



TUGAS AKHIR - TE 141599

**PENGATURAN KECEPATAN MOTOR INDUKSI TIGA FASA
DENGAN BEBAN REM MAGNETIK MENGGUNAKAN
METODE FUZZY PD PLUS INTEGRAL**

Dimas Bambang Sulaiman
NRP 2213 105 002

Dosen Pembimbing
Ir. Josaphat Pramudijanto, M.Eng.

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2015



FINAL PROJECT - TE 141599

**SPEED CONTROL OF THREE PHASE INDUCTION MOTOR
WITH MAGNETIC BRAKE LOAD USING PD FUZZY PLUS
INTEGRAL METHOD**

Dimas Bambang Sulaiman
NRP 2213 105 002

Supervisor
Ir. Josaphat Pramudijanto, M.Eng.

ELECTRICAL ENGINEERING DEPARTMENT
Faculty of Industrial Technology
Sepuluh Nopember Institute of Technology
Surabaya 2015

**PENGATURAN KECEPATAN MOTOR INDUKSI TIGA FASA
DENGAN BEBAN REM MAGNETIK MENGGUNAKAN
METODE FUZZY PD PLUS INTEGRAL**

TUGAS AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada
Bidang Studi Teknik Sistem Pengaturan
Jurusan Teknik Elektro
Institut Teknologi Sepuluh Nopember

Menyetujui :



Dosen Pembimbing

Ir. Josaphat Pramudijanto, M.Eng.
NIP. 1962 1005 1990 03 1003

**SURABAYA
JUNI, 2015**

PENGATURAN KECEPATAN MOTOR INDUKSI TIGA FASA DENGAN BEBAN REM MAGNETIK MENGGUNAKAN METODE *FUZZY PD PLUS INTEGRAL*

Nama : Dimas Bambang Sulaiman
Pembimbing : Ir. Josaphat Pramudijanto, M.Eng.

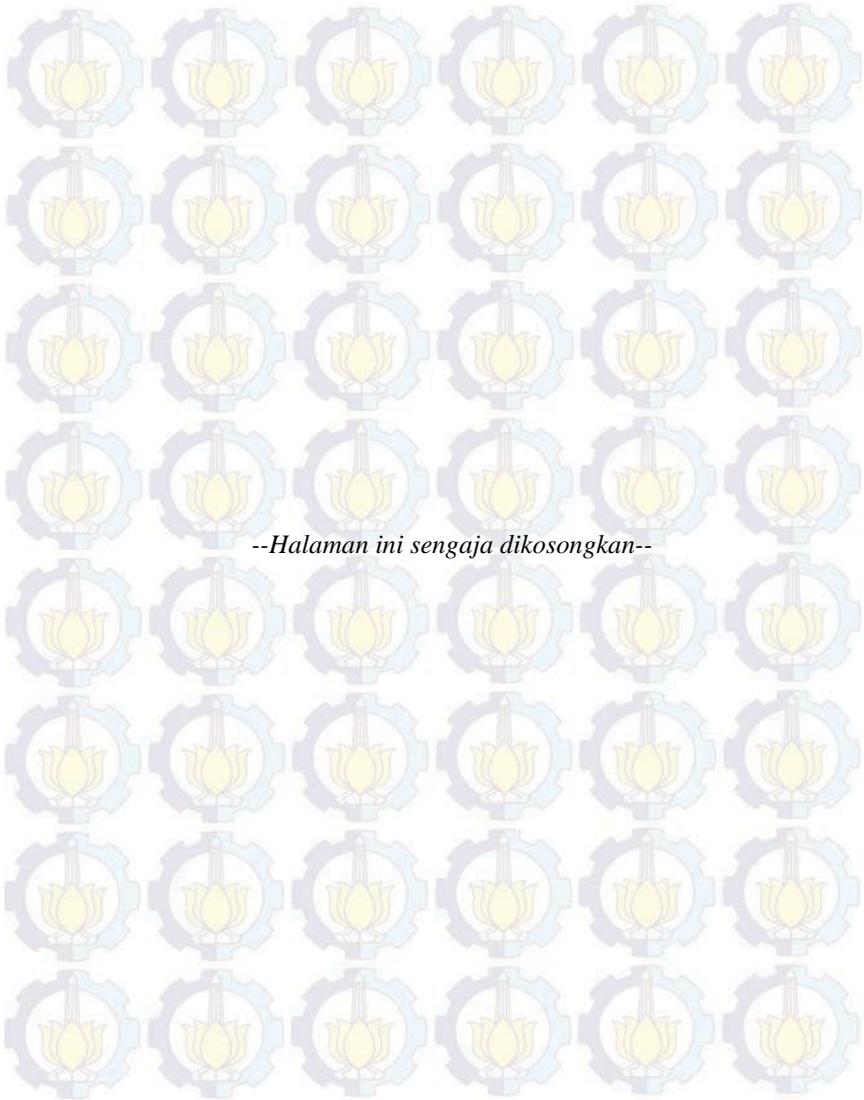
ABSTRAK

Pengaturan kecepatan motor induksi tiga fasa dibuat untuk mengatur putaran motor yang tidak linear, yang diakibatkan oleh gangguan beban, *noise*, dan banyak faktor lainnya. Tetapi ada beberapa kekurangan apabila motor dalam keadaan dinamis dimana keadaan motor dapat berubah secara terus-menerus, hal ini karena pengaruh dari beberapa faktor yang mengganggu dari beban dan *noise*. Sehingga dibutuhkan kontroler yang mampu menjaga *plant* agar tetap stabil.

Pada Tugas Akhir ini untuk menjaga kestabilan *plant* digunakan kontroler *Fuzzy PD plus* kontroler *Integral*. Kontroler *Fuzzy PD plus* kontroler *Integral* merupakan gabungan dari dua kontroler, yaitu kontroler *Fuzzy PD* dan kontroler *Integral*. Kontroler ini unggul dalam bekerja dengan sistem non-linear dan juga tanpa memperhatikan beban, kontroler mampu mendekati nilai referensi sehingga tidak membutuhkan model matematis *plant*.

Hasil pengujian antara simulasi dan implementasi menunjukkan bahwa hasil dari simulasi memiliki nilai % *error steady state* dan *settling time* 5% sebesar 0,044% dan 0,3 detik sedangkan hasil dari implementasi sebesar 0,015% dan 0,11 detik.

Kata Kunci : Kecepatan Motor, Kontroler PD *Fuzzy*, Labview, Rem Magnetik



THREE PHASE INDUCTION MOTOR SPEED CONTROL WITH MAGNETIC BRAKE LOAD USING PD FUZZY PLUS INTEGRAL METHOD

Name : Dimas Bambang Sulaiman
Supervisor : Ir. Josaphat Pramudijanto, M.Eng.

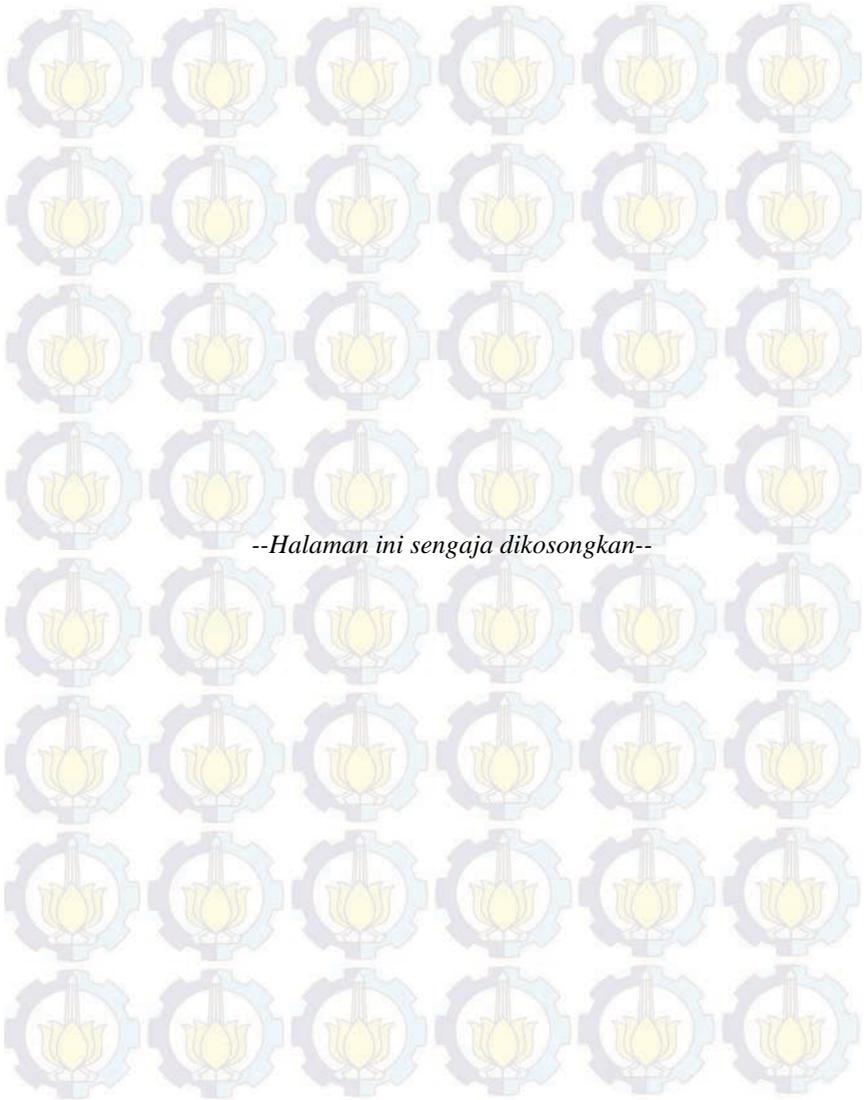
ABSTRACT

Speed control of three phase induction motor was made for motor's rotation that aren't linear caused by dynamic load, noise, and another factor. But there is some disadvantage if motor in dynamic condition where the condition of motor always change, this was caused by some factor that disturb from load and noise. Therefore controller is needed in order plant stable.

In this final project, to keep stabilize the plant, PD Fuzzy plus Integral Controller was used. PD Fuzzy plus Integral Controller is composited by two controller, that is PD Fuzzy and Integral Controller. Advantages of this controller is neglect load, controller able to approach setpoint value, so that model math plant is not needed and also controller have most superior with non-linear system.

Result of experiment between simulation and implementation indicate that result from simulation has value of %error steady state and settling time 5% is 0.044% and 0.3 second whereas result from implementation is 0.015% and 0.11 second.

Keyword : *Fuzzy PD Controller, Labview, Magnetic Load, Speed Motor*



KATA PENGANTAR

Puji syukur atas kehadiran Allah SWT karena berkat rahmat dan hidayah-Nya penulis dapat menyusun dan menyelesaikan laporan Tugas Akhir dengan judul:

Pengaturan Kecepatan Motor Induksi Tiga Fasa dengan Beban Rem Magnetik Menggunakan Metode *Fuzzy PD plus Integral*

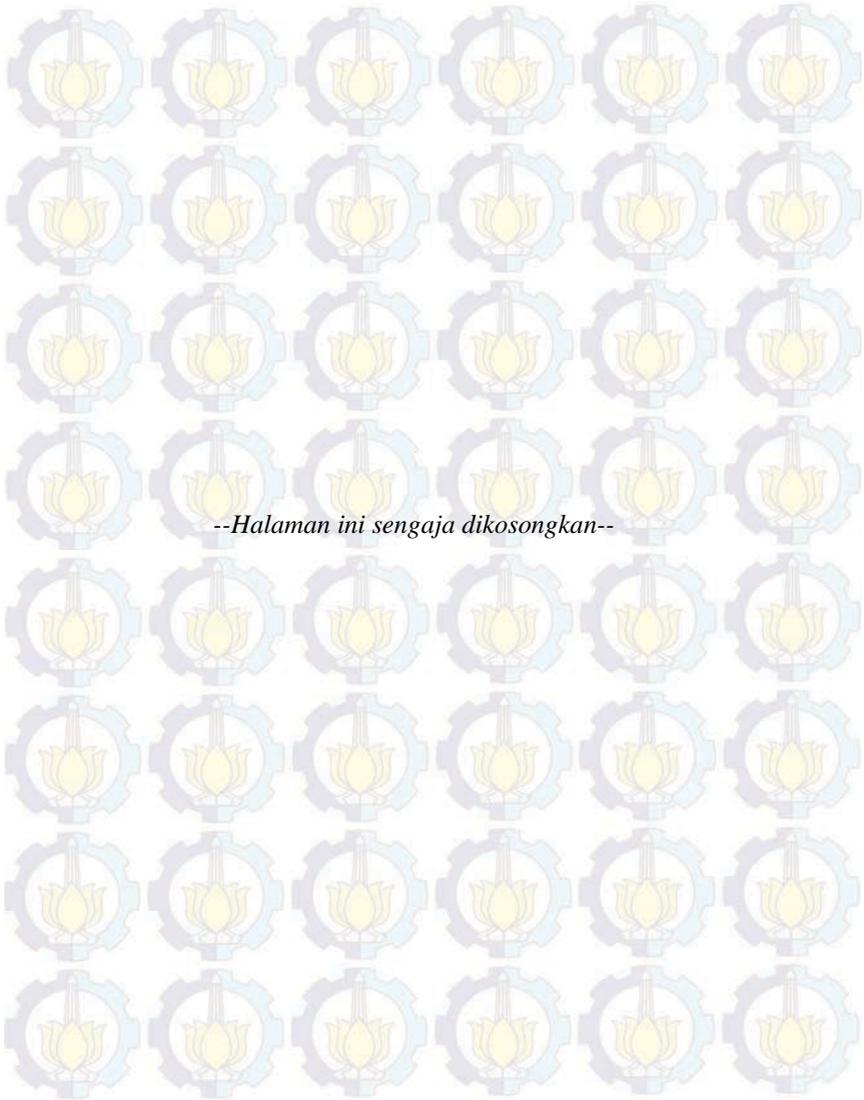
Pada kesempatan kali ini penulis ingin mengucapkan rasa syukur kehadiran Allah SWT yang telah memberikan petunjuk, kemudahan, dan kemurahan-Nya serta tidak lupa ucapan terimakasih yang sebesar-besarnya kepada beberapa pihak yang telah memberikan dukungan selama proses pengerjaan tugas akhir ini, antara lain:

1. Ibu, Bapak, kakak, sahabat, dan teman yang selalu menyemangati, mendukung, dan mendoakan dengan setulus hati tanpa henti,
2. Bapak Ir. Josaphat Pramudijanto, M.Eng. sebagai dosen pembimbing yang telah membantu dan membimbing hingga laporan ini terselesaikan,
3. Bapak dan Ibu penguji Tugas Akhir yang telah memberi masukan kepada penulis untuk menyusun buku ini menjadi lebih baik,
4. Tim Proyek Tugas Akhir, Mas Asep, Mas Fadli, dan Mas Buhari yang telah bahu membahu mengerjakan Tugas Akhir ini,
5. Teman-teman angkatan dan teman-teman Sistem Pengaturan yang telah berbagi suka dan duka, dan
6. Semua pihak yang turut membantu pengerjaan Tugas Akhir ini.

Penulis menyadari dan memohon maaf atas segala kekurangan pada Tugas Akhir ini dikarenakan keterbatasan kemampuan penulis. Penulis mengharapkan berbagai masukan dan kritik dari berbagai pihak agar lebih baik di masa yang akan datang.

