}	4.33	<i>mH</i>
4	L_{rotor}	4.46	<i>mH</i>
5	$L_{magnetization}$	270	<i>mH</i>
6	P_n	1,5	<i>HP</i>
7	V_{l-l}	220	<i>Volt</i>
8	<i>frekuensi</i>	50	<i>Hz</i>
9	<i>pole</i>	4	<i>unit</i>
10	$Rpm_{nominal}$	1435	<i>Rpm</i>

Diagram blok dari penala otomatis menggunakan PSO pada LabView ditunjukkan pada Gambar 11. Pada bagian ini dilakukan optimasi untuk mendapatkan *Global Best* dari populasi.



Gambar 11. Optimasi Parameter PID Metode PSO pada LabView

Diagram pada Gambar 10 didapatkan dari persamaan PSO yaitu posisi dan kecepatan partikel, yang terdapat pada sub bab 2.3, yang dituangkan dalam pemrograman grafis Labview. Hasil penalaan PID dengan PSO pada kendali kecepatan akan dibahas pada poin 3.3 dan 3.4 selanjutnya.

3.2 Desain *Real Time* Antar Muka pada LabView

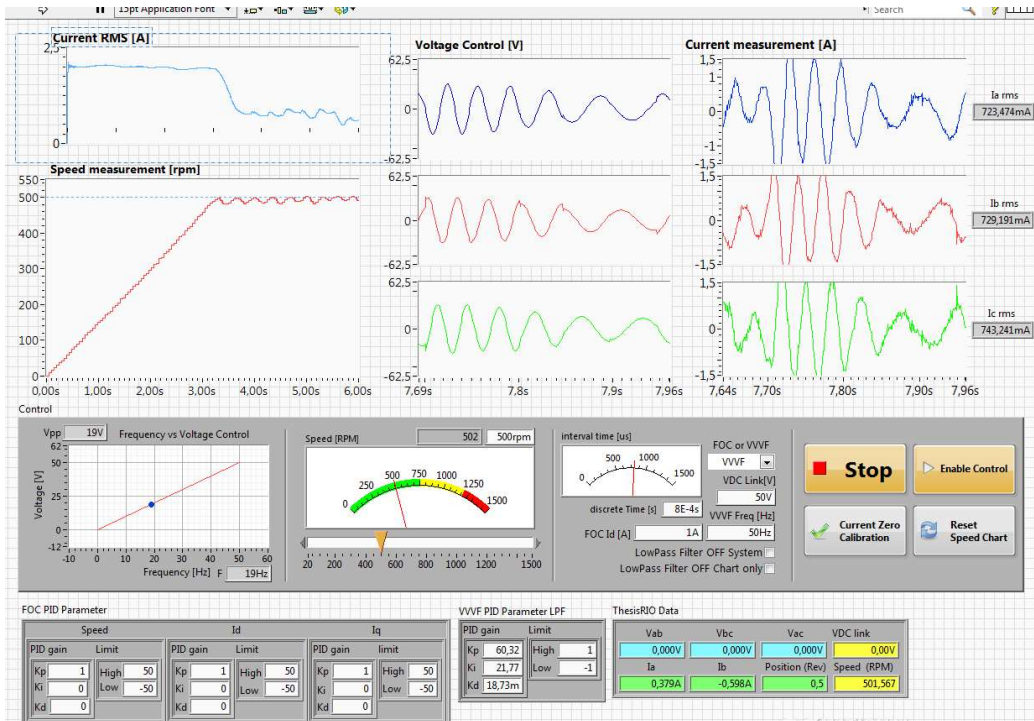
Penelitian ini menggunakan LabView sebagai antar muka untuk mengetahui dan mengamati respon motor induksi 3 fase. Pada panel depan *real time* LabView, terdapat panel untuk menampilkan arus tiap fase, tegangan tiap fase, nilai arus dalam RMS, respon kecepatan aktual motor, nilai perbandingan tegangan dibanding frekuensi, *text box* parameter PID untuk kendali skalar (VVFV PID Parameter), kecepatan acuan, dan beberapa tombol kendali. *Interval time* merupakan *monitoring* batas waktu yang dibutuhkan program *real time* untuk berjalan satu kali iterasi. Gambar 12 merupakan tampilan panel depan program *real time* LabView.

