



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**Badak LNG**  
Center of Excellence

PNJ – BADAK LNG

**RANCANG BANGUN SISTEM KONTROL BERBASIS PENGONTROL PID  
PADA MINIPLANT PEMBANGKIT LISTRIK BERTENAGA UAP DARI  
PANAS BUANG INCENERATOR SAMPAH**

Tugas Akhir

**POLITEKNIK  
NEGERI  
JAKARTA**  
oleh:  
**Jerry Farhan**  
NIM. 1902322013

**PROGRAM STUDI D3 TEKNIK KONVERSI ENERGI LNG**

**ACADEMY,**

**KERJASAMA PNJ – BADAK LNG**

**JURUSAN TEKNIK MESIN, PROGRAM STUDI D3 TEKNIK KONVERSI ENERGI**

**KONSENTRASI ELECTRICAL & INSTRUMENTATION**

**BONTANG, 2022**



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



**Badak LNG**  
Center of Excellence

**PNJ – BADAK LNG**

# **RANCANG BANGUN SISTEM KONTROL BERBASIS PENGONTROL PID PADA MINIPLANT PEMBANGKIT LISTRIK BERTENAGA UAP DARI PANAS BUANG INCENERATOR SAMPAH**

**LAPORAN TUGAS AKHIR**

Laporan ini disusun sebagai salah satu syarat untuk menyelesaikan pendidikan Diploma III Program Studi Teknik Teknik Konversi Energi, Jurusan Teknik Mesin

oleh:

**Jerry Farhan**

**NIM. 1902322013**

**POLITEKNIK  
NEGERI  
JAKARTA**

**PROGRAM STUDI D3 TEKNIK KONVERSI ENERGI LNG**

**ACADEMY,**

**KERJASAMA PNJ – BADAK LNG**

**JURUSAN TEKNIK MESIN, PROGRAM STUDI D3 TEKNIK KONVERSI ENERGI**

**KONSENTRASI ELECTRICAL & INSTRUMENTATION**

**BONTANG, 2022**

**HALAMAN PENGESAHAN  
LAPORAN TUGAS AKHIR**

**RANCANG BANGUN SISTEM KONTROL BERBASIS PENGONTROL PID PADA  
MINIPLANT PEMBANGKIT LISTRIK BERTENAGA UAP DARI PANAS BUANG  
INCENERATOR SAMPAH**

Oleh:

Jerry Farhan

NIM. 1902322013

Program Studi Teknik Konversi Energi

Telah berhasil dipertahankan dalam sidang tugas akhir di hadapan Dewan Penguji pada tanggal 28 Agustus 2022 dan diterima sebagai persyaratan untuk memperoleh gelar Diploma

III pada Program Studi Teknik Konversi Energi Jurusan Teknik Mesin

**DEWAN PENGUJI**

No.	Nama	Posisi Penguji	Tanda Tangan	Tanggal
1.	Hasvienda S.T.MT	Penguji 1		
2.	Dr. Haolia Rahman S.T.MT	Penguji 2		06/09/2022
3.	Ir. Ahmad Fadil Reviansyah, S.T, IPM	Penguji 3		31/08/2022
4.	Ir. Eko Wahyu Susilo, S.T, IPM	Penguji 4		31/08/2022
5.	Ir. Prima Patriana, S.T, IPM	Penguji 5		31/08/2022

Bontang, 28 Agustus 2022

Disahkan Oleh:

Ketua Program Studi



Dr. Yuli Mafendro Dedet Eka Saputra, S. Pd., M.T.

NIP. 199403092019031013

**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



## HALAMAN PENGESAHAN

### RANCANG BANGUN SISTEM KONTROL BERBASIS PENGONTROL PID PADA MINIPLANT PEMBANGKIT LISTRIK BERTENAGA UAP DARI PANAS BUANG INCENERATOR SAMPAH

Diajukan oleh:  
Jerry Farhan  
NIM. 1802322013  
Program Studi Teknik Konversi Energi

Disetujui oleh:

Pembimbing Tugas Akhir  
Politeknik Negeri Jakarta



Drs. Azwardi, ST., M.Kom.  
NIP. 195804061986031001

Pembimbing Tugas Akhir  
Badak LNG



Ir. Rivon Tridesman, S.T. LP.M  
NIP. 134464

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



## LEMBAR PERNYATAAN ORISINALITAS

Saya yang bertanda tangan di bawah ini:

Nama : Jerry Farhan

NIM : 1902322013

Program Studi : Teknik Konversi Energi

Menyatakan bahwa yang dituliskan di dalam Laporan Tugas Akhir ini adalah hasil karya saya sendiri bukan jiplakan (plagiasi) karya orang lain baik sebagian atau seluruhnya. Pendapat, gagasan, atau temuan orang lain yang terdapat di dalam Laporan Tugas Akhir telah saya kutip dan saya rujuk sesuai dengan etika ilmiah.