Surabaya, Juni 2015

Penulis



--Halaman ini sengaja dikosongkan--

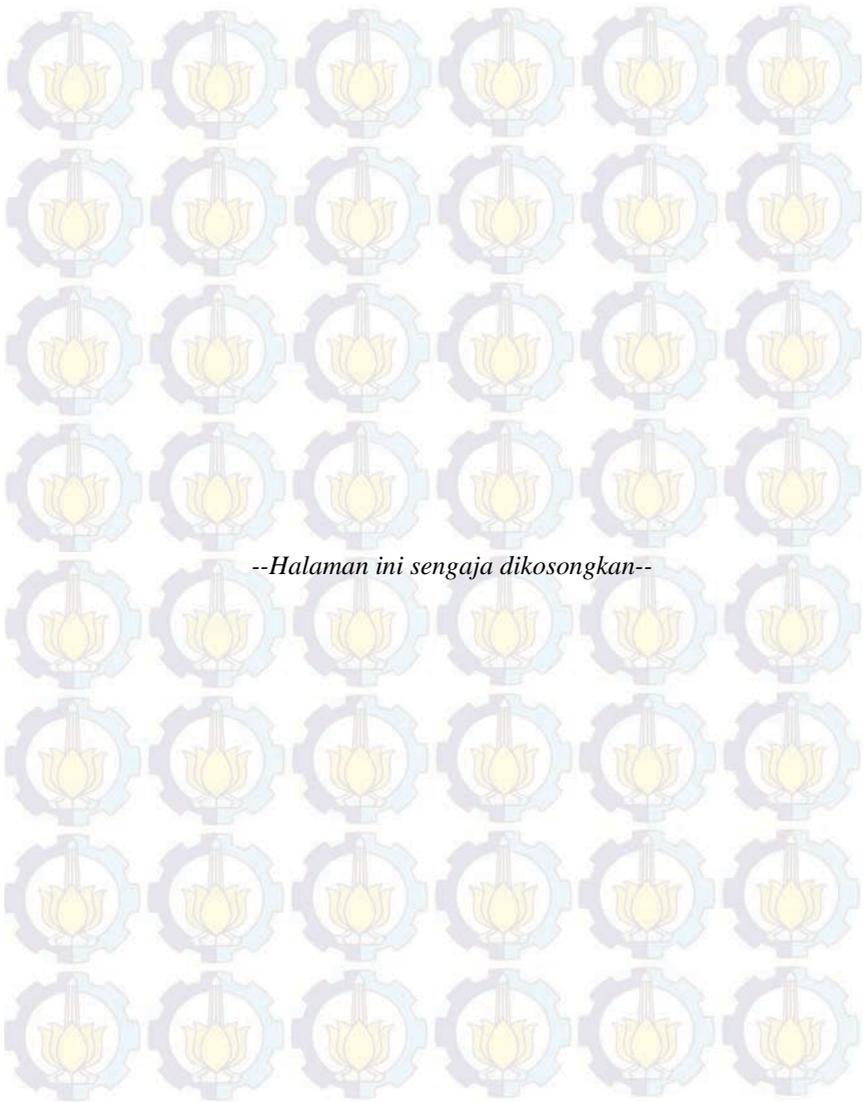
DAFTAR ISI

HALAMAN JUDUL	i
PERNYATAAN KEASLIAN TUGAS AKHIR	iii
HALAMAN PENGESAHAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Perumusan Masalah	2
1.3. Batasan Masalah	2
1.4. Tujuan Penelitian	2
1.5. Sistematika Penulisan	2
1.6. Relevansi	3
BAB II TEORI DASAR	5
2.1. Motor Induksi Tiga Fasa	5
2.2. <i>Inverter (Variable Frequency Drive)</i>	8
2.3. <i>Data Acquisition Module (DAQ)</i>	8
2.4. <i>Rotary Encoder</i>	10
2.5. Rem Magnetik	12
2.6. Sensor Temperatur	12
2.7. Validasi ISE	13
2.8. Labview	13
2.9. Kontroler <i>Fuzzy</i>	14
2.10. Kontroler PD <i>Fuzzy plus</i> Kontroler <i>Integral</i>	16
BAB III PERANCANGAN SISTEM	19
3.1. Perancangan <i>Hardware</i>	19
3.1.1. Perancangan Motor Induksi Tiga Fasa	19
3.1.2. Perancangan <i>Setting Inverter</i>	20
3.1.3. Perancangan <i>Rotary Encoder</i>	22
3.1.4. Perancangan DAQ	24
3.1.5. Desain dan Perancangan Indikator Temperatur	25

3.1.6. Desain dan Perancangan Panel Kontrol	26
3.2. Perancangan <i>Software</i>	27
3.2.1. Perancangan <i>Setting Software</i> DAQ	27
3.2.2. Perancangan Identifikasi Sistem	30
3.3. Perancangan Kontroler pada Labview	30
3.3.1. <i>Fuzzifikasi</i>	30
3.3.2. <i>Rule Base</i>	31
3.3.3. <i>Defuzzifikasi</i>	31
3.4. Perancangan Kontroler PD <i>Fuzzy plus</i> Kontroler <i>Integral</i>	31
3.5. Perancangan Mekanik Keseluruhan	33
BAB IV PENGUJIAN DAN ANALISA	37
4.1. Pengujian Hardware	37
4.1.1. Pengujian DAQ dan <i>Inverter</i>	37
4.1.2. Pengujian <i>Encoder</i>	38
4.1.3. Pengujian Sensor Temperatur	40
4.2. Pengujian Simulasi	41
4.2.1. Pengujian Simulasi Tanpa Kontroler	41
4.2.2. Pengujian Simulasi dengan Kontroler	42
4.3. Implementasi Pengujian Kontroler	43
4.3.1. Pengujian Implementasi Tanpa Kontroler	43
4.3.2. Pengujian Implementasi dengan Kontroler	44
4.3.3. Pengujian Implementasi dengan Beban	45
BAB V PENUTUP	47
5.1. Kesimpulan	47
5.2. Saran	47
DAFTAR PUSTAKA	49
LAMPIRAN

DAFTAR TABEL

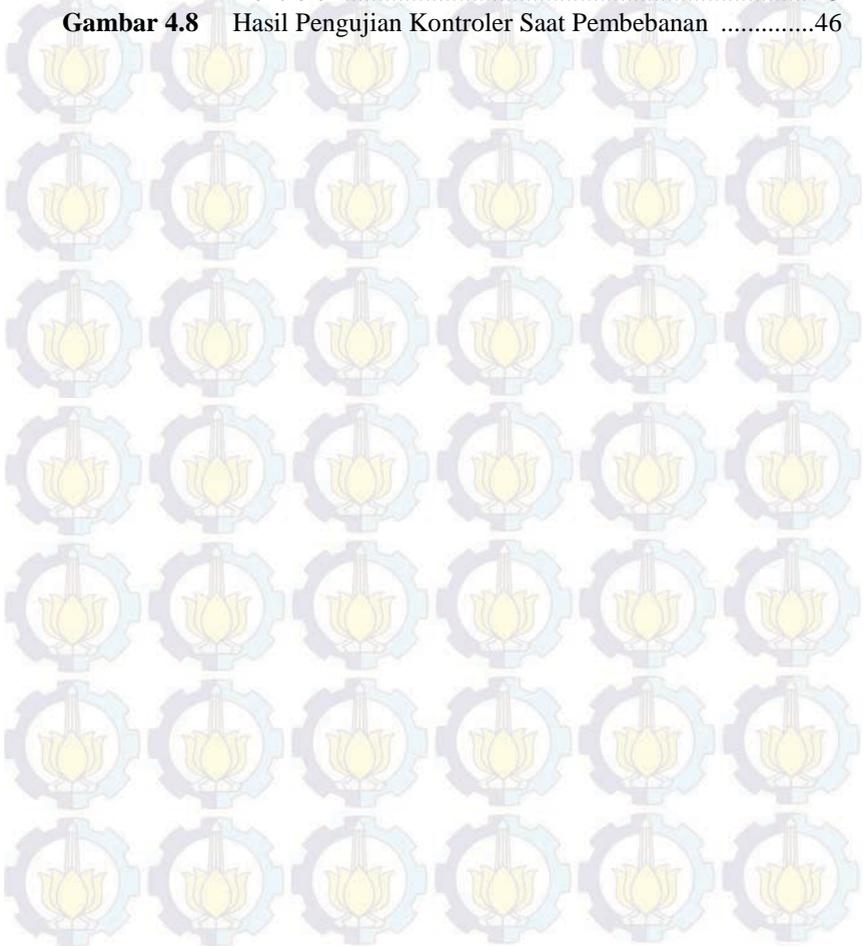
Tabel 2.1 Spesifikasi DAQ USB-4716	10
Tabel 3.1 Deskripsi Konektor Sinyal I/O	25
Tabel 4.1 Hasil Pengujian Antara DAQ, Kecepatan Motor, dan <i>Output Encoder</i>	38
Tabel 4.2 Hasil Pengujian Antara Kecepatan Motor dan <i>Encoder</i>	39
Tabel 4.3 Daerah Kerja Indikator Temperatur	41
Tabel 4.4 Hasil Spesifikasi Respon Transien Simulasi Menggunakan Kontroler	43
Tabel 4.5 Hasil Spesifikasi Respon Transien Sistem Menggunakan Kontroler	45



DAFTAR GAMBAR

Gambar 2.1	Motor Induksi Tiga Fasa	5
Gambar 2.2	Klasifikasi Motor Listrik	6
Gambar 2.3	Stator	6
Gambar 2.4	Jenis – Jenis Rotor	7
Gambar 2.5	<i>Inverter</i> Siemens Micromaster 420	8
Gambar 2.6	DAQ <i>Module</i> USB-4716	9
Gambar 2.7	Konektor I/O USB-4716	9
Gambar 2.8	<i>Rotary Encoder</i>	11
Gambar 2.9	Konstruksi dan <i>Wiring Encoder</i>	11
Gambar 2.10	Rem Magnetik	12
Gambar 2.11	IC Sensor Temperatur LM 35	13
Gambar 2.12	Logo Labview	13
Gambar 2.13	Hasil <i>defuzzifikasi</i> menggunakan metode COA	16
Gambar 2.14	Kontroler <i>Fuzzy PD Plus</i> Kontroler <i>Integral</i>	17
Gambar 3.1	Desain Perancangan Sistem Pengaturan Kecepatan Motor Induksi Tiga Fasa	19
Gambar 3.2	Spesifikasi Motor Induksi Tiga Fasa	20
Gambar 3.3	Desain <i>Rotary Encoder</i>	23
Gambar 3.4	Konfigurasi Instalasi <i>Encoder</i>	23
Gambar 3.5	<i>Wiring</i> antara <i>Encoder</i> dan DAQ	24
Gambar 3.6	Konfiurasi <i>Hardware</i> DAQ	24
Gambar 3.7	Rangkaian Indikator Temperatur	26
Gambar 3.8	Desain Panel Kontrol	26
Gambar 3.9	Langkah membuat <i>Analog Output</i> DAQ pada Labview	29
Gambar 3.10	Langkah Membuat <i>Counter</i> DAQ pada Labview	29
Gambar 3.11	<i>Fuzzifikasi</i> pada Labview	30
Gambar 3.12	<i>Rule Base</i> dan <i>Defuzzifikasi</i> pada Labview	31
Gambar 3.13	Kontroler Simulasi Menggunakan Labview	32
Gambar 3.14	Blok Diagram Kontroler <i>Fuzzy PD plus</i> kontroler <i>Integral</i> pada Labview	33
Gambar 3.15	Front Panel Kontroler <i>Fuzzy PD plus</i> kontroler <i>Integral</i> pada Labview	34
Gambar 3.16	Perancangan Mekanik Keseluruhan	35
Gambar 4.1	Skema Pengujian <i>Inverter</i> dan Kecepatan Motor	37
Gambar 4.2	Skema Pengujian <i>Encoder</i>	39

Gambar 4.3	Hasil Pengujian antara Pengujian Kecepatan Motor dan Encoder	40
Gambar 4.4	Hasil Pengujian Simulasi Tanpa Kontroler	42
Gambar 4.5	Hasil Pengujian Simulasi Menggunakan Kontroler	43
Gambar 4.6	Hasil Pengujian Implementasi Tanpa Kontroler	44
Gambar 4.7	Hasil Pengujian Implementasi Menggunakan Kontroler	45
Gambar 4.8	Hasil Pengujian Kontroler Saat Pembebanan	46

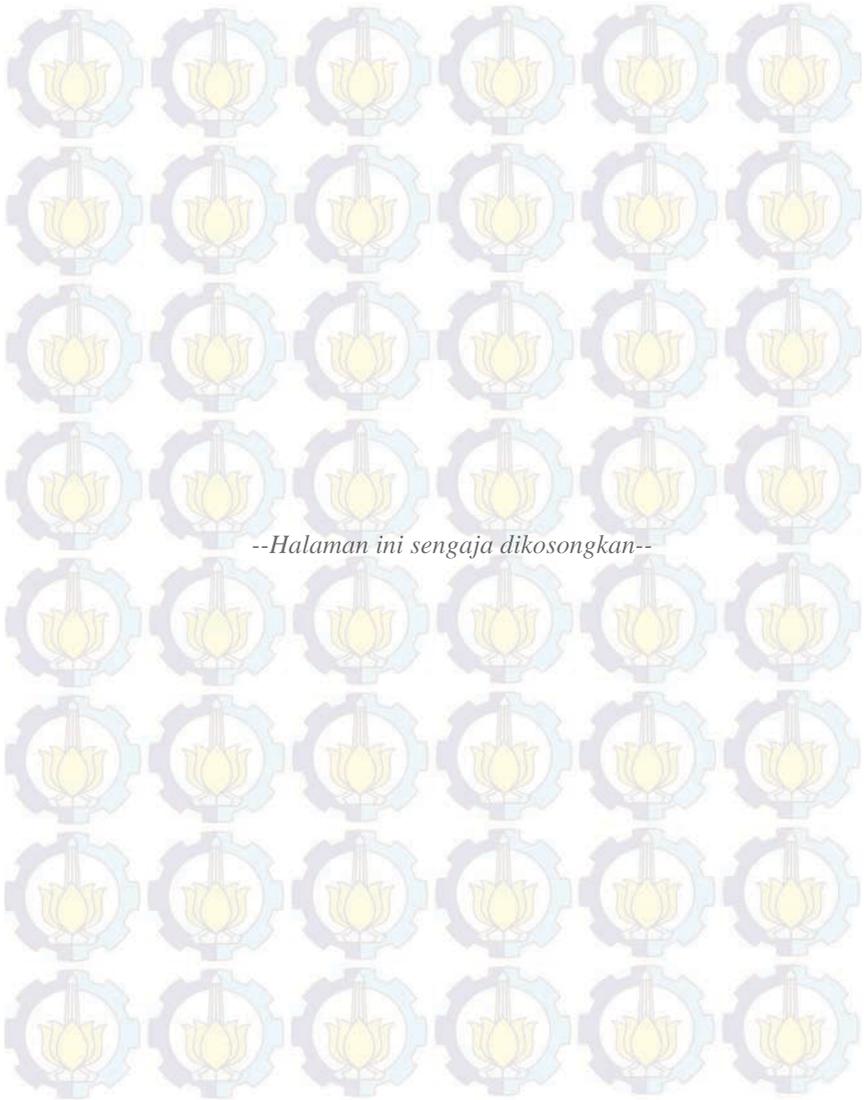


RIWAYAT HIDUP



Dimas Bambang Sulaiman merupakan anak bungsu dari pasangan Sugiyanto dan Darti yang dilahirkan pada tanggal 21 Oktober 1991. Lahir dan besar di Bandung penulis merupakan alumnus dari SDN XIII Cimahi dan SMPN 1 Cimahi. Untuk jenjang SMA penulis menyelesaikan di sekolah SMAN 9 Bandung. Penulis menjalani jenjang kuliah Diploma III Elektronika di Politeknik Negeri Bandung dengan bidang studi yang ditekuni yaitu elektronika. Setelah bekerja di perusahaan swasta selama beberapa bulan, penulis melanjutkan Strata Satu Lintas Jalur di Jurusan Teknik Elektro Institut

Teknologi Sepuluh Nopember (ITS) Surabaya pada tahun 2013. Bidang studi yang ditekuni oleh penulis adalah Teknik Sistem Pengaturan. Pada awal tahun 2015 penulis mengerjakan Tugas Akhir dibawah bimbingan Ir. Josaphat Pramudijanto, M.Eng. dengan judul Pengaturan Kecepatan Motor Induksi Tiga Fasa dengan Beban Rem Magnetik Menggunakan Metode Fuzzy PD Plus Kontroler Integral dan pada bulan juni mengikuti seminar dan ujian Tugas Akhir di Bidang Studi Teknik Sistem Pengaturan Jurusan Teknik Elektro ITS Surabaya sebagai salah satu persyaratan untuk memperoleh gelar Sarjana Teknik Elektro. Email penulis adalah : dimas_b_s@yahoo.com.



--Halaman ini sengaja dikosongkan--

BAB I

PENDAHULUAN

Bab ini menjelaskan latar belakang, perumusan masalah, batasan masalah, dan tujuan dari penelitian pada Tugas Akhir ini. Selain itu, dijelaskan juga mengenai sistematika dalam penyelesaian Tugas Akhir, serta relevansi untuk penelitian selanjutnya.

1.1 Latar Belakang

Dalam suatu proses di industri, motor induksi tiga fasa merupakan komponen *vital* yang memiliki banyak kegunaan. Motor induksi adalah sebuah motor asinkron AC yang memiliki kelebihan seperti mekanik yang tangguh, konstruksi yang sederhana, dan perawatan yang mudah. Motor induksi seringkali diaplikasikan pada pompa, kipas angin, AC, pabrik kertas dan tekstil. Aplikasi ini membutuhkan sensor kecepatan seperti *shaft encoder* untuk pengaturan kecepatan.