Nilai I_d dan I_q pada panel ini tidak digunakan dalam penelitian kendali skalar. Panel FOC PID Parameter, parameter dari kendali vektor, juga tidak digunakan pada penelitian ini karena penelitian ini berfokus pada kendali skalar.

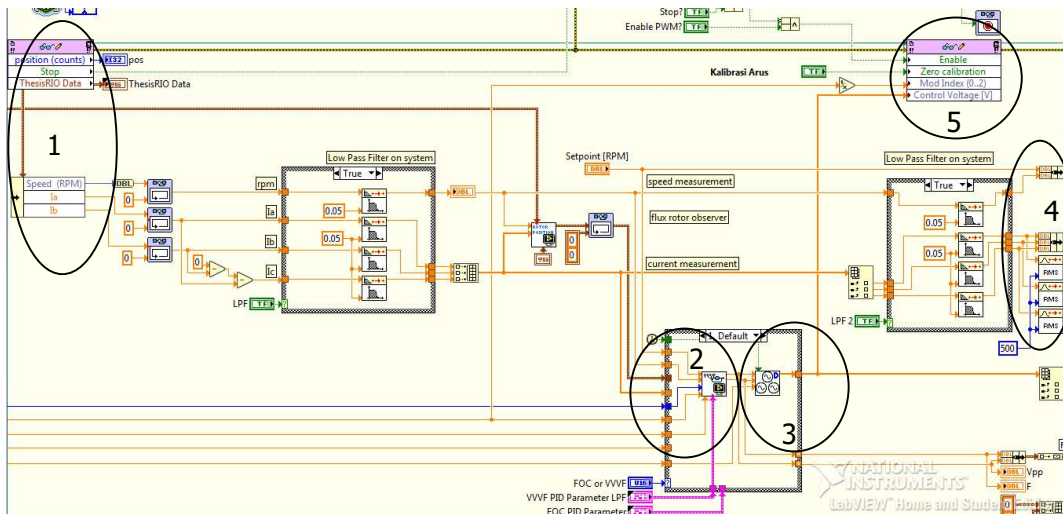
Pada panel program *real time* LabView, nilai tegangan antar fase bernilai 0 Volt karena MyRio tidak terpasang sensor tegangan. Adapun tegangan pada grafik *Voltage Control* merupakan tegangan kendali keluaran dari diagram blok penghasil tegangan 3 fase, sehingga nilai tegangan disini merupakan nilai hasil perhitungan dalam program. Nilai arus pada fase A dan B merupakan hasil pembacaan sensor arus yang sudah dikalibrasi dengan alat ukur amperemeter digital. Nilai arus pada fase C merupakan perhitungan menggunakan hukum kirchoff dimana $I_c = 0 - I_a - I_b$.

Seperti yang dilihat bahwa 1 iterasi program ini berjalan kurang dari 800 mikrodetik. Batasan program yang *real time* adalah kurang dari 2 milidetik. Selain itu nilai pembacaan sensor arus dan sensor kecepatan sudah dikalibrasi dengan alat ukur sehingga data yang ditampilkan pada *real time* LabView merupakan data aktual dalam uji coba penelitian.

Implementasi MyRIO pada Kendali Kecepatan Motor Induksi 3 Fase dengan Berbasis *Particle Swarm Optimization* (PSO)



Gambar 12. Tampilan Panel Depan Program *Real Time* Labview



Gambar 13. Diagram Blok Program *Real Time* Labview

Gambar 13 merupakan diagram blok dari antar muka *real time* LabView. Lingkaran 1 merupakan cara membaca nilai sensor arus dan sensor kecepatan pada pin ADC dan digital MyRIO. Diagram blok *flux observer* tidak digunakan pada penelitian ini karena blok ini hanya digunakan pada kendali vektor. Sedangkan kendali skalar yang terpenting adalah blok VVF dan blok pembangkit sinyal sinus 3 fase pada lingkaran 2. Pada blok VVF inilah nilai tegangan dan frekuensi dijaga agar perbandingannya konstan. Hasil perbandingan tersebut kemudian dimasukkan dalam pembangkit sinyal agar diubah menjadi 3 sinyal sinus untuk menggerakkan motor induksi 3 fase. Lingkaran 3 merupakan bagian mengeluarkan sinyal sinus 3 fase melalui pin-pin dari MyRIO. Lingkaran 4 merupakan bagian menampilkan hasil pembacaan sensor dan hasil kendali V/f pada antar muka dari program *real time* LabView. Sedangkan lingkaran 5