Demikian pernyataan ini saya buat dengan sebenar-benarnya.

Bontang, 28 Agustus 2022



Jerry Farhan

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



## RANCANG BANGUN SISTEM KONTROL BERBASIS PENGONTROL PID PADA MINIPLANT PEMBANGKIT LISTRIK BERTENAGA UAP DARI PANAS BUANG INCENERATOR SAMPAH

Jerry Farhan, Azwardi, Rivon Tridesan

Program Studi Diploma III Teknik Konversi Energi, Jurusan Teknik Mesin, Politeknik

Negeri Jakarta, Kampus UI Depok, 16424

PT. Badak NGL, Bontang – Kalimantan Timur, 75324

Email: [jerryfarhan15.jf@gmail.com](mailto:jerryfarhan15.jf@gmail.com)

### ABSTRAK

*Incinerator non-Industrial* yang akan dibangun merupakan *Waste Burner Hybrid Steam Generator* berbahan bakar biodiesel hasil pengolahan limbah minyak goreng di *Nursery Badak LNG*. Api itulah yang kemudian akan membakar sampah pada *chamber* bakar di bagian atas *waste burner*. Kalor hasil pembakaran tersebut kemudian diusulkan untuk di-*recovery* sebagai pemanas lanjutan dari *water reservoir* kedua yang akan menghasilkan uap air bertekanan, yang nantinya akan digunakan sebagai penggerak pembangkit listrik, suhu dan tekanan api keluaran dari incinerator akan dikontrol dengan mengatur jumlah tekanan uap menggunakan *control valve*, yang bertujuan untuk mengurangi nilai rasio udara dan bahan bakar yang terdapat pada incinerator tersebut. Uap pada tanki air juga akan dikontrol waktu pelepasannya, agar tekanan yang ditahan dapat memutar turbin dan menghasilkan energi listrik. *Valve* yang digunakan adalah jenis *gate valve* yang akan digerakkan oleh motor stepper, bukaan valve nya bergantung dari jumlah putaran yang akan diberikan oleh motor stepper. Jenis kontrol yang akan diterapkan pada system ini adalah system kontrol PID yang mana dapat melakukan adaptasi terhadap jumlah *error* pada pembacaan sensor terhadap proses. Dengan demikian, operator tidak perlu memikirkan gangguan yang dapat mempengaruhi proses kontrol tersebut.

**Kata Kunci:** *Generator, motor stepper, PID, control valve, tekanan uap air.*

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumunkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





## KATA PENGANTAR

Puji serta syukur penulis panjatkan kehadiran Tuhan YME, yang telah melimpahkan rahmat dan karunianya-nya, sehingga penulis dapat menyelesaikan tugas akhir yang berjudul “**Rancang Bangun Sistem Kontrol Berbasis Pengontrol PID pada Miniplant Pembangkit Listrik Bertenaga Uap dari Panas Buang Incinerator**”. Tugas akhir ini disusun untuk memenuhi salah satu syarat dalam menyelesaikan studi Diploma-III program studi teknik konversi energi, jurusan teknik mesin, Politeknik Negeri Jakarta.

Penulisan Tugas Akhir ini tidak lepas dari bantuan dari berbagai pihak, oleh karena itu penulis ingin menyampaikan ucapan terimakasih kepada:

1. Bapak Dr. Eng. Muslimin, S.T., M.T. selaku Ketua Jurusan Teknik Mesin Politeknik Negeri Jakarta.
2. Bapak Johan Anindhito Indriawan selaku Direktur LNG Academy PT Badak NGL.
3. Bapak Drs. Azwardi, ST., M.Kom. selaku Dosen Pembimbing dari Politeknik Negeri Jakarta yang telah memberikan bimbingan dalam penyelesaian Tugas Akhir ini.
4. Bapak Ir. Rivon Tridesman, S.T, I.P.M selaku Dosen Pembimbing dari PT Badak NGL yang telah memberikan bimbingan dalam penyelesaian Tugas Akhir ini.
5. Bapak Ir. Eko Wahyu Susilo, S.T, I.P.M selaku Ketua Jurusan konsentrasi Pengolahan Gas yang telah memberikan bantuan dalam mengarahkan dalam pelaksanaan Tugas Akhir ini.
6. Seluruh pekerja MHE, Instrument Section, Lab&EC Section, dan MPTA Section yang telah membantu dalam penyusunan Tugas Akhir kami.
7. Kakak tingkat LNG Academy di berbagai seksi yang telah membantu kelancaran tugas akhir kami.
8. Pihak-pihak yang berasal dari PNJ dan PT Badak NGL yang membantu penyelesaian tugas akhir ini yang tidak kami sebutkan satu persatu.
9. Kedua orang tua yang telah memberikan doa dan dukungan moral kepada penulis sehingga Tugas Akhir ini dapat diselesaikan.
10. Mpu alit dan Raden Mirza Gading Kandiawan selaku teman satu kelompok dalam Tugas Akhir ini

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



1. Teman–teman LNG Academy angkatan IX yang telah membantu dan memberikandukungan dalam proses penyelesaian Tugas Akhir.
2. Teman pendamping, serta pihak lain yang tidak bisa penulis sebutkan satu persatu.

Penulis sangat menyadari betapa banyak kesalahan dan kekurangan yang mungkin ada pada laporan ini. Oleh karena itu, jika pembaca memiliki pesan dan saran mohon disampaikan kepada penulis sebagai rujukan bagi penulis dimasa yang akan datang.

Akhir kata, penulis mengucapkan terima kasih kepada pembaca yang telah meluangkan waktunya untuk membaca laporan ini dan berharap laporan yang disusun ini dapat bermanfaat bagi pembaca juga bagi penulis dan bagi ilmu pengetahuan.



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





Hak Cipta

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

DAFTAR ISI

ABSTRAK..... 6
KATA PENGANTAR ..... 7
DAFTAR ISI ..... 9
DAFTAR GAMBAR ..... 11
DAFTAR TABEL ..... 13
1.1 Latar Belakang ..... 10
1.2 Rumusan Masalah ..... 15
1.3 Tujuan ..... 16
1.4 Batasan Masalah ..... 16
1.5 Manfaat Penulisan ..... 16
1.6 Sistematika Penulisan ..... 17
BAB I PENDAHULUAN ..... 17
BAB II TINJAUAN PUSTAKA ..... 19
BAB III METODE PELAKSANAAN ..... 17
BAB IV BIAYA DAN JADWAL KEGIATAN ..... 17
DAFTAR PUSTAKA ..... 18
LAMPIRAN 18
BAB II ..... 19
TINJAUAN PUSTAKA ..... 19
2.1 Pembangkit Listrik Tenaga Uap..... 19
2.2 Generator DC ..... 20
2.4 Grounding dan Bonding ..... 22
2.5 Circuit Breaker ..... 23
2.6 Power Charge Controller ..... 24
2.7 Relay ..... 25
2.8 Baterai atau Akumulator ..... 27
2.9 Sistem Kontrol PID ..... 28
2.9.1. Sistem Kontrol kalang terbuka ..... 30
2.9.2. Sistem kontrol kalang tertutup ..... 30
2.9.3. Respon Sistem ..... 31
2.9.4. Transfer Function ..... 33
2.9.5. Pengganggu dan ketidakpastian ..... 33
2.10 Motor stepper ..... 34
2.11 Motor Driver TB6600 ..... 34
2.12 Arduino ..... 35
2.13 Sensor Tekanan ..... 36
2.14 Proximity sensor ..... 38
2.15 Gate Valve ..... 39
2.16 Input output ..... 39
2.17 Liquid Crystal Display (LCD) ..... 40
2.18 Solenoid Valve ..... 40
BAB III METODE PENELITIAN ..... 42
3.1 Diagram Alir Pengerjaan ..... 42
3.2 Prosedur Kerja ..... 43