Karakteristik kecepatan terhadap torsi untuk motor induksi yang non-linear, menyulitkan operator untuk dapat mengontrol kecepatan motor induksi tiga fasa. Sehingga dirancanglah suatu kontrol yang dapat mengontrol motor tiga fasa secara efektif.

Pengaturan kecepatan motor induksi tiga fasa dibuat untuk mengatur putaran motor yang tidak linear, yang diakibatkan oleh gangguan beban, *noise*, dan banyak faktor lainnya. Tetapi ada beberapa kekurangan apabila motor dalam keadaan dinamis dimana keadaan motor dapat berubah secara terus-menerus, hal ini karena pengaruh dari beberapa faktor yang mengganggu dari beban dan *noise*. Sehingga dibutuhkan kontroler yang mampu menjaga *plant* agar tetap stabil. Maka dari itu, untuk menjaga kestabilan *plant* digunakan kontroler *Fuzzy PD plus* kontroler *Integral*. Perubahan *error* dan $\Delta error$ *plant*, akan diproses oleh logika *Fuzzy PD* yang lalu akan ditambahkan dengan kontroler *integral*.

Penggunaan *software* LabView diharapkan dapat menjadi *software* pengendali utama antara DAQ dan *plant*. *Software* LabView dipilih karena LabView memiliki kapasitas pemrograman yang besar, mampu berkomunikasi dengan DAQ, dan mampu menampilkan kondisi *plant* secara *real-time*.

1.2 Perumusan Masalah

Pada aplikasi seperti *conveyor*, ketika diberi beban berubah-ubah, kecepatan motor pun ikut berubah-ubah. Apabila kecepatan *conveyor* berubah-ubah, maka waktu yang dibutuhkan *plant* untuk memproduksi produk akan berbeda. Untuk menjaga kecepatan motor dalam Tugas Akhir ini diusulkan sebuah pendekatan.

1.3 Batasan Masalah

Dari perumusan masalah yang sudah dipaparkan, diperlukan batasan masalah agar penelitian dapat tercapai. Batasan masalah dari penelitian tersebut sebagai berikut:

1. Daya magnet yang digunakan oleh rem magnetik tidak diperhitungkan secara matematis. Pembebanan diberikan dengan menggunakan rem magnetik yang dieksitasi tegangan dari *variac*.
2. Penelitian ini merupakan modifikasi dari penelitian sebelumnya dan dikerjakan secara berkelompok dengan kesepakatan bahwa titik kerja anggota kelompok berbeda-beda.
3. Penambahan indikator temperatur merupakan hasil modifikasi dari penelitian ini.
4. Model matematis *plant* dibutuhkan hanya untuk simulasi, kontroler *Fuzzy PD plus* kontroler *Integral* tidak membutuhkan model matematis *plant*.
5. Proses mendapatkan parameter kontrol diperoleh menggunakan metode *manual tuning*.

1.4 Tujuan Penelitian

Untuk mengatasi permasalahan yang ada, pada Tugas Akhir ini dirancang dan diimplementasi sistem pengaturan motor induksi tiga fasa.

1.5 Sistematika Penulisan

Sistematika penulisan dalam penelitian ini terdiri dari lima bab yaitu pendahuluan, teori dasar, perancangan sistem, pengujian dan analisa, dan penutup.

BAB I PENDAHULUAN

Pada bab ini, akan dibahas tentang latar belakang, perumusan masalah, batasan masalah, tujuan penelitian, sistematika penulisan dan relevansi.

BAB II TEORI PENUNJANG

Dasar pemikiran dan pengetahuan dari sistem yang akan dirancang seperti teori motor induksi tiga fasa, *Data Acquisition Module*, *inverter*, *rotary encoder*, rem magnetik, sensor temperatur, validasi ISE, Labview, kontroler *Fuzzy* dan kontroler *Fuzzy PD plus* kontroler *Integral* akan dibahas pada bab ini.

BAB III PERANCANGAN SISTEM

Bab ini membahas tentang perancangan sistem yang akan dibangun, yang terbagi menjadi perancangan perangkat keras (*hardware*), perancangan perangkat lunak (*software*), identifikasi sistem, dan perancangan simulasi dan implementasi kontroler PD *Fuzzy plus* kontroler *Integral*.

BAB IV PENGUJIAN DAN ANALISA

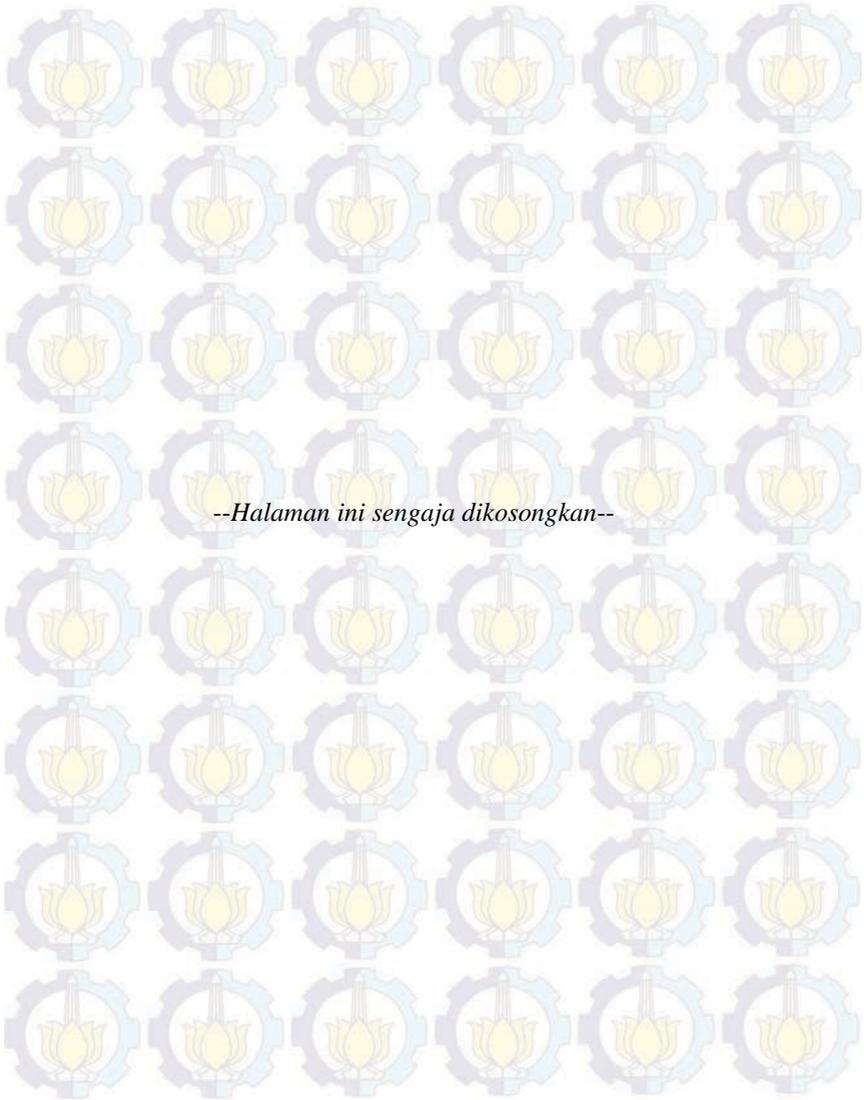
Bab ini membahas pengujian perangkat keras (*hardware*), perangkat lunak (*software*), pengujian simulasi dan implementasi, dan analisa sistem secara keseluruhan.

BAB V PENUTUP

Membahas tentang kesimpulan akhir dari penelitian ini dan saran-saran untuk kemungkinan pengembangan penelitian lebih lanjut.

1.6 Relevansi

Hasil yang diperoleh dari Tugas Akhir ini diharapkan dapat menjadi referensi dalam pengaturan kecepatan motor induksi tiga fasa menggunakan DAQ dan LabView. Selain itu penelitian ini dapat menjadi perbandingan metode pengaturan kecepatan motor induksi tiga fasa dimasa yang akan datang.



BAB II TEORI PENUNJANG

Bab ini berisi teori dasar yang digunakan dalam penyelesaian dari Tugas Akhir ini. Beberapa hal yang dibahas pada Bab ini meliputi motor induksi tiga fasa, *Inverter*, *Data Acquisition Module*, *Rotary Encoder*, Rem Magnetik, Sensor Temperatur, Validasi ISE, LabVIEW, dan Kontroler *Fuzzy PD plus Integral*.

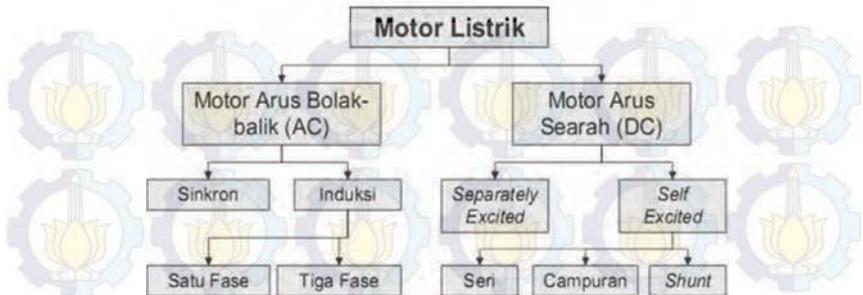
2.1 Motor Induksi Tiga Fasa[1]

Motor memiliki dua jenis arus penggerak yaitu motor AC (*Alternating Current*) dan motor DC (*Direct Current*). Motor AC adalah motor listrik yang digerakkan oleh arus bolak-balik. Jadi perbedaan utama motor AC dan motor DC adalah sumber arusnya. Motor AC dibedakan menjadi 2 jenis, yaitu motor sinkron dan motor induksi/motor asinkron.



Gambar 2.1 Motor Induksi Tiga Fasa

Motor sinkron didefinisikan sebagai motor yang memiliki *output* kecepatan putaran motornya yang sinkron/sebanding (tanpa *slip*) dengan frekuensi listrik yang masuk ke stator. Sedangkan motor induksi didefinisikan sebagai motor yang memiliki *output* kecepatannya asinkron/tidak sebanding dengan frekuensi yang masuk ke stator. Klasifikasi motor listrik untuk lebih jelasnya ditunjukkan pada Gambar 2.2.



Gambar 2.2 Klasifikasi Motor Listrik

2.1.1 Konstruksi Motor Induksi Tiga Fasa

Motor induksi memiliki dua bagian utama, yaitu :

a. Stator

Stator merupakan bagian yang tidak bergerak (stasioner/statis) pada motor induksi dan umumnya berada pada sisi luar motor induksi. Gambar stator ditunjukkan pada Gambar 2.3. Stator mempunyai beberapa bagian, yaitu :

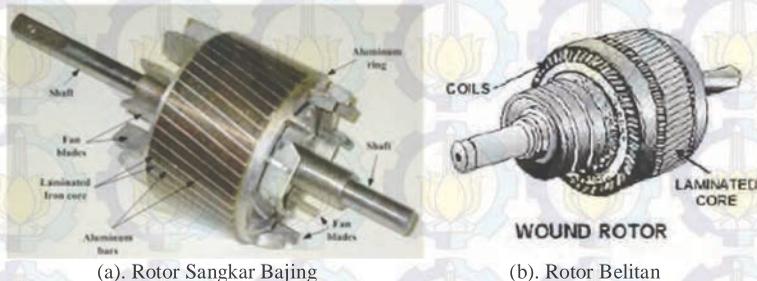
- Rumah stator, fungsinya sebagai penopang dan sebagai pelindung bagian dalam mesin.
- Inti stator, terbuat dari laminasi logam yang disusun berlapis.
- Kumputan stator, terbuat dari tembaga.



Gambar 2.3 Stator

b. Rotor

Rotor adalah bagian dari motor induksi yang mengalami perputaran yang berada pada bagian tengah konstruksi motor. Perputaran rotor disebabkan karena adanya medan magnet dan lilitan kawat pada rotor. Sedangkan torsi dari perputaran rotor ditentukan oleh banyaknya lilitan dan juga diameter kawat.



Gambar 2.4 Jenis – Jenis Rotor

Berdasarkan konstruksinya rotor dapat diklasifikasikan menjadi dua jenis yaitu:

- Rotor Sangkar Baging (*Squirrel Cage*)
Terdiri dari batang penghantar tebal yang dilekatkan dalam petak-petak slots paralel. Batang-batang tersebut diberi hubungan pendek pada kedua ujungnya dengan alat cincin hubungan pendek.
- Rotor Belitan (*Wound Rotor*)
Rotor yang memiliki gulungan tiga fase, lapisan ganda dan terdistribusi. Dibuat melingkar sebanyak kutub stator. Tiga fase digulungi kawat pada bagian dalamnya dan ujung yang lainnya dihubungkan ke cincin kecil yang dipasang pada batang as dengan sikat yang menempel padanya.

2.1.2 Prinsip Kerja Motor Induksi Tiga Fasa

Pada motor induksi tidak terdapat hubungan listrik antara stator dengan rotor, karena arus pada rotor merupakan arus induksi. Listrik diberikan ke stator yang akan menghasilkan medan magnet. Medan magnet ini bergerak dengan kecepatan sinkron disekitar rotor. Arus rotor

menghasilkan medan magnet kedua, yang berusaha untuk melawan medan magnet stator, yang menyebabkan rotor berputar. Walaupun begitu, didalam praktiknya motor tidak pernah bekerja dengan kecepatan sinkron namun pada kecepatan dasar yang lebih rendah. Terjadinya perbedaan antara dua kecepatan tersebut disebabkan adanya *slip*/pergeseran yang meningkat dengan meningkatkan beban.

2.2 *Inverter (Variable Frequency Drive)*[2]

Inverter atau yang bisa disebut *Variable Speed Drive* atau *Variable Frequency Drive*, adalah rangkaian elektronika daya yang digunakan untuk mengkonversikan tegangan searah (DC) ke suatu tegangan bolak-balik (AC). Terdapat dua jenis *inverter* berdasarkan jenis sumber daya dan topologi dari rangkaian sumbernya, yaitu *Voltage Source Inverter (VSI)* dan *Current Source Inverter (CSI)*. Hasil *output* dari *inverter* berupa tegangan dan frekuensi yang dapat diatur, maka *inverter* dapat diaplikasikan sebagai pengatur kecepatan motor induksi. Gambar *inverter* ditunjukkan pada Gambar 2.5.



Gambar 2.5 *Inverter* Siemens Micromaster 420

Prinsip kerja dari *inverter* yaitu tegangan yang masuk dari jala-jala 50 Hz dialirkan ke *rectifier*/penyearah DC dan ditampung ke kapasitor tegangan besar. Kemudian *output* DC tersebut dikembalikan lagi ke *inverter* untuk dijadikan tegangan AC kembali dengan frekuensi sesuai kebutuhan.

2.3 *Data Acquisition Module (DAQ)*[3]

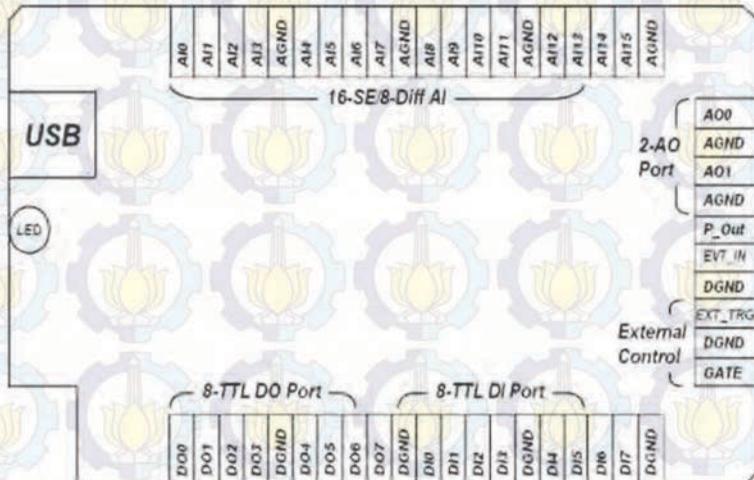
Data Acquisition atau DAQ adalah produk dan atau proses pengambilan contoh data yang mengukur fenomena fisik atau elektrik

kemudian mengubahnya ke dalam nilai digital yang dapat disimulasi dengan menggunakan komputer.