merupakan blok untuk mengeluarkan sinyal-sinyal SPWM dari program menuju pin-pin MyRIO, atau disebut dari perangkat lunak menuju perangkat keras. Blok *low pass filter* berisi filter digital agar pembacaan dari sensor dan keluaran dari program nilainya lebih halus, penggunaan filter dapat memperlambat waktu iterasi hingga 900 milidetik.

Diagram blok dari program *real time* LabView tampak seperti gambar 13. Gambar ini sama seperti diagram blok dari program simulasi (Gambar 10). Letak perbedaannya adalah disini tidak digunakan blok motor induksi karena program ini langsung menggerakkan motor induksi sebenarnya. Masukan dan keluaran dari program adalah pada pin-pin MyRIO, pin-pin masukan berupa ADC dari sensor arus, digital dari sensor kecepatan. Sedangkan pin keluaran adalah sinyal SPWM pada pin digital sebagai kendali tegangan. Pin-pin yang digunakan pada penelitian ini bisa dilihat pada Gambar 7.

3.3 Penentuan Parameter PSO

PSO merupakan algoritma pencarian yang cepat dalam konvergen tetapi rawan terjebak pada lokal optimal. Sehingga diperlukan penentuan rentang nilai Kp, Ki, dan Kd yang sesuai sebelum mulai penalaan secara otomatis. Hal yang dilakukan adalah menentukan rentang nilai parameter secara bebas, kemudian dilakukan penalaan otomatis dengan PSO pada program simulasi. Lakukan dengan nilai kecepatan acuan yang berubah-ubah, sehingga dapat dilihat pola memusatnya dari nilai-nilai parameter. Pada penelitian ini dilakukan pengujian penalaan PID dengan rentang Kp 0-150, Ki 0-50, Kd 0-0.02. Sehingga didapatkan hasil pada Tabel 2.

Tabel 2. Data Uji Parameter Bebas PSO

RPM	Kp	Ki	Kd	RPM	Kp	Ki	Kd	RPM	Kp	Ki	Kd
100	57.39	17.23	17.84m	700	74.96	30.08	18.72m	900	64.34	12.75	18.76m
300	52.16	27.39	17.03m		68.92	21.94	17.71m		77.87	19.36	19.10m
	53.79	26.55	18.09m		49.29	10.28	17.52m		66.00	16.99	13.82m
500	56.62	14.27	19.71m		65.73	26.89	14.62m		53.75	11.64	17.39m
	49.01	28.03	16.47m		78.60	24.57	19.30m		62.91	19.85	12.20m
	60.84	22.25	18.01m		73.93	29.69	16.73m		58.41	23.75	12.61m
	56.49	28.13	18.65m		58.51	22.64	18.73m		53.80	23.26	19.13m
700	59.86	16.76	19.02m		68.21	11.90	18.57m		68.14	26.74	14.44m
	77.95	22.95	19.77m		104.45	19.67	19.41m		60.07	10.56	17.38m
	87.12	23.70	18.93m		59.03	29.38	18.11m		68.69	21.54	19.88m
	64.09	14.48	15.72m		82.64	15.52	19.32m		54.96	17.86	16.03m
	64.81	19.12	19.74m		86.86	14.34	16.71m		61.53	30.39	19.83m
	70.84	23.69	16.58m	73.44	30.08	17.03m	71.21	19.72	15.68m		

Data di atas merupakan semua individu yang lolos dari pengujian karakter kecepatan menggunakan PSO. Dari data Tabel 2 dapat diketahui konvergen rentang PID adalah Kp = 50-65, Ki = 15-25, dan Kd = 1.2-19m dengan tiap kecepatan acuan memiliki 50% lebih data lolos (warna hijau) pada rentang tersebut. Rentang ini kemudian akan dijadikan rentang penalaan PID agar terhindar dari lokal optimal metode PSO.