3.3	Teknis Perancangan Sistem .....	44
3.4	Pengambilan Data .....	50
3.5	Analisis Data .....	50

**DAFTAR IV PEMBAHASAN ..... 52**

4.1	Hasil Visual pengerjaan .....	52
4.1	Pemograman dan instalasi arduino .....	52
4.2	Pemasangan konverter sinyal .....	54
4.3	<i>Smoothing signal</i> .....	55
4.4	Instalasi perkabelan .....	57
4.5	Pengujian rangkaian driver dan motor Stepper .....	59
4.6	Pengujian rangkaian sensor tekanan .....	60
4.7	Pemasangan sensor kecepatan generator .....	62
4.8	Pemasangan motor stepper kepada katup .....	64
4.11	Pemodelan kontrol PID .....	67
4.11	Pengambilan Data time-domain .....	67
4.11	Tuning PID dengan metode Cohen-Coon dan Ziegler-Nichols .....	69
4.11	Mencari nilai PID dari plant .....	70
4.11	Mencari Fungsi Transfer dari <i>Plant</i> .....	72
4.11	Analisa Kestabilan sistem .....	74
4.11.6.	PID Tuning .....	77
4.12	Kontrol keamanan .....	79
4.13	Hasil Pengontrolan .....	79
BAB V KESIMPULAN DAN SARAN .....		82
5.1.	Kesimpulan .....	82
5.2.	Saran .....	82

**DAFTAR PUSTAKA ..... 83**

Lampiran 1	84
Lampiran 2	87
Lampiran 2	93
Lampiran 3	94
Lampiran 4	95

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :  
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.  
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta

2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





## DAFTAR GAMBAR

Gambar 2. 1 Sistematika kerja PLTU .....	16
Gambar 2. 2 Generator DC .....	17
Gambar 2. 3 arus DC (a) dan arus AC (b).....	18
Gambar 2. 4 Power Charge Controller.....	20
Gambar 2. 5 Relay.....	21
Gambar 2. 6 Penggolongan relay.....	22
Gambar 2. 7 Pengontrol kalang terbuka.....	26
Gambar 2. 8 Grafik respon sistem.....	28
Gambar 2. 9 Bagian Arduino .....	32
Gambar 2. 10 Sensor Tekanan .....	34
Gambar 2. 11 Solenoid valve .....	37
Gambar 3. 1 Diagram alir pengerjaan .....	38
Gambar 3. 2 Perancangan sistem .....	40
Gambar 3. 3 Process Flow Diagram.....	40
Gambar 3. 4 Perancangan Sistem Control Valve .....	44
Gambar 4. 1 Bagan dari konverter .....	50
Gambar 4. 2 Data yang belum di smoothing, dan yang sudah di smoothing.....	52
Gambar 4. 3 Pemasangan Stepper motor pada katup.....	60
Gambar 4. 4 Pemodelan PID menggunakan stepper motor .....	62
Gambar 4. 5 Data open loop menggunakan time-domain data.....	63
Gambar 4. 6 Fungsi data streamer pada Excel.....	63
Gambar 4. 7 Grafik data PID dengan nilai Kp 600 dan Ki 1 .....	65
Gambar 4. 8 Hasil pendekatan orde 1 terhadap data yang sudah didapatkan .....	67
Gambar 4. 9 Estimasi nilai kedekatan fungsi transfer terhadap data yang asli.....	68
Gambar 4. 10 Manipulasi blok unity feedback .....	69
Gambar 4. 11 memasukkan numerator untuk analisa kestabilan sistem.....	70
Gambar 4. 12 memasukkan Denominator untuk analisa kestabilan sistem .....	70
Gambar 4. 13 Memasukkan kode untuk memulai fungsi root locus.....	70
Gambar 4. 14 Grafik diagram root locus .....	71
Gambar 4. 15 Pemodelan block diagram di dalam simulink .....	73

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



Gambar 4. 16 Nilai PID yang didapat dari blok simulink..... 73  
 Gambar 4. 17 grafik yang didapat dari tuning menggunakan simulink..... 74



© Hak Cipta milik Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## DAFTAR TABEL

Tabel 4. 1 Pin arduino .....	47
Tabel 4. 2 pengujian motor stepper berdasarkan sudut aktuasi .....	54
Tabel 4. 3 Pengujian sensor tekanan .....	55
Tabel 4. 4 Spesifikasi Sensor Tekanan .....	56
Tabel 4. 5 Spesifikasi proximity sensor .....	58
Tabel 4. 6 Spesifikasi motor stepper .....	59
Tabel 4. 7 Rumus PI metode Cohen-Coon.....	64
Tabel 4. 8 Tabel rumus PI metode Ziegler-Nichols .....	64





Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## BAB I

### PENDAHULUAN

#### 1.1 Latar Belakang

*Corporate Social Responsibility (CSR)* Badak LNG berfokus pada empat pilar: Program Pemberdayaan, Program Peningkatan Kapasitas, Program Amal dan Program Infrastruktur. Fokus utama Badak LNG adalah pada program pemberdayaan atau pemberdayaan masyarakat dengan fokus utama diharapkan dapat terwujud di masyarakat sekitar cincin 1, cincin 2, dan cincin 3 serta di cincin luar yang berpartisipasi sebagai mitra Badak LNG menjadi masyarakat yang mandiri.