Gambar 2.6 DAQ Module USB-4716

DAQ yang digunakan pada penelitian kali ini adalah DAQ Advantech Module USB-4716. USB-4716 mempunyai 1 port USB, 16 port Analog Input, 8 port Digital Output, 8 port Digital Input, 2 Analog Output dan 3 port External Control. Beberapa spesifikasi pada DAQ bisa dilihat pada Tabel 2.1. Untuk tampilan luar DAQ bisa dilihat pada Gambar 2.6, sedangkan untuk tampilan dalam bisa dilihat pada Gambar 2.7.



Gambar 2.7 Konektor I/O USB-4716

Beberapa kelebihan menggunakan DAQ yaitu :

- Mudah digunakan.
- DAQ mendapat daya *input* dari *port* USB sehingga tidak membutuhkan daya *input* dari komponen lain.
- Mampu bekerja dengan *software* MATLAB dan LabView.
- Tidak mudah terkena sinyal *noise*.
- Memiliki kemampuan akurasi yang tinggi.

Tabel 2.1 Spesifikasi DAQ USB-4716

Spesifikasi DAQ USB-4716		
Analog Input	Channels	16 SE/8 Diff
	Sampling Rates	200 kS/s
	Resolution	16 bits
	Unipolar Input (V)	0 ~ 10, 0 ~ 5, 0 ~ 2.5, 0 ~ 1.25
	Bipolar Inputs (V)	$\pm 10, \pm 5, \pm 2.5, \pm 1.25, \pm 0.625$
Analog Output	Resolution	16 bits
General	Power Consumption	Typical: 5V @ 360 mA; Max.: 5 V @ 450 mA
	I/O Connectors	Onboard screw terminal
	Dimensions	132 x 80 x 32 mm (5.2" x 3.15" x 1.26")
	Bus	USB 2.0
Timer/Counter	Resolution	32 bits
	Time Base	1 kHz
	Channels	1

2.4 Rotary Encoder[4]

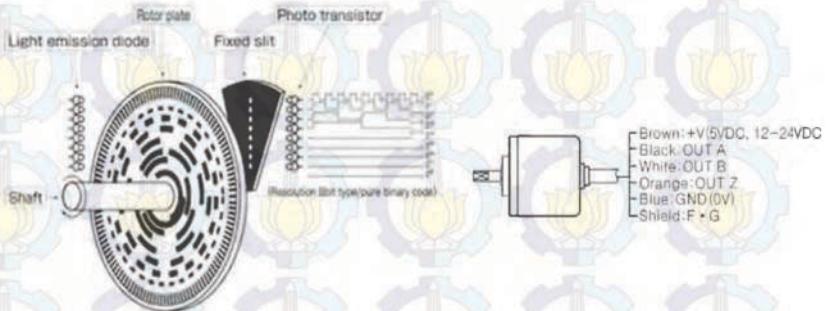
Rotary Encoder, bisa disebut juga *shaft encoder*, adalah suatu komponen elektromekanik yang memiliki fungsi untuk memonitoring posisi angular pada suatu poros yang berputar. Dari perputaran poros tersebut data yang termonitoring akan diubah ke dalam bentuk data digital oleh *encoder* berupa lebar pulsa yang kemudian akan dihubungkan ke kontroler (Mikrokontroler/PLC). Data yang di dapat

berupa posisi angular (sudut) kemudian dapat diolah oleh kontroler sehingga mendapatkan data berupa kecepatan, arah, dan posisi dari perputaran porosnya. Tampilan *rotary encoder* ditunjukkan pada Gambar 2.8.



Gambar 2.8 *Rotary Encoder*

Gambar 2.9 merupakan konstruksi dan *wiring* dari *encoder*. Konstruksi *encoder* berupa piringan tipis yang di kopel dengan poros yang berputar, umumnya di kopel langsung dengan *shaft* motor. Terdapat lubang di sepanjang pinggir lingkarannya pada piringan tipis tersebut. Di bagian sisi-sisi piringan terdapat sebuah led dan phototransistor di bagian bersebrangan.



(a). Konstruksi *Encoder*

(b). *Wiring Encoder*

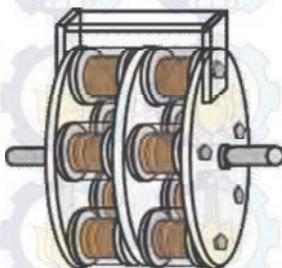
Gambar 2.9 Konstruksi dan *Wiring Encoder*

Fungsi dari lubang-lubang yang berada di sepanjang pinggir lingkaran tersebut akan menghantarkan cahaya led ke phototransistor, sebaliknya jika cahaya led tidak menembus lubang piringan maka cahaya akan tertahan. Piringan tersebut akan berputar sesuai dengan kecepatan putaran motor sehingga phototransistor akan saturasi ketika cahaya led menembus lubang-lubangnya. Semakin banyak lubang yang berada pada piringan tentu saja semakin banyak pulsa yang dihasilkan selama satu putaran, hal tersebut berbanding lurus dengan tingkat akurasi yang dihasilkan oleh *encoder*.

2.5 Rem Magnetik [5]

Rem magnetik menghentikan atau memperlambat pergerakan dengan daya elektromagnet. Biasanya rem magnetik diaplikasikan pada kereta, industri, dan aplikasi robotik.

Cara kerja rem magnetik yaitu ketika kumparan kawat diberikan arus listrik, maka kumparan akan menjadi magnet, sehingga dapat mengurangi kecepatan putaran piringan besi yang dikopel dengan motor. Tampilan rem magnetik ditunjukkan pada Gambar 2.10.



Gambar 2.10 Rem Magnetik

2.6 Sensor Temperatur

Sensor temperatur adalah suatu komponen yang dapat mengubah besaran temperatur menjadi besaran elektrik. Terdapat beberapa tipe sensor temperatur, seperti *Resistance Temperature Detector*, *Thermocouple*, Thermistor, dan IC Temperatur. IC Temperatur LM 35 dipilih karena karakter IC memiliki sifat linearitas yang paling tinggi dibanding dengan sensor temperatur yang lain.



Gambar 2.12 Logo LabVIEW

Program-program yang menggunakan satu atau lebih fungsi-fungsi LabVIEW disebut sebagai instrumen virtual. Setiap instrumen virtual mempunyai sebuah panel depan dan sebuah diagram. Panel depan dapat dipandang sebagai penggambaran panel depan dari sebuah instrumen berikut instrumen kontrol dan penampil.

LabVIEW dapat diintegrasikan dengan beberapa *software* dan *hardware* dalam teknologi komputer terbaru secara efektif. LabVIEW seringkali digunakan untuk akuisisi data, kontrol instrumentasi, dan otomasi industri.

2.9 Kontroler Fuzzy[9][10]

Kontroler logika *Fuzzy* adalah suatu proses perhitungan untuk mendapatkan sinyal kontrol yang diinginkan melalui himpunan *fuzzy* meliputi *Fuzzifikasi*, *Rule Base*, *Inference System*, dan *Defuzzifikasi*. Logika *fuzzy* sendiri adalah pernyataan logika yang digunakan untuk mengungkapkan keadaan tersebut menjadi samar dan tidak memiliki batasan yang jelas terhadap suatu himpunan. Konsep logika *fuzzy* menggantikan konsep “benar-salah” dari logika *boolean* menjadi derajat tingkat kebenaran. Dimana logika *fuzzy* lebih baik daripada logika *boolean* karena logika *fuzzy* lebih mendekati cara berfikir manusia. Teori *fuzzy* menyatakan keanggotaan suatu objek ke dalam fungsi derajat keanggotaan (*membership function*).

2.9.1 Fuzzifikasi

Proses pertama pada sistem kontroler *fuzzy* yaitu *fuzzifikasi*. *Fuzzifikasi* adalah proses pemetaan masukan dan keluaran sistem agar sesuai dengan himpunan *Fuzzy*. Transformasi ini bertujuan agar *input* kontroler *fuzzy* bisa dipetakan menuju jenis yang sesuai dengan himpunan *fuzzy*. Pemetaan digunakan dengan cara yang disebut fungsi keanggotaan *fuzzy* (*membership function*).

Fungsi keanggotaan yang digunakan adalah fungsi segitiga dimana fungsi segitiga memiliki tiga parameter $\{a,b,c\}$ yang didefinisikan pada Persamaan 2.1. Parameter fungsi keanggotaan segitiga ditunjukkan dengan $\{a, b, c\}$ (dengan $a < b < c$) yang menentukan koordinat x dari ketiga sudut segitiga tersebut.

$$\mu_{\text{segitiga}}(x, a, b, c) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 0, & c \leq x \end{cases} \quad (2.1)$$

2.9.2 Aturan Dasar (*Rule Base*)

Aturan dasar *Fuzzy* adalah kaidah dasar yang berisi aturan-aturan secara linguistik yang menunjukkan kepakaran terhadap *plant*. Jumlah basis aturan dari suatu sistem *fuzzy* ditentukan dari jumlah variabel *input* dan jumlah *membership function* pada variabel masukkan. Salah satu format umum untuk menunjukkan suatu kepakaran dalam suatu keputusan adalah aturan *IF-THEN*.

2.9.3 Inference System

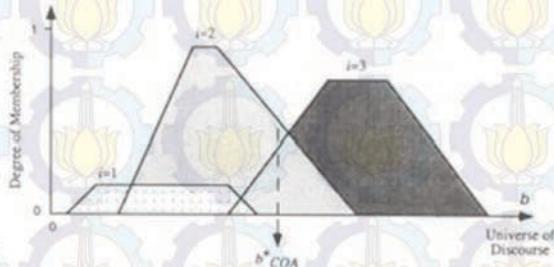
Fuzzy Inference System adalah sebuah proses formulasi pemetaan masukan terhadap keluaran dengan menggunakan logika *Fuzzy*. Proses dari inferensi *Fuzzy* melibatkan fungsi keanggotaan operator logika *Fuzzy*, dan aturan *IF-THEN*.

Terdapat dua metode inferensi yang paling dikenal yaitu metode inferensi Mamdani dan metode inferensi Takagi-Sugeno. Metode Mamdani menggunakan operasi *Max-Min* atau *Max-product* dalam memperoleh keluaran. Sedangkan pada metode Takagi-Sugeno kesimpulan logika *fuzzy* berupa suatu persamaan.

2.9.4 Defuzzifikasi

Defuzzifikasi merupakan proses untuk mengubah kembali variabel *Fuzzy* menjadi variabel nyata, atau dengan kata lain aksi pengaturan *Fuzzy* yang masih berupa himpunan, dirubah menjadi nilai nyata yang berupa nilai tunggal. Terdapat beberapa metode dalam proses defuzzifikasi yaitu *Center of Area* (COA), *Center of Sum* (COS), *Center*

of Maximum (COM), dan Mean of Maximum (MOM). Metode yang digunakan dalam proses defuzzifikasi penelitian ini adalah Center of Area (COA).



Gambar 2.13 Hasil defuzzifikasi menggunakan metode COA

Metode *Center Of Area* (COA) digunakan untuk menentukan titik tengah area yang merupakan titik pusat massa dari kombinasi fungsi-fungsi keanggotaan. Secara umum persamaan pada metode *Center of Area* (COA) terdapat pada Persamaan 2.2.

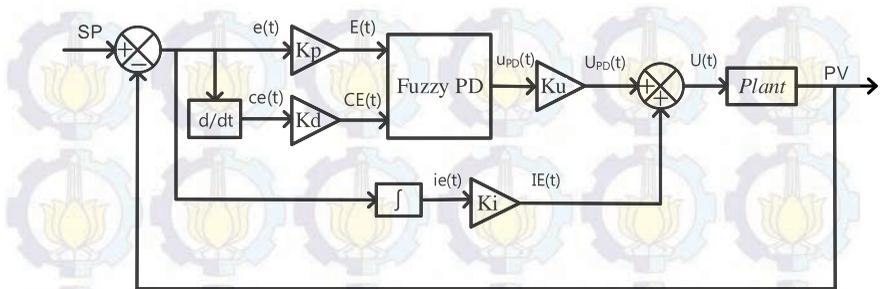
$$COA = \frac{\int f(x) \cdot x dx}{\int f(x) dx} \quad (2.2)$$

2.10 Kontroler Fuzzy PD plus Kontroler Integral[11]

Kontroler Fuzzy PD plus kontroler Integral merupakan metode yang menggabungkan antara kontroler Fuzzy PD dan kontroler Integral. Kontroler ini membutuhkan masukan sinyal *error* dan sinyal $\Delta error$, kontroler Fuzzy PD akan menghasilkan sinyal kontrol melalui proses *learning* untuk mempercepat sistem mencapai keadaan *steady state* kemudian sinyal kontrol *integral* akan mengurangi nilai error dan mempertahankan kondisi *steady state* tersebut.

Persamaan matematis kontroler Fuzzy PD plus kontroler Integral ditunjukkan pada Persamaan 2.3. Sedangkan diagram blok kontroler yang digunakan pada Tugas Akhir ini ditunjukkan pada Gambar 2.14.

$$U(t) = ((K_p * e(t)) + (K_d * ce(t))) * K_u + K_i * ie \quad (2.3)$$



Gambar 2.14 Kontroler *Fuzzy PD Plus* Kontroler *Integral*

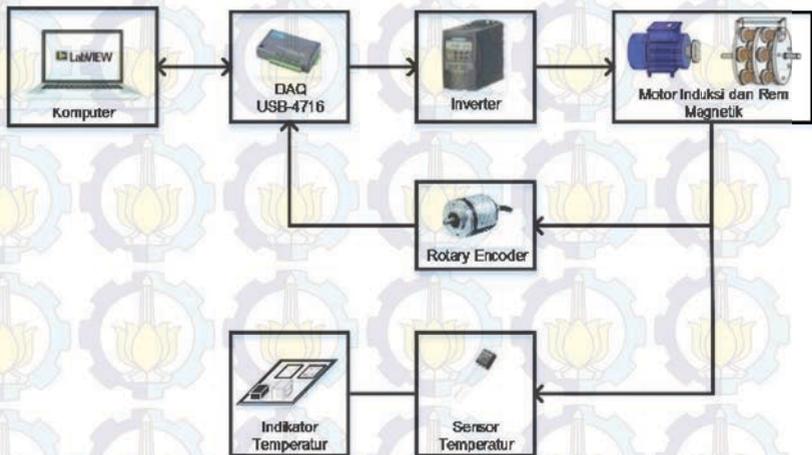


BAB III PERANCANGAN SISTEM

Pada bab ini dijelaskan mengenai perancangan sistem pengaturan kecepatan motor induksi tiga fasa menggunakan kontroler *Fuzzy PD plus* kontroler *Integral* dengan *software* LabVIEW, yaitu perancangan *hardware*, perancangan *software*, perancangan simulasi, dan perancangan mekanik keseluruhan.

3.1 Perancangan Hardware

Perancangan *hardware* (perangkat keras) dari Tugas Akhir ini diantaranya merancang motor induksi dan rem magnetik, *setting inverter*, DAQ, *encoder*, dan *hardware* pendukung lainnya. Berikut akan dijelaskan mengenai *hardware* yang dirancang.



Gambar 3.1 Desain Perancangan Sistem Pengaturan Kecepatan Motor Induksi Tiga Fasa

3.1.1 Perancangan Motor Induksi Tiga Fasa

Motor induksi yang digunakan adalah motor induksi yang sudah digunakan pada Tugas Akhir sebelumnya. Motor induksi tiga fasa ini merupakan produk dari Alliance – Italy dengan tipe IEC - 34 CE. Motor

induksi ini dihubungkan dengan *inverter* sebagai *driver* atau pengendali. *Inverter* akan merubah arus DC dari DAQ menjadi arus AC tiga fasa untuk mengatur putaran motor induksi tiga fasa. Gambar 3.2 merupakan spesifikasi dari motor induksi yang digunakan.

Alliance - Italy			IEC 34 - CE	
TYPE AY 638 - 4			No. 02030688	
0,18 KW	0,25 HP	1,07 / 0,62 A		
220/380 V	1310 r/min	LW52 dB(A)		
CONN. Δ / Y	PROT GRADE IP 55	50 Hz	4,7 Kg	
JB / T8680, 1-1998	Work Rule S1	INS. CLASS F	DATE 02	

Gambar 3.2 Spesifikasi Motor Induksi Tiga Fasa

3.1.2 Perancangan *Setting Inverter*

Pada Tugas Akhir ini digunakan *Inverter* Siemens Micromaster 420 (MM420) sebagai pengendali frekuensi untuk kecepatan putar motor induksi tiga fasa. Untuk dapat menggunakan *inverter* diperlukan pemberian *setting* parameter *inverter* yang dipilih sesuai dengan batas yang diizinkan operasi motor.