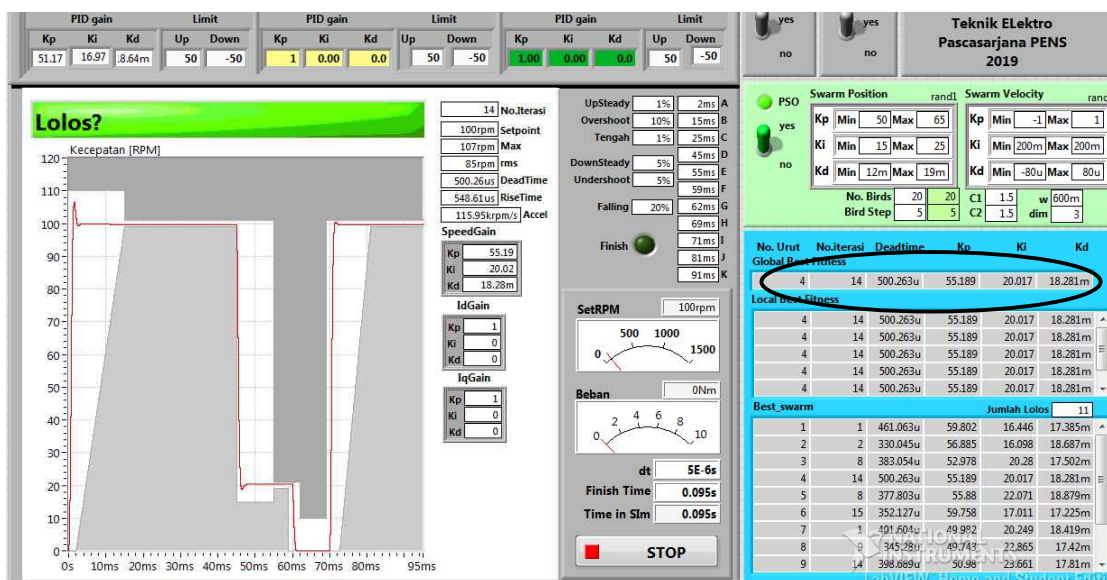
3.4 Implementasi PSO pada *Real Time* LabView

Uji ini mengamati karakteristik dinamis motor menggunakan kendali PID yang sudah ditala secara otomatis menggunakan PID. Pengujian ini memiliki spesifikasi motor seperti pada Tabel 1 dan spesifikasi listrik pada Tabel 3, menggunakan sumber tegangan dari *variable transformer* yang terhubung pada jala-jala.

Tabel 3. Data Parameter Listrik pada Pengujian

No	Parameter	Nilai	Satuan	No	Parameter	Nilai	Satuan
1	$V_{DC Link}$	50.1	V	4	f_{sinus}	50	Hz
2	$C_{DC Link}$	3760	F	5	$f_{carrier}$	15	kHz
3	$V_{Gate Driver}$	12	V				

Pada program simulasi, hasil penalaan nilai PID terdapat pada panel biru *Global Best* yang kemudian digunakan pada pengujian *real time*. Nilai parameter pada kecepatan acuan 100 Rpm yaitu $K_p = 55.18$, $K_i = 20.02$, dan $K_d = 18.28m$ seperti pada Gambar 14. Respon kecepatan dari hasil simulasi tercatat nilai kecepatan acuan 100 Rpm, kecepatan maksimal 107 Rpm, pengujian tanpa beban, *dead time* 556 μs , dan *error steady* 1%.