Permasalahan sampah kertas berupa dokumen rahasia yang seharusnya dimusnahkan secara terkontrol untuk mencegah tersebarnya data perusahaan menjadi hal yang perlu diperhatikan. Dokumen ini secara berkala dalam program 6 bulan-an harus dimusnahkan. Pada *Corporate Communication Department* misalnya, terdapat 1-3 kg sampah kertas yang diproduksi per hari nya. Badak LNG melalui program CSR-nya bekerja sama dengan mahasiswa LNG Academy dalam membantu melakukan pengembangan alat pembakar sampah atau incinerator non-industrial untuk menangani permasalahan sampah ini.

Pada awalnya ide pembuatan waste burner hadir melalui usulan dari President Director & CEO Badak LNG, Bapak Gema Iriandus Pahalawan untuk memanfaatkan oli bekas yang melimpah di masyarakat dan tergolong murah untuk dijadikan sebagai bahan bakar untuk mengatasi masalah sampah rumah tangga. Namun, seiring berjalannya waktu dan dilakukan diskusi dalam pelaksanaan pengembangannya. Ditemukan bahwa waste burner ini dapat diaplikasikan dan digunakan sebagai solusi pemusnah masalah sampah dokumen confidential dari suatu perusahaan.

Pada kesempatan lain pun mengingat kebijakan Badak LNG terkait limbah domestic seperti limbah waste cooking oil (WCO) yang diproses menjadi biodiesel oleh *Nursery* Badak LNG, mendorong kami untuk mengangkat topik pemanfaatan biodiesel sebagai bahan bakar waste burner ini, selain karena ketersediaannya yang



dapat digunakan untuk masyarakat di kota Bontang, biodiesel juga menghasilkan emisi yang jauh lebih kecil dibandingkan dengan oli bekas.

Umumnya pembakar sampah dengan kapasitas tinggi membutuhkan blower sebagai pemasok udara untuk bahan bakarnya, sedangkan pembakar sampah yang akan dikembangkan memanfaatkan *steam* menggunakan teknologi *steam ejector* menjadi salah satu opsi pada desain dari *waste burner* ini sebagai pendorong dari api dan untuk menarik uap bahan bakar yang berada pada reservoir bahan bakar.

Adapun pengembangan dari *waste burner* ini ialah dengan mengutilisasi panas hilang yang ada sebagai boiler untuk memutar steam turbine sebagai penggerak dari generator. Di lain sisi, terdapat isu kehilangan energi dalam bentuk panas yang hilang. Perlu adanya upaya penanganan akan fenomena ini dengan memanfaatkan panas menjadi bentuk energi yang lain sebagai bentuk optimisasi proses. Namun proses tersebut tidak akan tercapai tanpa adanya beberapa proses kontrol terhadap alat yang akan digunakan. Sehingga untuk mencapai sebuah keharmonisan sistem, dan juga faktor keselamatan, maka diperlukannya sistem kontrol, yang nantinya akan menunjang proses keberjalanan dari *waste burner*, dan juga pembangkit listrik itu sendiri. Dengan sejumlah kontrol yang akan ditambahkan, maka akan menaikkan efisiensi dari operasi penghasil listrik dan memudahkan operator dalam mengoperasikan alat yang akan dibuat. Dengan menambahkan kontrol otomatisasi, maka operator tidak perlu banyak melakukan kerja yang mana nantinya semua proses dapat bekerja dengan sendirinya.

Kontrol yang dibuat juga merupakan sebuah konsep sistem yang biasa digunakan di skala industrial, yakni kontrol berbasis PID, dengan demikian diharapkan keluaran dari hasil kontrol dapat bekerja dengan baik sebagaimana kontrol pada industri yang dapat menghasilkan produk yang baik juga.