Berikut langkah-langkah dalam proses *Quick Commissioning* :

1. P003 (*User Access Level*)

P003 adalah kode untuk akses pengguna. Untuk mengakses parameter dapat dilakukan dengan menekan tombol “P”, kemudian akan muncul 3 pilihan yang terdapat dalam pengaturan ini. Pilih “1”, kemudian menekan kembali tombol “P” untuk menyimpan nilai parameter.

2. P0010 (*Start Quick Commissioning*)

P0010 (kode untuk memulai *Quick Commissioning*) adalah pengaturan untuk memulai mengubah parameter dari *inverter* sesuai dengan karakteristik motor. Dengan cara menekan tombol “P” yang ada di *inverter*, maka untuk memulai *Quick Commissioning* dengan memilih “1”, untuk menyimpan parameter yang sudah diatur dengan cara menekan tombol “P” pada *inverter*.

3. P0100 (*Operation for Europe / America*)

P0100 (kode untuk meengoperasikan mesin buatan Eropa/Amerika) adalah parameter untuk memilih frekuensi operasi yang akan digunakan untuk menggerakkan motor. Ada 3

pilihan dalam pengaturan ini. Karena frekuensi di Indonesia sendiri hanya menyediakan jaringan listrik dengan frekuensi 50 Hz, sehingga diharuskan memilih opsi “0”.

4. P0304 (*Rated Motor Voltage*)

P0304 (kode untuk menentukan nilai *supply* tegangan motor) adalah parameter untuk menentukan nilai *supply* tegangan motor. Besar nominal tegangan motor yang tertera pada *nameplate* adalah 380 V, sehingga nilai yang harus diisi adalah 380.

5. P0305 (*Rated Motor Current*)

P0305 (kode untuk menentukan nilai *supply* arus) adalah parameter untuk nilai arus nominal dari motor. Nilai arus nominal yang ada pada *nameplate* motor yaitu 0,62 A.

6. P0307 (*Rated Motor Power*)

P0307 (kode untuk menentukan nilai *supply* daya) adalah parameter yang menentukan nilai daya motor. Besar nominal daya motor yang tertera pada *nameplate* yaitu 0,18 kW.

7. P0310 (*Rated Motor Frequency*)

P0310 (kode untuk menentukan nilai *supply* frekuensi) adalah parameter yang menentukan nilai frekuensi motor. Pada bagian ini kisaran yang diperbolehkan 12 – 650 Hz. Besar nominal frekuensi motor yang tertera pada *nameplate* yaitu 50 Hz.

8. P0311 (*Rated Motor Speed*)

P0311 (kode untuk menentukan nilai *supply* kecepatan motor) adalah pengaturan untuk menentukan nilai kecepatan motor. Besar nominal kecepatan motor (rpm) pada *nameplate* yaitu 1310 rpm.

9. P0700 (*Selection of Command Source*)

P0700 (kode untuk menentukan sumber perintah) adalah parameter untuk pemilihan sumber perintah, dimana nantinya akan muncul tiga pilihan, karena semua pengaturan berasal dari *inverter* itu sendiri tanpa memerlukan perangkat lain maka memilih angka “1” *Basic Operator Panel*.

10. P1000 (*Selection of Frequency Setpoint*)

P1000 (kode untuk menentukan setpoint frekuensi) adalah parameter untuk menentukan pengontrolan frekuensi *inverter*. Untuk pengendalian frekuensinya menggunakan pilihan “2”, yaitu *analog setpoint*.

11. P1080 (*Minimum Frequency*)

P1080 (kode untuk menentukan nilai minimum frekuensi) adalah parameter untuk menentukan nilai minimal frekuensi motor dengan kisaran frekuensi 0 – 650 Hz. Motor yang digunakan diatur minimal frekuensi motor sebesar 0 Hz.

12. P1082 (*Maximum Frequency*)

P1082 (kode untuk menentukan nilai maksimum frekuensi) adalah parameter untuk menentukan nilai maksimum frekuensi motor dengan kisaran sebesar 0 – 650 Hz, dimana motor yang digunakan frekuensinya diatur maksimal sebesar 50 Hz.

13. P1120 (*Ramp-up Time*)

P1120 (kode untuk menentukan nilai *Ramp-up time*) adalah parameter untuk menentukan nilai *ramp-up time*. *Ramp-up time* adalah waktu yang dibutuhkan oleh motor dari keadaan diam sampai frekuensi motor maksimum. Waktu yang dibutuhkan untuk mencapai frekuensi motor maksimum adalah sebesar 4s.

14. P1121 (*Ramp-down time*)

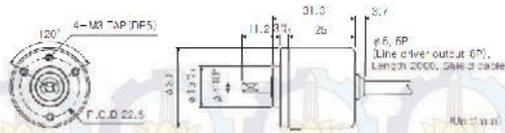
P1121 (kode untuk menentukan nilai *Ramp-down time*) adalah parameter untuk menentukan nilai *ramp-down time*. *Ramp-down time* adalah waktu yang dibutuhkan oleh motor untuk mengurangi kecepatan motor pada saat motor dalam keadaan frekuensi motor maksimum sampai berhenti. Waktu yang dibutuhkan untuk mencapai frekuensi motor dalam keadaan maksimum sampai berhenti adalah sebesar 4s.

15. P3900 (*End Quick Commissioning*)

P3900 (kode untuk menyelesaikan *setting*) adalah parameter untuk menentukan *End Quick Commissioning*. Setelah semua parameter telah diatur, maka yang perlu dilakukan adalah memilih angka “1”, yaitu *End Quick Commissioning* dengan mengatur ulang semua pengaturan pabrik.

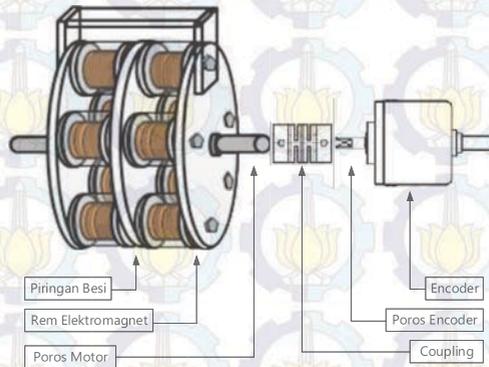
3.1.3 Perancangan *Rotary Encoder*

Encoder digunakan untuk mengukur kecepatan putaran (poros) motor dimana *encoder* akan mengubah data putaran/rotasi (RPM) menjadi data frekuensi (Hz). Secara tertulis kemampuan motor bisa bekerja hingga 1310 RPM, tetapi karena motor sudah berumur cukup lama, ketika motor diberi tegangan maksimal, motor hanya mampu berputar sebesar 1160 RPM.



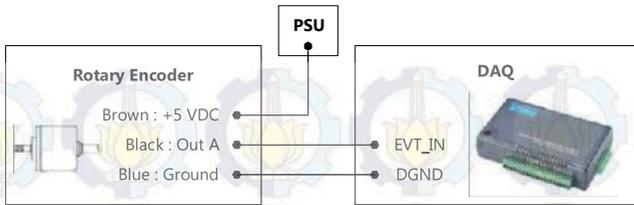
Gambar 3.3 Desain Rotary Encoder

Encoder tipe E30S4-100-3-N-5 mampu untuk menghitung putaran motor sebanyak 5000 RPM. Penelitian kali ini menggunakan *encoder* tipe *rotary encoder* produksi Autonic. *Encoder* yang membutuhkan *supply* tegangan 5 Volt ini memiliki diameter 30 mm dengan diameter *shaft* sebesar 4 mm yang menghasilkan 100 pulsa/revolusi dan memiliki 3 sinyal *out*.



Gambar 3.4 Konfigurasi Instalasi Encoder

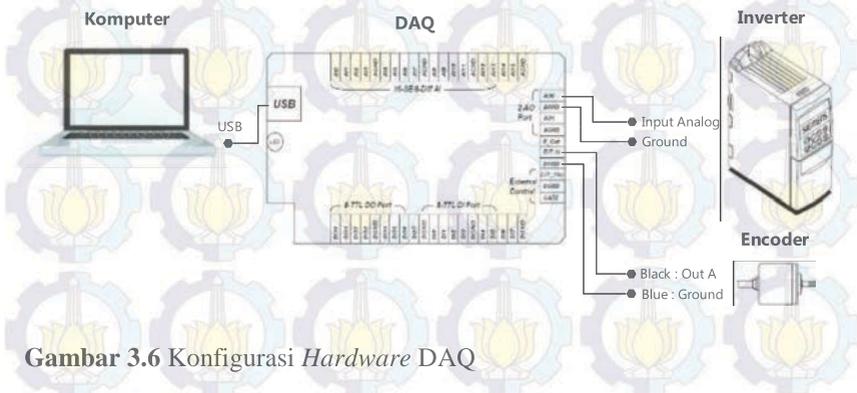
Untuk dapat menghitung putaran motor, hubungkan poros *encoder* dengan sebuah *coupling* yang sudah dihubungkan dengan poros motor seperti yang ditunjukkan pada Gambar 3.4. Diameter lubang *coupling* harus sama dengan diameter poros *encoder* dan poros motor, agar tidak terjadi *slip*. Sedangkan untuk dapat memberikan *output* data ke DAQ, hubungkan kabel hitam dan biru dari *encoder* ke *pin* EVT_IN dan *pin* DGND pada DAQ, lalu hubungkan kabel coklat ke *power supply* 5Volt. Untuk lebih jelasnya, proses *wiring* antara *encoder* dan DAQ ditunjukkan pada Gambar 3.5.



Gambar 3.5 Wiring Antara Encoder dan DAQ

3.1.4 Perancangan DAQ

Salah satu kelebihan karakter DAQ USB-4716 adalah *plug and play* sehingga komponen ini tidak memerlukan *jumper* atau *switch (setting)* pada modul. Tetapi *setting* dilakukan pada program bawaan DAQ tersebut.



Gambar 3.6 Konfigurasi Hardware DAQ

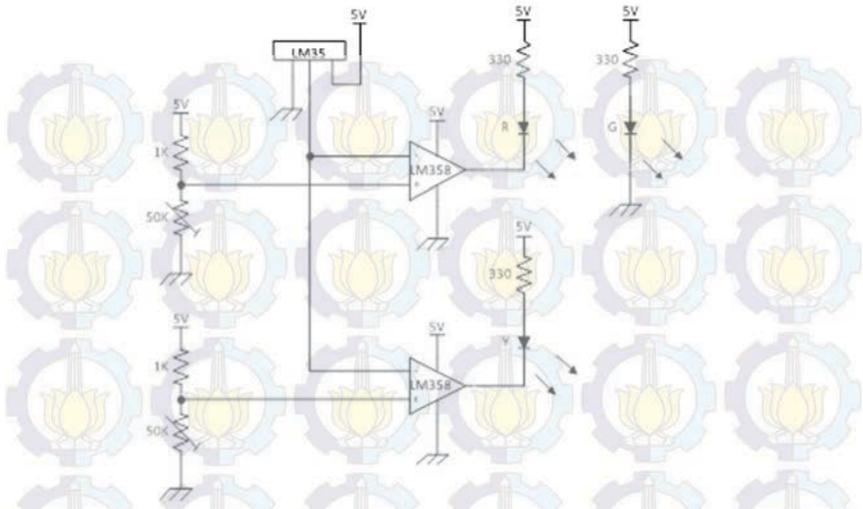
Untuk konfigurasi *hardware*, DAQ berhubungan dengan komputer, *Inverter*, dan *Encoder*. Agar komputer dapat berkomunikasi dengan DAQ cukup hubungkan *port* USB pada komputer dan *port* RS 232 pada DAQ dengan kabel RS 232 to USB. DAQ dapat menghitung *counter* yang masuk dari *Encoder* jika kabel hitam dan biru pada *Encoder* dihubungkan dengan *pin* EVT_IN dan DGND pada DAQ. *Pin* EVT_IN merupakan *pin* input untuk menghitung *counter*. Untuk menghubungkan *inverter* dengan DAQ, *pin* AO0 dan AGND pada DAQ perlu dihubungkan dengan *port* input analog dan *ground* pada *inverter*. Konfigurasi *hardware* DAQ telah ditunjukkan pada Gambar 3.6.

Tabel 3.1 Deskripsi Konektor Sinyal I/O

Nama Sinyal	Referensi	Arah	Deskripsi
AI<0...15>	AGND	Input	Analog Input Channels 0 - 15.
AIGND	-	-	Analog Input Ground.
AO0 dan AO1	AGND	Output	Analog Output Channels 0/1.
AOGND	-	-	Analog Output Ground. Pin ini adalah referensi dari tegangan analog output.
DI<0..7>	DGND	Input	Digital Input Channels.
DO<0..7>	DGND	Output	Digital Output Channels.
DGND	-	-	Digital Ground. Pin ini memberikan supply referensi untuk channel digital pada konektor I/O.
GATE	DGND	Input	A/D External Trigger Gate. Ketika pin ini dikoneksikan dengan +5 V, maka akan men-disable sinyal external trigger ke input.
EXT_TRG	DGND	Input	A/D External Trigger. Pin ini adalah sinyal input external trigger untuk konversi A/D.
EVT_IN	DGND	Input	External Events Input Channel.
P_OUT	DGND	Output	Pulse Output Channel.

3.1.5 Desain dan Perancangan Indikator Temperatur

Saat koil tembaga didalam magnet panas, magnet tidak bekerja, sehingga perlu diberikan indikator temperatur agar saat temperatur pada koil panas pengguna bisa langsung menyadari bahwa magnet tidak bekerja. Sensor temperatur LM 35 ditempelkan pada kedua sisi piringan magnet. Indikator temperatur ini memiliki tiga warna indikator, yaitu warna hijau, kuning, dan merah. Warna hijau menandakan panas antara range 0 C sampai 40 C. Warna kuning menandakan panas antara range 40 C sampai 60 C, sedangkan warna merah menandakan panas sudah melebihi 60 C dan juga menandakan untuk segera menghentikan pengereman karena pada kondisi ini rem magnet sudah tidak bekerja secara optimal. Gambar 3.7 merupakan gambar rangkaian indikator temperatur yang digunakan.



Gambar 3.7 Rangkaian Indikator Temperatur

3.1.6 Desain dan Perancangan Panel Kontrol



Gambar 3.8 Desain Panel Kontrol

Pada penelitian ini dibutuhkan ruang panel seluas 30 x 20 cm². Desain panel bagian depan bisa dilihat pada Gambar 3.8 dimana panel terbuat dari bahan besi.

Panel depan diberi komponen lampu indikator *On/Off*, lubang untuk melihat nilai frekuensi pada *inverter*, dan tombol *Emergency Stop*. Untuk bagian dalam panel, dibutuhkan ruang untuk memasukkan *inverter*, DAQ, trafo, terminal AC, dan MCB.

3.2 Perancangan Software

Perancangan *software* yang dilakukan yaitu dengan memberikan *setting* pada DAQ, memberikan masukan program kontroler PD *Fuzzy plus* kontroler *Integral* ke dalam *software* LabView. Selain itu juga diperlukan program untuk identifikasi sistem menggunakan program Matlab.

3.2.1 Perancangan Setting Software DAQ

Untuk memberikan *setting* pada DAQ, komputer sebagai kontrol utama perlu dihubungkan dengan DAQ. Setelah komputer dihubungkan dengan DAQ USB-4716, diperlukan instalasi beberapa *software* pendukung, seperti *software driver* DAQ dan *software driver* DAQ terhadap LabView.

Setelah *software* pendukung sudah ter-*instal*, maka bisa dilakukan pengecekan apakah DAQ sudah terhubung dengan komputer atau belum. Pengecekan ini bisa dilakukan dengan membuka *software* Advantech Navigator. Apabila pada sub *Installed Devices* sudah terdapat “USB-4716,BID#0” maka DAQ sudah terhubung dan *setting* DAQ sudah bisa dilakukan.

Untuk memasukkan komponen DAQ ke Labview, diharuskan untuk memilih komponen DAQNAvi Assistant agar kita dapat melakukan *setting* pada DAQ. Untuk melakukan *setting* DAQ sebagai *Analog Output*, pilih *Analog Output* pada jendela pertama, lalu pilih *next*, pilih *Static AO*, pilih *next*, pilih USB-4716,BID#0, pilih *next*, untuk *channel 0* dan *channel 1* pilih 0 ~ 5 V, pilih *next*, biarkan dalam kondisi *default*, pilih *next*, maka akan tampil *Overview* dari *setting* yang sudah dilakukan. Cek hasil *setting* kembali, apabila sudah selesai, pilih *finish*. Gambar langkah-langkah *setting Analog Output* bisa dilihat pada Gambar 3.9.