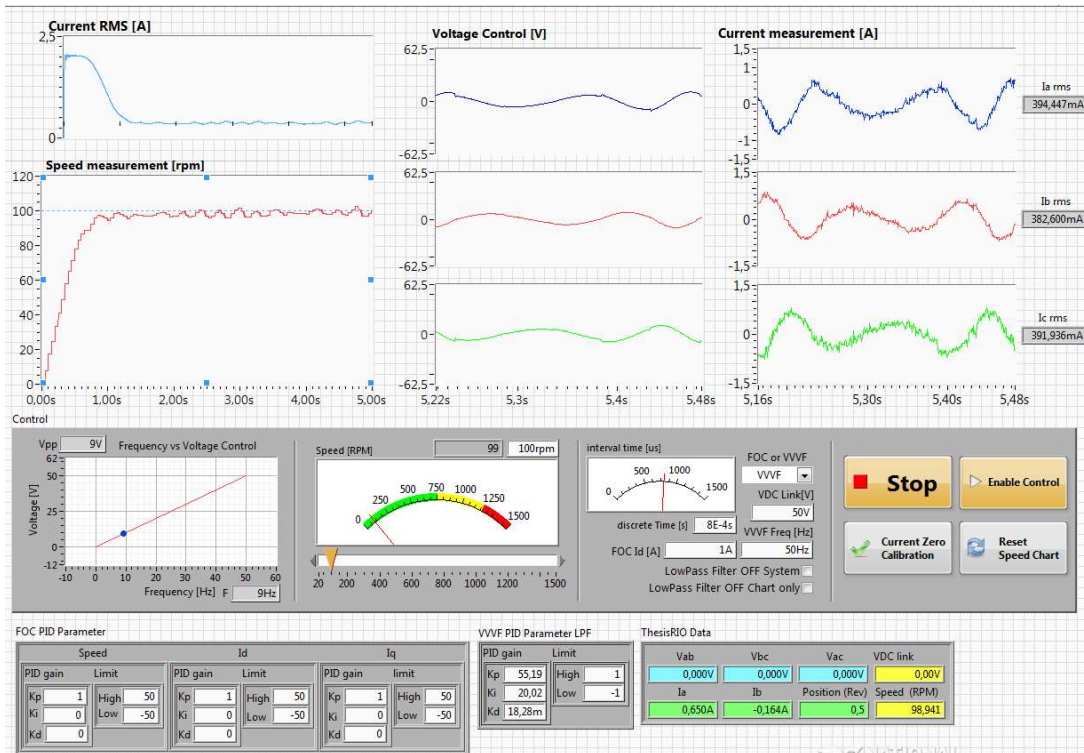


Gambar 14. Hasil Penalaan PSO-PID Kecepatan Acuan 100 Rpm

Pengujian dilakukan untuk membandingkan karakter PID menggunakan PSO secara simulasi dan implementasi. Pengujian dilakukan dengan variasi kecepatan seperti pada Tabel 4. Karena yang diamati adalah karakter dinamis, maka data yang diambil adalah *dead time*, *rise time*, *overshoot*, *error steady* dan konstan V/f. Data karakter dinamis saat simulasi ditulis pada baris Sim, sedangkan untuk implementasi ditulis pada baris RT (*Real Time*). Gambar pengujian *real time* dengan *monitoring* LabView pada kecepatan 100 Rpm disajikan di Gambar 15, sedangkan gambar simulasinya adalah Gambar 14.

Tabel 4. Perbandingan Karakteristik Kecepatan Motor Hasil Simulasi dan Implementasi *Real Time*

Kecepatan Acuan (Rpm)	K_p	K_i	K_d	<i>Dead Time</i> (ms)		<i>Rise Time</i> (ms)		<i>Overshoot</i> (%)		<i>Error Steady</i> (%)		Konstan V/f
				Sim	RT	Sim	RT	Sim	RT	Sim	RT	
100	55.19	20.01	18.28m	0.50	50	0.55	750	7.0	-	1	2.0	9/9
300	54.36	24.44	15.83m	0.63	80	0.73	1820	1.6	-	1	3.1	14/14
500	60.32	21.77	18.73m	0.98	80	1.15	2920	1.2	-	1	4.0	24/24
700	60.84	20.11	15.34m	2.35	80	4.57	4120	2.5	-	1	4.5	32/32
900	61.17	19.74	17.44m	3.81	80	3.74	5000	1.8	-	1	5.5	42/42



Gambar 15. Hasil Pengujian PSO-PID pada Real Time Labview

Pada Gambar 15, grafik pada kotak *Speed Measurement* merupakan respon kecepatan aktual motor. Pada pengujian implementasi secara *real time* menggunakan LabView, hasil respon kecepatan memiliki *error steady* sebesar 2%, *rise time* 750 mili detik, dan tanpa *overshoot*. Sedangkan untuk kendali V/f memiliki nilai tegangan 9V dan frekuensi 9Hz.