Untuk melakukan *setting* DAQ sebagai *Counter*, pilih *Counter* pada jendela pertama, pilih *Frequency Measurement*, pilih *next*, pilih USB-4716,BID#0, pilih *next*, pada *channel 0* pilih *CountingPulseByDevTime*,

pilih *next*, maka akan muncul *Overview* dari *setting* yang sudah dilakukan. Cek hasil *setting* kembali, apabila sudah selesai, pilih *finish*. Gambar langkah-langkah *setting Counter* bisa dilihat pada Gambar 3.10.



Gambar 3.9 Langkah membuat Analog Output DAQ pada Labview



(a)



(b)



(c)



(d)



(e)

Gambar 3.10 Langkah Membuat *Counter* DAQ pada Labview

3.2.2 Perancangan Identifikasi Sistem

Identifikasi sistem dilakukan menggunakan *System Identification Tool* pada program *Toolbox* Matlab dimana hasil identifikasi sistem digunakan untuk keperluan simulasi saja. Simulasi dilakukan agar didapat hasil respon sistem yang mendekati hasil implementasi sistem *real plant*.

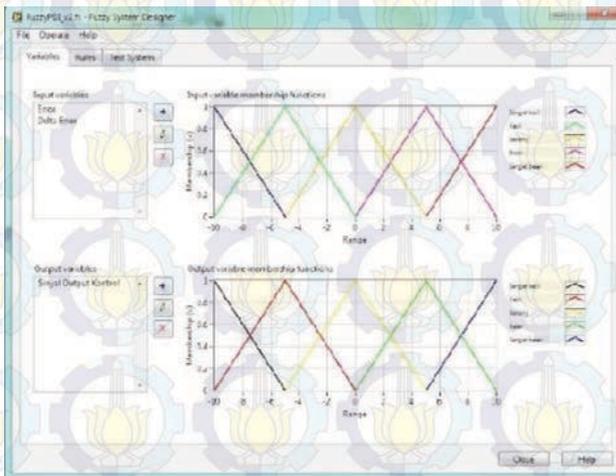
Identifikasi sistem yang dilakukan yaitu identifikasi dinamis. Sinyal PRBS dijadikan *input plant* dengan keadaan sistem *open loop*, lalu data hasil *output* dari *plant* disimpan untuk dianalisa.

3.3 Perancangan Kontroler pada Labview

Pada Tugas Akhir ini kontroler *Fuzzy PD* menggunakan *error* dan $\Delta error$ sebagai *input* dimana hasil sinyal kontrol dari *Fuzzy PD* akan ditambahkan sinyal *integral error* yang dihasilkan dari kontroler *Integral*. Selanjutnya akan dijelaskan mengenai perancangan *fuzzifikasi*, *rule base*, dan *defuzzifikasi* pada Labview.

3.3.1 Fuzzifikasi

Input dari *Fuzzifikasi* merupakan *error* dan $\Delta error$, dan masing-masing *input* memiliki lima *membership function* dimana masing-masing *input* memiliki nama *membership function* yaitu “Sangat Kecil”, “Kecil”, “Sedang”, “Besarnya”, dan “Sangat Besar”.



Gambar 3.11 Fuzzifikasi pada Labview

Begitu juga dengan sinyal *output*, sinyal *output* memiliki lima *membership function* yang memiliki nama yang sama dengan nama *membership function input*. Gambar 3.11 merupakan tampilan *fuzzifikasi* pada Labview yang digunakan.

3.3.2 Rule Base

Kontroler Fuzzy yang dirancang memiliki dua input, yaitu *error* dan $\Delta error$. Dimana masing-masing *input* memiliki lima *membership function*, maka akan dihasilkan *rule base* sebanyak dua puluh lima buah. Tampilan *rule base* dan *defuzzifikasi* ditunjukkan pada Gambar 3.12.



Gambar 3.12 Rule Base dan Defuzzifikasi pada Labview

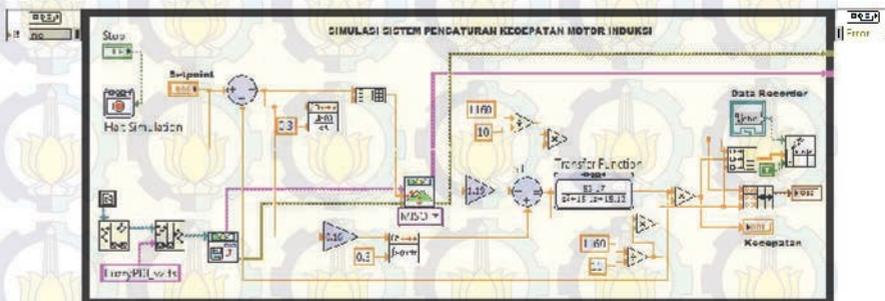
3.3.3 Defuzzifikasi

Metode yang digunakan untuk *defuzzifikasi* yaitu metode *Center of Area* karena metode ini yang paling umum digunakan untuk bentuk segitiga *membership function*.

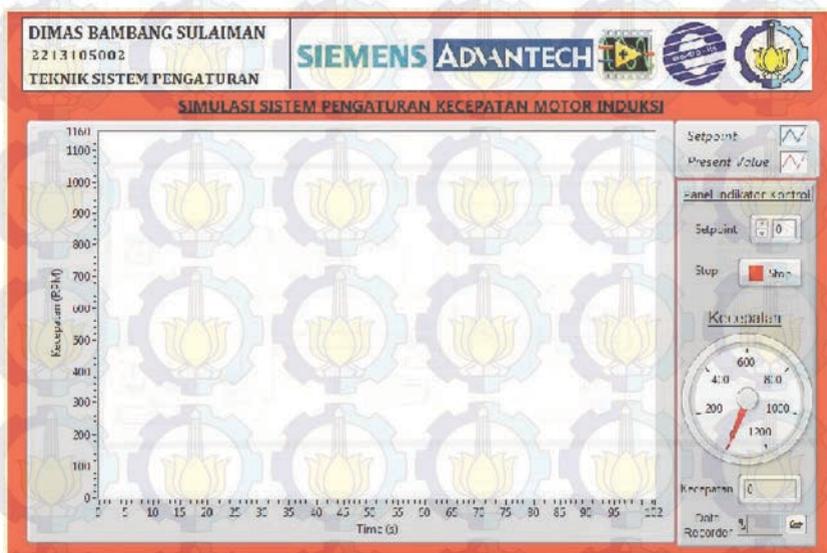
3.4 Perancangan Kontroler Fuzzy PD plus Kontroler Integral

Perbedaan program antara kontroler simulasi dan kontroler implementasi, terdapat pada *loop* yang digunakan. Pada kontroler simulasi menggunakan *loop* simulasi sedangkan pada kontroler implementasi menggunakan *loop* proses. Untuk pengaturan komponen

Fuzzy pada kontroler implementasi, disamakan dengan pengaturan komponen *Fuzzy* pada kontroler simulasi karena tidak terdapat perbedaan pada proses menggunakan komponen *Fuzzy* tersebut.

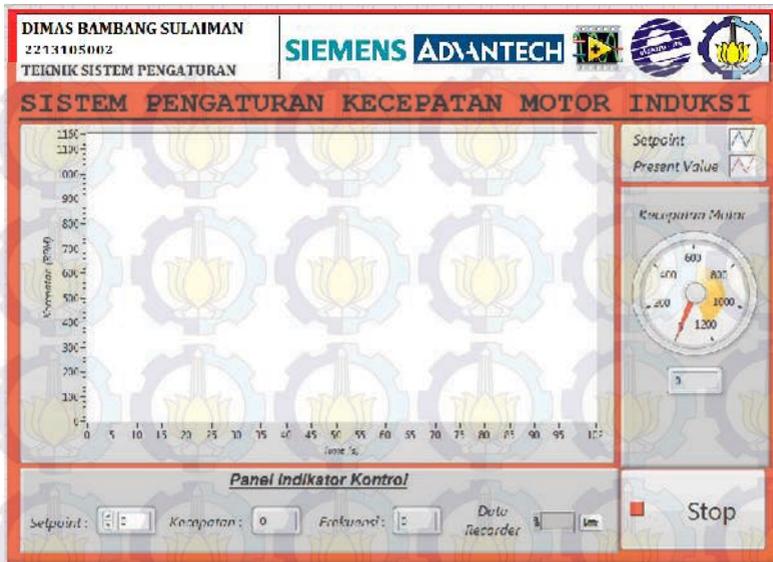


(a) Block Diagram pada Labview



(b) Front Panel pada Labview

Gambar 3.13 Kontroler Simulasi Menggunakan Labview



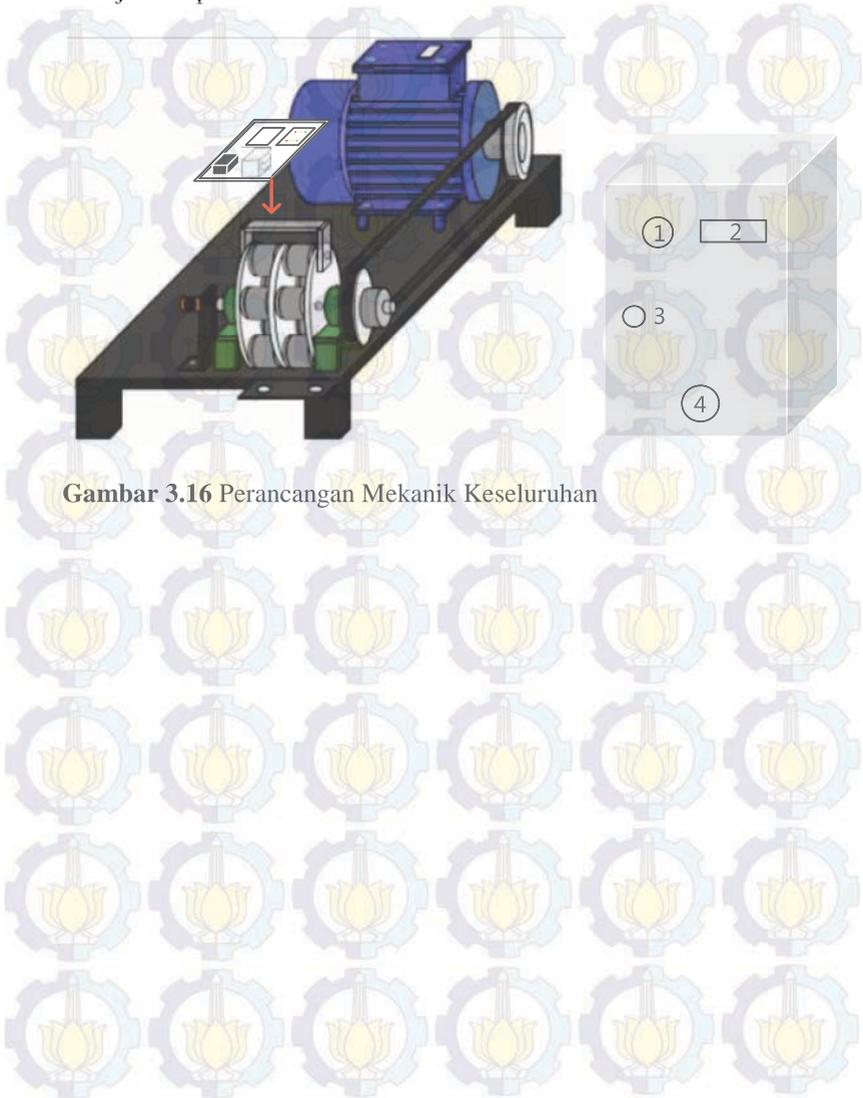
Gambar 3.15 Front Panel Kontroler Fuzzy PD plus kontroler Integral pada Labview

3.5 Perancangan Mekanik Keseluruhan

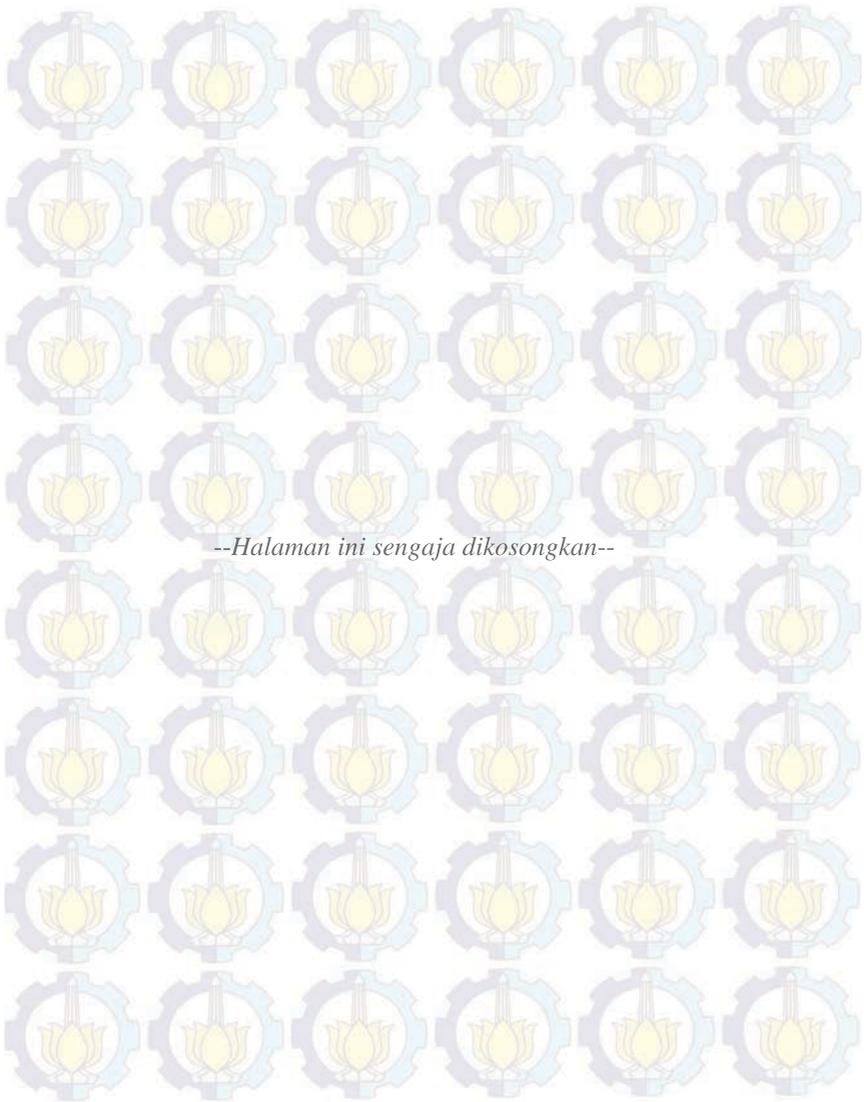
Plant yang digunakan merupakan hasil modifikasi dari Tugas Akhir dari kelompok rem magnetik sebelumnya. Tugas akhir yang digunakan saat ini dimodifikasi dengan melakukan penambahan sensor temperatur untuk mendeteksi panas pada piringan magnet, sehingga jika sensor temperatur sudah mulai menunjukkan indikasi panas, maka proses pengujian harus dihentikan karena saat piringan panas, daya magnet sudah tidak dapat bekerja optimal.

Seperti yang terlihat pada gambar, motor induksi akan di-*couple* menggunakan *belt* dengan piringan besi melalui poros rem magnetik. Sehingga saat sistem bekerja, piringan motor akan berputar sesuai dengan kecepatan putaran motor, lalu apabila diberikan beban rem magnetik, kecepatan putaran piringan besi akan berkurang. Kecepatan putaran piringan besi didapat dari *encoder* dengan cara poros *encoder* yang di-*couple* dengan poros rem magnetik. Sehingga setiap putaran poros rem magnetik akan menghasilkan frekuensi untuk *encoder* yang dapat

diproses oleh DAQ. Perancangan mekanik secara keseluruhan ditunjukkan pada Gambar 3.16.



Gambar 3.16 Perancangan Mekanik Keseluruhan



BAB IV PENGUJIAN DAN ANALISA

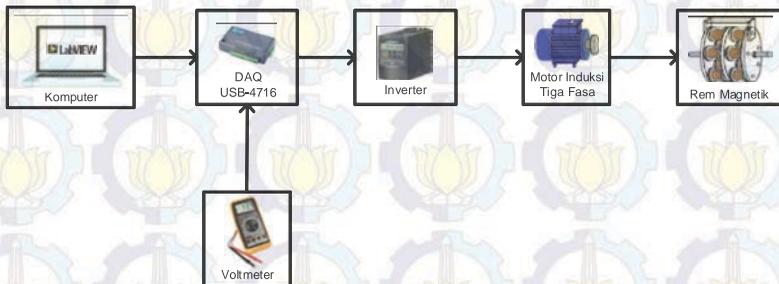
Pengujian dan analisa pada Tugas Akhir ini dibagi menjadi empat bagian, yaitu pengujian *hardware*, pengujian *software*, implementasi kontroler, dan analisa hasil simulasi dan hasil implementasi. Pada pengujian *hardware* akan ditunjukkan hasil pengujian dari kecepatan motor, DAQ, *encoder*, dan *indikator* temperatur. Pada pengujian *software* akan ditunjukkan hasil pengujian simulasi dan pengujian kontroler *Fuzzy PD plus* kontroler *Integral*. Selanjutnya akan ditunjukkan hasil implementasi kontroler dan analisa hasil simulasi dan hasil implementasi.

4.1 Pengujian *Hardware*

Pengujian *hardware* dilakukan dengan menggunakan beberapa alat ukur seperti digital multimeter, *function generator*, osiloskop, *tachometer* dan *thermometer*. Pengujian *hardware* dilakukan dengan tujuan untuk mengetahui apakah *hardware* berfungsi dengan semestinya.

4.1.1 Pengujian DAQ dan *Inverter*

Untuk melakukan pengujian DAQ dan *inverter* diperlukan alat ukur seperti *voltmeter*. Pengujian dilakukan dengan memberikan tegangan *output* analog pada DAQ, lalu mengukur frekuensi yang dihasilkan oleh *inverter*. Skema pengujian *inverter* ditunjukkan pada Gambar 4.1.



Gambar 4.1 Skema Pengujian *Inverter*

Pengujian DAQ dan *inverter* dilakukan untuk mengetahui apakah *output* dari DAQ sudah sesuai dengan *input* yang diberikan ke DAQ. Sedangkan pengujian untuk *inverter* dilakukan untuk mengetahui berapa besarnya frekuensi yang dihasilkan *inverter*. Data pengujian DAQ dan *inverter* ditunjukkan pada Tabel 4.1.

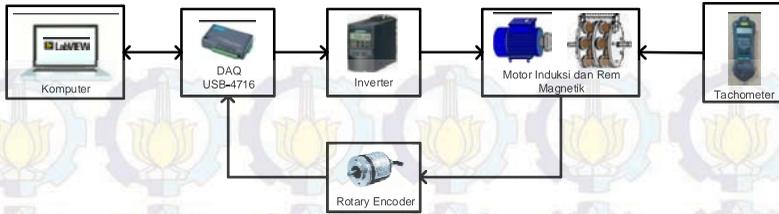
Tabel 4.1 Hasil Pengujian Antara DAQ, Kecepatan Motor, dan *Output Encoder*

No.	Input dari Labview (Volt)	Analog Output DAQ (Volt)	Output Inverter (Hz)
1	1	0,994	4,78
2	2	1,989	9,78
3	3	2,984	14,78
4	4	3,979	19,78
5	5	4,98	24,78
6	6	5,98	29,78
7	7	6,98	34,78
8	8	7,98	39,78
9	9	8,98	45,78
10	10	9,98	49,78

Berdasarkan data hasil pengujian pada Tabel 4.1 dapat dilihat bahwa *output* analog DAQ mengalami drop tegangan sebesar 0,02 Volt. Sedangkan besarnya frekuensi yang dihasilkan oleh inverter terjadi penurunan sebesar 0,22 Hz setiap titik ujinya, sehingga dapat diambil kesimpulan bahwa DAQ dan *inverter* dapat digunakan untuk mengoperasikan motor induksi tiga fasa.

4.1.2 Pengujian *Encoder*

Pada pengujian *encoder*, diperlukan sebuah *tachometer* sebagai sebuah acuan kecepatan motor. *Tachometer* Lutron DT-1236L digunakan untuk melakukan pengujian *encoder* kali ini.



Gambar 4.2 Skema Pengujian *Encoder*

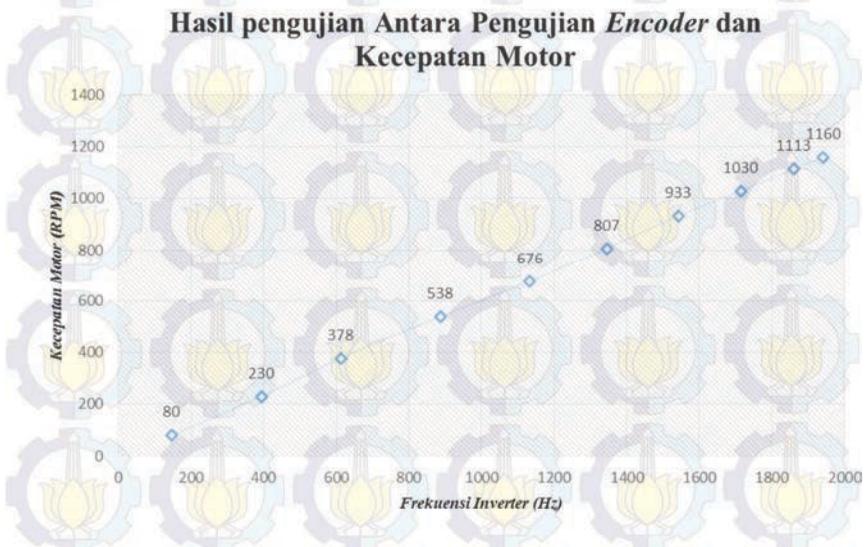
Digital tachometer adalah sensor kecepatan yang memanfaatkan sensor cahaya LDR (*Light Dependent Resistor*). Agar *tachometer* dapat menangkap kecepatan motor, poros motor harus diberi warna yang berbeda. Perbedaan warna tersebut membuat lampu LDR membaca nilai tegangan yang berbeda. Pada motor ini, warna yang digunakan di sekeliling poros motor adalah warna hitam, dan warna garis yang digunakan sebagai pembeda tegangan adalah warna putih. Perbedaan nilai tegangan pada perbedaan warna garis inilah yang ditangkap LDR sebagai satu putaran penuh. Hasil dari pengujian kecepatan motor menggunakan *tachometer* dan *encoder* ditunjukkan pada Tabel 4.2.

Tabel 4.2 Hasil Pengujian Antara Kecepatan Motor dan *Encoder*

No.	Input dari Labview (Volt)	Kecepatan Motor (RPM)	Output Encoder (Hz)
1	1	80	147
2	2	230	394
3	3	378	613
4	4	538	887
5	5	676	1134
6	6	807	1346
7	7	933	1542
8	8	1030	1715
9	9	1113	1860
10	10	1160	1942

Korelasi antara data hasil pengukuran kecepatan motor yang diukur menggunakan *tachometer* dengan hasil yang diukur *encoder* menghasilkan kurva lineritas yang ditunjukkan pada Gambar 4.3. Grafik ini menunjukkan bahwa motor induksi memiliki letak lineritas pada saat kecepatan 400 – 1000 rpm. Sehingga bisa diambil persamaan linier yang menunjukkan hubungan kecepatan motor induksi dan *output encoder*.

$$\text{Kecepatan Motor} = \frac{1160}{1942} \times \text{Output Encoder} \quad (4.1)$$



Gambar 4.3 Hasil Pengujian antara Pengujian Kecepatan Motor dan *Encoder*

4.1.3 Pengujian Sensor Temperatur

Sensor temperatur berfungsi sebagai indikator apabila temperatur pada rem magnetik sudah panas. Saat rem magnetik berada pada kondisi temperatur tinggi, kerja rem magnet tidak optimal dimana kondisi ini harus dihindari. Indikator temperatur ini memiliki tiga keadaan yang diwakili dengan tiga LED yaitu hijau, kuning, dan merah. Pada saat rem magnet sudah berada pada keadaan panas maksimal, LED merah akan aktif. Tabel 4.3 menunjukkan daerah kerja indikator temperatur.

Tabel 4.3 Daerah Kerja Indikator Temperatur

Temperatur	Hijau		Kuning		Merah	
	Kiri	Kanan	Kiri	Kanan	Kiri	Kanan
20	ON	ON	OFF	OFF	OFF	OFF
25	ON	ON	OFF	OFF	OFF	OFF
30	ON	ON	OFF	OFF	OFF	OFF
35	ON	ON	OFF	OFF	OFF	OFF
40	ON	ON	ON	ON	OFF	OFF
45	ON	ON	ON	ON	OFF	OFF
50	ON	ON	ON	ON	OFF	OFF
55	ON	ON	ON	ON	OFF	OFF
60	ON	ON	ON	ON	ON	ON

Dari Tabel 4.3 menunjukkan bahwa indikator temperatur akan menunjukkan warna hijau saat indikator menyala, Warna kuning akan aktif saat sensor temperatur mencapai temperatur 40 °Celcius dan warna merah akan aktif saat temperatur mencapai 60 °Celcius.

4.2 Pengujian Simulasi

Sebelum melakukan implementasi, terlebih dulu dilakukan simulasi. Simulasi dilakukan menggunakan program Labview 2013. Pengujian simulasi dilakukan tanpa kontroler dan dengan kontroler. Tujuan dari pengujian ini yaitu untuk mengetahui perbedaan respon sistem dengan kontroler dan tanpa kontroler.

4.2.1 Pengujian Simulasi Tanpa Kontroler

Simulasi dengan menggunakan rancangan yang sudah dibahas pada Bab 3.4 dijelaskan pada bagian ini. Pada saat diberi masukan sebesar 8 Volt atau setara dengan 929 rpm, *output* respon menghasilkan 1000 rpm. Sehingga dapat disimpulkan bahwa sistem memiliki *error* sebanyak 72 rpm. Sedangkan pada saat diberi masukan 9 Volt atau setara dengan 1044 rpm, *output* respon menghasilkan 1126 rpm. Sehingga sistem memiliki *error* sebanyak 82 rpm. Hasil pengujian simulasi tanpa kontroler ditunjukkan pada Gambar 4.4. Dimana garis warna biru

merepresentasikan sinyal referensi dan garis warna hijau merepresentasikan sinyal *output* sistem.

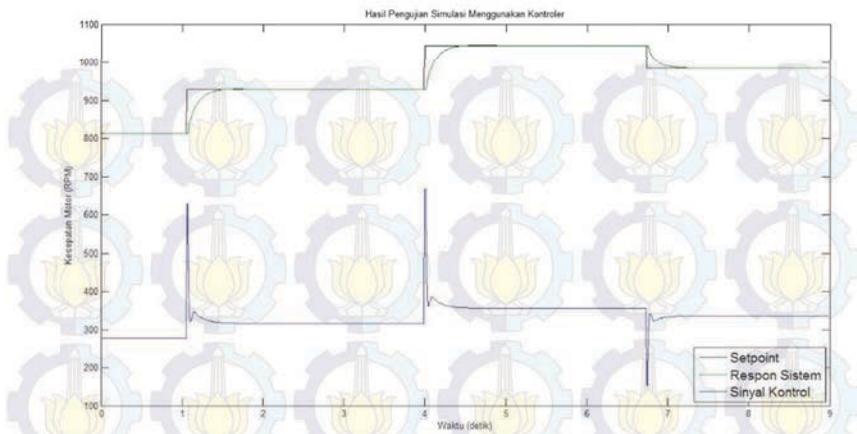


Gambar 4.4 Hasil Pengujian Simulasi Tanpa Kontroler

4.2.2 Pengujian Simulasi dengan Kontroler

Dengan melakukan simulasi dengan kontroler, diharapkan *error* yang terdapat pada sistem sebelum diberi kontroler dapat berkurang. Ketika diberi *input* 8 Volt atau setara dengan 929 rpm, *output* respon menghasilkan nilai sebesar 929 rpm. Sedangkan ketika diberi *input* 9 volt yang setara dengan 1044 rpm, *output* respon pun menghasilkan putaran motor sebesar 1044 rpm.

Data hasil pengujian simulasi menggunakan kontroler ditunjukkan pada Gambar 4.5 dimana garis warna hitam merepresentasikan sinyal referensi, garis warna hijau merepresentasikan sinyal respon sistem, dan garis warna biru merepresentasikan sinyal kontrol. Setelah data pengujian didapat, dilakukan perhitungan spesifikasi respon sistem. Waktu yang dibutuhkan sistem untuk *steady state* (t_s ($\pm 5\%$)) yaitu sebesar 0,3 detik. Waktu yang dibutuhkan sistem untuk mencapai titik *steady state* pertama kali (t_r) yaitu sebesar 0,229 detik. Sedangkan waktu yang dibutuhkan untuk mencapai setengah dari nilai *steady state* (t_d) adalah 0,07 detik dan memiliki *%error* sebesar 0,044 %. Hasil spesifikasi respon simulasi bisa dilihat pada Tabel 4.4.



Gambar 4.5 Hasil Pengujian Simulasi Menggunakan Kontroler

Tabel 4.4 Hasil Spesifikasi Respon Simulasi Menggunakan Kontroler

Spesifikasi	Hasil
<i>Settling Time</i> ($\pm 5\%$)	0,3 detik
<i>Rise Time</i>	0,229 detik
<i>Delay Time</i>	0,07 detik
<i>% error</i>	0,044%

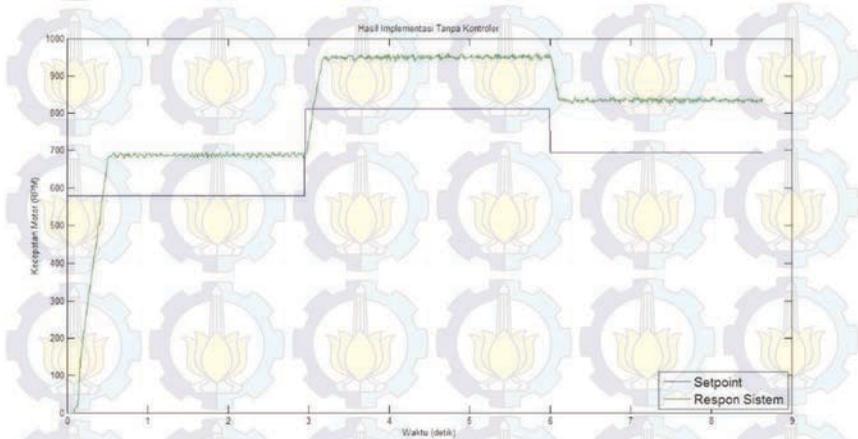
4.3 Implementasi Pengujian Kontroler

Implementasi dilakukan menggunakan *software* Labview 2013, dan menggunakan *interface* yaitu DAQ Advantech USB-4716. Implementasi dilakukan dengan tujuan untuk mengetahui performansi *plant*, dan menguji kemampuan kontroler yang sudah dirancang.

4.3.1 Pengujian Implementasi Tanpa Kontroler

Implementasi dengan menggunakan rancangan yang sudah dibahas pada Bab 3.4 dijelaskan pada bagian ini. Pada saat diberi masukan sebesar 5 Volt atau setara dengan 580 rpm, *output* respon menghasilkan 688,27 rpm. Sehingga dapat disimpulkan bahwa sistem memiliki *error* sebanyak 108,27 rpm. Sedangkan pada saat diberi masukan 7 Volt atau setara dengan 812 rpm, *output* respon menghasilkan 949,94 rpm.

Sehingga sistem memiliki *error* sebanyak 137,94 rpm. Dapat disimpulkan bahwa sistem memiliki karakter yang tidak linear karena selisih yang dihasilkan saat diberi *setpoint* berbeda-beda. Hasil pengujian simulasi tanpa kontroler ditunjukkan pada Gambar 4.6, dimana garis warna biru merepresentasikan sinyal referensi dan garis warna hijau merepresentasikan sinyal respon sistem.

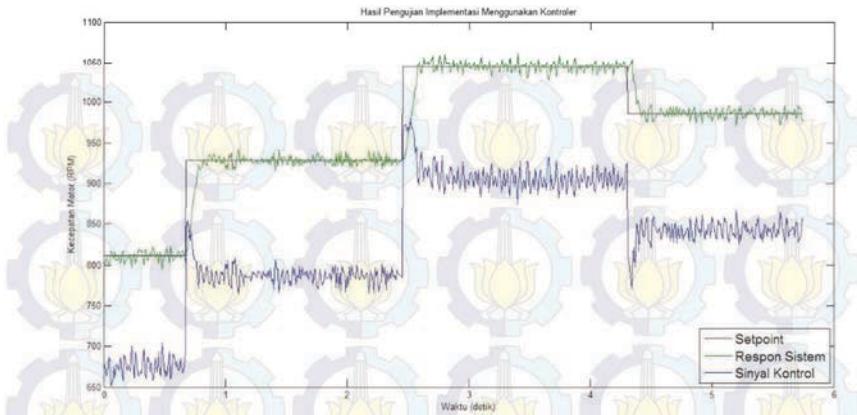


Gambar 4.6 Hasil Pengujian Implementasi Tanpa Kontroler

4.3.2 Pengujian Implementasi Dengan Kontroler

Pengujian implementasi dengan kontroler diharapkan dapat memperkecil *error* yang dimiliki sistem. Pada saat sistem diberi *input* sebesar 8 Volt atau setara dengan 928 rpm, respon sistem menghasilkan nilai sebesar 928,626 rpm. Maka selisih antara *setpoint* dan kecepatan saat *steady state* hanya terpaut 0,626 rpm. Hasil pengujian implementasi menggunakan kontroler ditunjukkan pada Gambar 4.7, dimana garis warna hitam merepresentasikan sinyal referensi, garis warna hijau merepresentasikan sinyal respon sistem, dan garis warna biru merepresentasikan sinyal kontrol.