4. KESIMPULAN

Implementasi penalaan PID menggunakan algoritma *Particle Swarm Optimization* (PSO) pada LabView dengan perangkat keras MyRIO bisa dilakukan dan dimonitor secara *real time* dengan waktu iterasi kurang dari 600 μ s. Hasil penalaan PID dari PSO memiliki hasil yang memuaskan ketika diimplementasikan dengan catatan penentuan nilai parameter yang tepat. Rentang nilai parameter PID pada penalaan menggunakan PSO yaitu $K_p = 50-65$, $K_i = 15-25$, dan $K_d = 1.2-19m$. Tiap perubahan acuan kecepatan akan mengubah nilai *Global Best* dari penalaan PSO. Dengan batasan *error steady* 1% pada simulasi, menghasilkan maksimal *error steady* 5% pada implementasi. *Rise time* dan *dead time* pada implementasi memiliki perbedaan jauh dari simulasi akan tetapi hal tersebut wajar karena kendali V/f yang tidak memperhatikan karakteristik dari motor induksi, khususnya momen inersia motor. Penambahan PID pada kendali kecepatan dapat meningkatkan karakter dinamis dari motor induksi.

DAFTAR RUJUKAN

- Ali, J. A., Hannan, M. A., & Mohamed, A. (2014). PSO algorithm for three phase induction motor drive with SVPWM switching and V/f control. *Conference Proceeding - 2014 IEEE International Conference on Power and Energy*, (pp. 250–254).

- Alsofyani, I. M., & Idris, N. R. N. (2013). A review on sensorless techniques for sustainable reliability and efficient variable frequency drives of induction motors. *Renewable and Sustainable Energy Reviews*, 24, 111–121.
- Bhavin, A, Khichada, Chusdashma, K. J., M, Vyas. Darshan., & D, Shiyal. Jignesh. (2016). 3-Phase Induction Motor Parameter Monitoring and Analysis Using Labview. *International Journal of Electrical Engineering & Technology (IJEET)*, 7(6), 81–91.
- Chao, K. H., Lin, Y. S., & Lai, U. D. (2015). Improved particle swarm optimization for maximum power point tracking in photovoltaic module arrays. *Applied Energy*, 158, 609–618.
- Ferdiansyah, I., Sutedjo, Yanaratri, D. S., & Raharja, L. P. S. (2019). Comparative study of maximum boost control Z-source inverter with SPWM-VSI for induction motor drive. *International Conference on Information Technology, Information Systems and Electrical Engineering, ICITISEE 2018*, (pp. 380–384).
- 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.
- International Energy Agency. (2017). Global EV Outlook 2017. In *IEA Pub*.
- Nhizanth, A. G. R., & Gopalakrishnan, S. K. (2015). *Stepper Motor Control using LabVIEW and NI-myRIO Saranathan College of Engineering Trichy , India*. 2(12), 478–480.
- Ramli, L., Sam, Y. M., Mohamed, Z., Khairi Aripin, M., Fahezal Ismail, M., & Ramli, L. (2015). Composite nonlinear feedback control with multi-objective particle swarm optimization for active front steering system. *Jurnal Teknologi*, 72(2), 13–20.
- Reza, C. M. F. S., Islam, M. D., & Mekhilef, S. (2014). A review of reliable and energy efficient direct torque controlled induction motor drives. *Renewable and Sustainable Energy*, 37, 919-932.
- Rind, S. J., Ren, Y., Hu, Y., Wang, J., & Jiang, L. (2017). *Configurations and Control of Traction Motors for Electric Vehicles: A Review*. 3(3), 13–14.
- Saidur, R., Mekhilef, S., Ali, M. B., Safari, A., & Mohammed, H. A. (2012). Applications of variable speed drive (VSD) in electrical motors energy savings. *Renewable and Sustainable Energy Reviews*, 16(1), 543–550.
- Salem, F., Awadallah, M. A., & Bayoumi, E. H. E. (2015). *Model Predictive Control for Deadbeat Performance of Induction Motor Drives*. 14, 303–311.
- Suetake, M., Da Silva, I. N., & Goedel, A. (2011). Embedded DSP-based compact fuzzy system and its application for induction-motor V/f speed control. *IEEE Transactions on Industrial Electronics*, 58(3), 750–760.