Untuk menguji implementasi menggunakan kontroler, diberikan *input* sebesar 8 Volt atau setara dengan 928 rpm. Saat diberikan *input*, sistem menghasilkan kecepatan sebesar 928,1389 rpm.



Gambar 4.7 Hasil Pengujian Implementasi Menggunakan Kontroler

Spesifikasi respon sistem menunjukkan bahwa sistem memiliki nilai t_s ($\pm 5\%$) yaitu sebesar 0,11 detik. Waktu yang dibutuhkan sistem untuk mencapai titik *steady state* pertama kali (t_r) yaitu sebesar 0,076 detik. Sedangkan waktu yang dibutuhkan untuk mencapai setengah dari nilai *steady state* (t_d) adalah 0,06 detik dan memiliki % *error* sebesar 0,015 %.

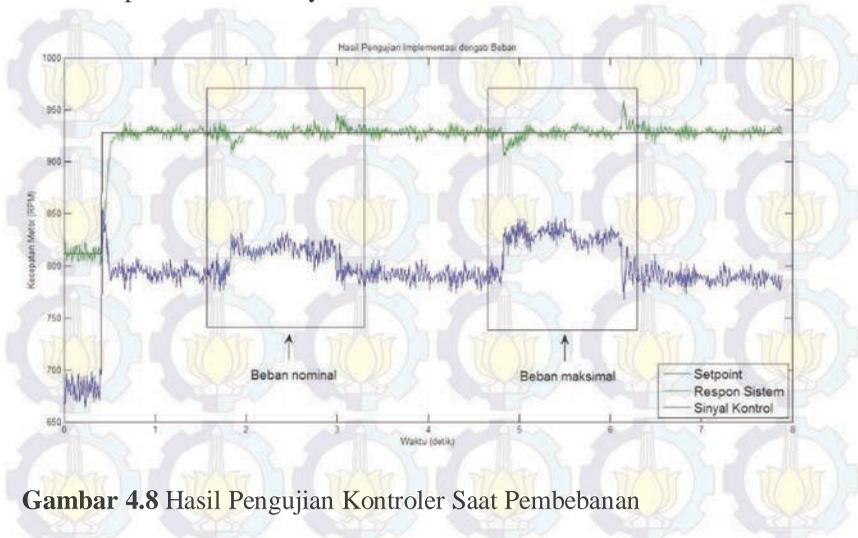
Tabel 4.5 Hasil Spesifikasi Respon Sistem Menggunakan Kontroler

Spesifikasi	Hasil
<i>Settling Time</i> ($\pm 5\%$)	0,11 detik
<i>Rise Time</i>	0,076 detik
<i>Delay Time</i>	0,06 detik
% <i>error</i>	0,015%

4.3.3 Pengujian Implementasi Dengan Beban

Pengujian ini dilakukan untuk mengetahui apakah sistem dapat bekerja saat diberi beban dan juga untuk mengetahui berapakah besar daya pengereman rem magnetik yang diberikan untuk kondisi beban nominal dan beban maksimal. Hasil pengujian ditunjukkan pada Gambar 4.8. dimana garis warna hitam merepresentasikan sinyal referensi, garis

warna hijau merepresentasikan sinyal respon sistem, dan garis warna biru merepresentasikan sinyal kontrol.



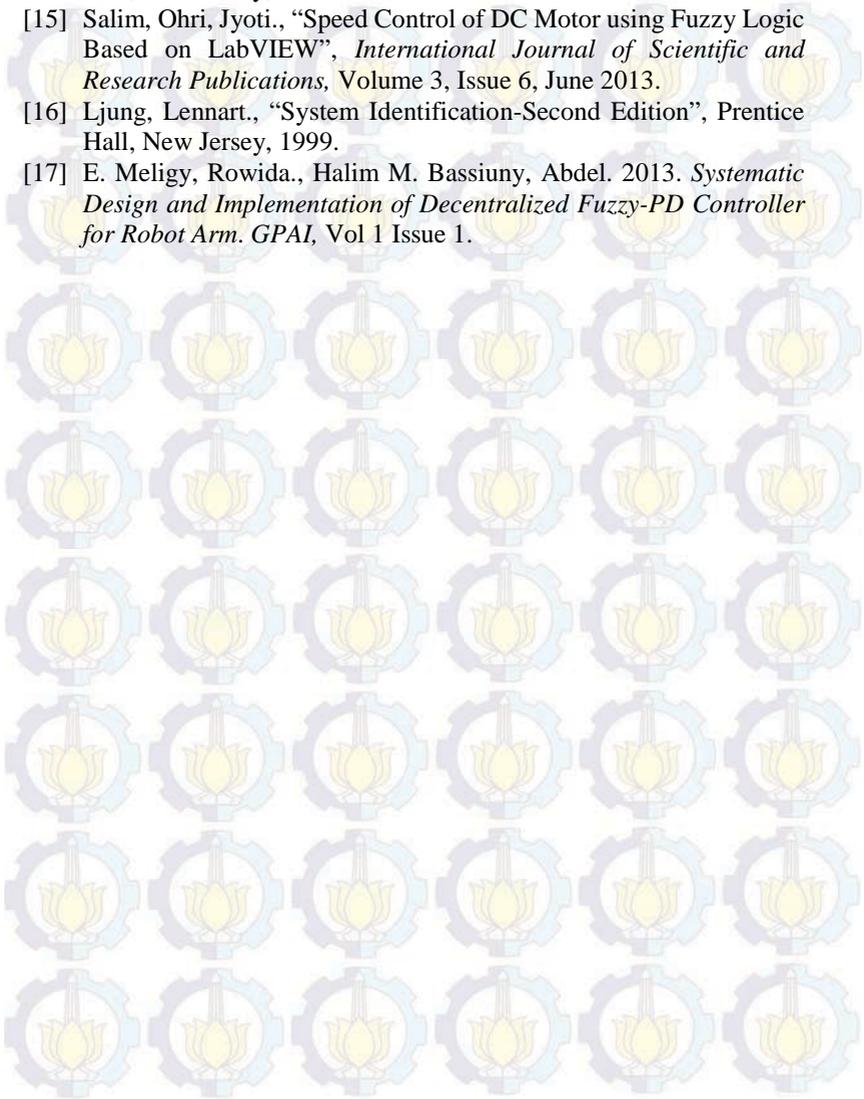
Gambar 4.8 Hasil Pengujian Kontroler Saat Pembebanan

Pada Gambar 4.8 terlihat bahwa saat beban diberikan, kecepatan motor berkurang dan tak lama kemudian, motor bekerja ke kecepatan mendekati referensi. Tetapi bisa dilihat saat beban diberikan, sinyal kontroler bertambah besar. Hal ini menandakan kontroler bekerja lebih besar saat diberi beban. Dapat dilihat juga perbedaan nilai sinyal kontrol yang diberikan saat beban nominal dan saat beban maksimal. Sinyal kontrol yang dihasilkan saat diberikan beban maksimal lebih besar daripada saat diberi beban nominal. Hal ini menandakan kontroler bersifat *regulator* yaitu kontroler dapat mengikuti referensi walaupun diberi beban yang berubah-ubah. Setelah dilakukan perhitungan, saat diberi beban nominal, sinyal kontroler memberikan output sebesar 0,22 Volt dan saat diberi beban maksimal, sinyal kontroler memberikan output sebesar 0,324 Volt.

DAFTAR PUSTAKA

- [1] Rashid, Muhammad H., *“Power Electronics Handbook : Devices, Circuits, and Applications Handbook - Third Edition”*, Elsevier, USA, 2011.
- [2] _____, *“Micromaster 420 Operating Instructions”*, Siemens User Documentation.
- [3] _____, *“USB-4716 User Manual – First Edition”*, Advantech Co., 2006.
- [4] _____, *“Rotary Encoder E30S4 Series Manual”*, Autonics Co.,
- [5] Semit, Arwanjer. *“Perancangan dan Implementasi Kontroler Linear Quadratic Regulator (LQR) pada Pengaturan Kecepatan Motor Induksi 3 Phasa”*, *Tugas Akhir Teknik Elektro*, Institut Teknologi Sepuluh Nopember, Juni, 2014.
- [6] _____, *“Performance Measures - Integral Square Error (ISE)”*.<http://www.facstaff.bucknell.edu/mastascu/econtrolhtml/Design/Perf5IntSqError.htm>, Mei, 2015.
- [7] _____, *“Lab VIEW Basics I Course Manual”*, National Instruments, 1999.
- [8] ÇELİK, Emre., DALCALI, Adem ., *“An Adaptive PI Controller Schema Based on Fuzzy Logic Controller for Speed Control of Permanent Magnet Synchronous Motors”*, *4th International Conference on Power Engineering, Energy and Electrical Drives*, May, 2013.
- [9] Patra, M.J., Mlynek.D.M., *“Fuzzy Logic : Implementation and Applications”*, Wiley Teubner, England, 1996.
- [10] Akbiyik, Birkan., Eksin, İbrahim., *“Evaluation of The Performance Of Various Fuzzy PID Controller Structures On Benchmark Systems”*, *Istanbul Technical University*,
- [11] Ana Ratna, Dwi., *“Sistem Kendali Cerdas”*, Graha Ilmu, Yogyakarta, 2011.
- [12] Li, Wei., *“Design of a Hybrid Fuzzy Logic Proportional Plus Conventional Integral-Derivative Controller”*, *IEEE Transactions On Fuzzy Systems*, Vol. 6, No. 4, November, 1998.
- [13] Er, Meng Joo., *“Hybrid Fuzzy Proportional–Integral Plus Conventional Derivative Control of Linear and Nonlinear Systems”*, *IEEE Transactions On Industrial Electronics*, Vol. 48, No. 6, December, 2001.

- [14] Ogata, K., “*Modern Control Engineering- Third Edition*”, Prentice Hall, New Jersey, 1997.
- [15] Salim, Ohri, Jyoti., “Speed Control of DC Motor using Fuzzy Logic Based on LabVIEW”, *International Journal of Scientific and Research Publications*, Volume 3, Issue 6, June 2013.
- [16] Ljung, Lennart., “*System Identification-Second Edition*”, Prentice Hall, New Jersey, 1999.
- [17] E. Meligy, Rowida., Halim M. Bassiuny, Abdel. 2013. *Systematic Design and Implementation of Decentralized Fuzzy-PD Controller for Robot Arm. GPAI*, Vol 1 Issue 1.



BAB V

PENUTUP

5.1 Kesimpulan

Dari hasil pengujian simulasi dan implementasi pada pengaturan kecepatan motor induksi tiga fasa menggunakan *Fuzzy PD plus* kontroler *Integral* didapatkan beberapa kesimpulan, yaitu:

- a. Kontroler *Fuzzy PD plus* kontroler *Integral* pada simulasi Labview mampu mengikuti sinyal *setpoint* yang diberikan. Spesifikasi respon sistem dengan *settling time* 0,3 detik, *rise time* 0,229 detik, *delay time* 0,07 detik dan *error steady state* sebesar 0,044 %.
- b. Kontroler *Fuzzy PD plus* kontroler *Integral* pada implementasi Labview mampu mengikuti sinyal *setpoint* yang diberikan. Spesifikasi respon transien dengan *settling time* 0,11 detik, *rise time* 0,076 detik, *delay time* 0,06 detik dan *error steady state* sebesar 0,015 %.
- c. Sinyal kontrol bertambah sebesar 0,22 V saat diberikan beban nominal, sedangkan saat diberikan beban maksimal sinyal kontrol bertambah sebesar 0,324 V.

5.2 Saran

Beberapa saran yang bisa diberikan untuk untuk pengembangan penelitian lebih lanjut, yaitu :

- a. Melakukan perbaikan rem magnetik agar daya cengkram motor lebih besar.
- b. Melakukan *setting* ulang poros piringan motor agar guncangan bisa dikurangi.



b. Datasheet DAQ

USB-4716

200 kS/s, 16-bit, 16-ch Multifunction USB Module



CE FCC RoHS

Features

- Supports USB 2.0
- Portable
- Bus-powered
- 16 analog input channels
- 16-bit resolution AI
- Sampling rate up to 200 kS/s
- 8-ch DVS-ch DO, 2-ch AO and one 32-bit counter
- Detachable screw terminal on modules
- Suitable for DIN-rail mounting
- One lockable USB cable for secure connection included

Introduction

The USB-4700 series consists of true plug & play data acquisition devices. No more opening up your computer chassis to install boards—just plug in the module, then get the data. It's easy and efficient. USB-4716 offers 16 single-ended/ 8 differential inputs with 16-bit resolution, up to 200 kS/s throughput, 16 digital I/O lines and 1 user counter, add two 16-bit analog outputs. The high performance makes USB-4716 your best choice for test & measurement applications in the production line or in the lab.

Reliable and rugged enough for industrial applications, yet inexpensive enough for home projects, the USB-4716 is the perfect way to add measurement and control capability to any USB capable computer. The USB-4700 series is fully plug & play and easy to use. It obtains all required power from the USB port, so no external power connection is ever required.

Specifications

Analog Input

- **Channels** 16 single-ended/ 8 differential (software programmable)
 - **Resolution** 16 bits
 - **Max. Sampling Rate** 200 kS/s (for USB 2.0)
- Note: The sampling rate for each channel will be affected by used channel number. For example, if 4 channels are used, the sampling rate is 200k/4 = 50 kS/s per channel.
- **FIFO Size** 1,024 samples
 - **Overvoltage Protection** 30 Vp-p
 - **Input Impedance** 1 GΩ
 - **Sampling Modes** Software, onboard programmable preset, or external
 - **Input Range (V, software programmable) & Absolute Accuracy**

	N/A	0 - 10	0 - 5	0 - 2.5	0 - 1.25
Single Ended	N/A	0 - 10	0 - 5	0 - 2.5	0 - 1.25
Differential	±10	±5	±2.5	±1.25	±0.625
Absolute Accuracy (% of FSR)*	0.05	0.03	0.02	0.02	0.1

*: ±1 LSB is added as the derivative for absolute accuracy

Analog Output

- **Channels** 2
- **Resolution** 16 bits
- **Output Rate** Static update
- **Output Range** (V, software programmable)

Internal Reference	Unipolar	Bipolar
	0 - 5, 0 - 10	±5, ±10

- **Slew Rate** 0.7 V/μs
- **Driving Capability** 5 mA
- **Output Impedance** 0.1 Ω max.
- **Operation Mode** Single output
- **Accuracy** Relative: ±1 LSB

Digital Input

- **Channels** 8
- **Compatibility** 3.3 V/5 V TTL
- **Input Voltage** Logic 0: 1.0 V max.
Logic 1: 2.0 V min.

Digital Output

- **Channels** 8
- **Compatibility** 3.3 V TTL
- **Output Voltage** Logic 0: 0.4 V max.
Logic 1: 2.4 V min.
- **Output Capability** Sink: 6 mA (sink)
Source: 6 mA (source)

Event Counter

- **Channels** 1
- **Compatibility** 3.2V/TTL
- **Max. Input Frequency** 1 kHz

General

- **Bus Type** USB 2.0
- **I/O Connector** Onboard screw terminal
- **Dimensions (L x W x H)** 132 x 80 x 32 mm (5.2" x 3.15" x 1.26")
- **Power Consumption** Typical: 5 V @ 360 mA
Max.: 5 V @ 450 mA
- **Operating Temperature** 0 - 60°C (32 - 158°F) (refer to IEC 68-2-1, 2)
- **Storage Temperature** -20 - 70°C (-4 - 158°F)
- **Operating Humidity** 5 - 85% RH non-condensing (refer to IEC 68-1, -2, -3)
- **Storage Humidity** 5 - 95% RH non-condensing (refer to IEC 68-1, -2, -3)

Ordering Information

- **USB-4716-AE** 200 kS/s, 16-bit, 16-ch Multi. USB Module

Accessories

- **1960004544** Wallmount Bracket
- **1960005788** VESA Mount Bracket