## 1.2 Rumusan Masalah

Pada Tugas Akhir ini yang akan diangkat menjadi permasalahan utama adalah melakukan kontrol pada uap air baik dari incinerator maupun yang berasal dari tangki air atau *water jacket* pada bejana bakar sampah. Dengan melakukan kontrol tersebut diharapkan uap air bertekanan dari hasil pembakaran sampah mampu memutar turbin dan dapat menghasilkan tegangan keluaran dari generator untuk melakukan pengisian daya pada akumulator atau baterai. Kontrol yang dilakukan merupakan kontrol PID terhadap tekanan pada incinerator dan juga jumlah putaran



dari generator. Uap air pada *water jacket* akan ditahan hingga tekanan tertentu sebelum melepas tekanan tersebut ke dalam turbin.

### 1.3 Tujuan

- Tujuan umum:
  - 1) Sebagai salah satu syarat untuk menyelesaikan program Pendidikan Diploma III Politeknik Negeri Jakarta.
  - 2) Mengembangkan ilmu pengetahuan dan teknologi khususnya dalam bidang listrik instrumentasi, serta mengaplikasikan ilmu yang telah diperoleh selama mengikuti proses belajar mengajar.
- Tujuan khusus:
  - 1) Merancang instalasi instrumentasi untuk pengontrolan miniplant pembangkit listrik tenaga uap.
  - 2) Merancang sistem kontrol PID untuk besaran tekanan dan kecepatan, serta safety system pada pembangkit listrik sehingga sistem dapat menjadi lebih ekonomis, efisien, dan efektif.
  - 3) Menganalisa performa dan kestabilan sistem kontrol PID pada miniplant pembangkit listrik tenaga uap

### 1.4 Batasan Masalah

Batasan masalah tugas akhir ini adalah sebagai berikut:

- 1) Studi ini hanya berfokus pada perancangan sistem instrumentasi dan kontrol pada miniplant pembangkit listrik tenaga uap
- 2) Besaran yang akan dikontrol yaitu tekanan pada incinerator dan water jacket serta kecepatan putar pada generator
- 3) Turbin, generator, pengisian daya, dan penyimpanannya, merupakan peralatan siap pakai sehingga tidak dilakukan observasi mendalam dan lebih lanjut

### 1.5 Manfaat Penulisan

Manfaat penulisan laporan tugas akhir ini adalah sebagai berikut:

- Bagi Penulis
  - a) Sebagai syarat untuk memenuhi penyusunan Tugas Akhir guna mendapatkan gelar Diploma III dari Program Studi Teknik Konversi Energi di Politeknik Negeri Jakarta.



**Hak Cipta :**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

- b) Menambah pengalaman dan keterampilan dalam merancang bangun suatu alat industri.
- c) Dapat mengimplementasikan pengetahuan yang telah diperoleh selama masa perkuliahan dengan mempraktikkannya secara nyata.

- Bagi LNG Academy dan Politeknik Negeri Jakarta  
Sebagai media pembelajaran sistem waste burner hybrid bertenantkan steam dari hasil pembakaran sampah dan program Kerjasama LNG Academy dengan *Corporate Communication Departement*
- Bagi PT Badak NGL
  - a) Berkontribusi dalam program CSR Badak LNG.
  - b) Berkontribusi dalam program lingkungan (PROPER) PT Badak NGL.
  - c) Menambah sumber energi listrik yang handal di lingkungan PT Badak NGL.
  - d) Mengurangi biaya penyimpanan biodiesel di Nursery Badak LNG.

## 1.6 Sistematika Penulisan

### BAB I PENDAHULUAN

BAB I menguraikan latar belakang pemilihan topik, perumusan masalah, tujuan umum dan khusus, ruang lingkup penelitian dan batasan masalah, lokasi objek tugas akhir, manfaat yang akan didapat, dan sistematika penulisan keseluruhan proposal tugas akhir.

### BAB II TINJAUAN PUSTAKA

BAB II menguraikan studi pustaka atau literatur, memaparkan rangkuman kritis atas pustaka yang menunjang penyusunan atau penelitian, meliputi pembahasan tentang topik yang akan dikaji lebih lanjut dalam tugas akhir.

### BAB III METODE PELAKSANAAN

BAB III menguraikan tentang metodologi, yaitu metode yang digunakan untuk menyelesaikan masalah atau penelitian, meliputi prosedur, pengambilan sampel dan pengumpulan data, pengumpulan data, teknik analisis data atau teknis perancangan.

### BAB IV BIAYA DAN JADWAL KEGIATAN

BAB IV menguraikan tentang :



- Ringkasan anggaran biaya sebagai berikut:
  1. Peralatan penunjang, yang ditulis sesuai kebutuhan;
  2. Bahan habis pakai, yang ditulis sesuai dengan kebutuhan;
  3. Biaya akomodasi; dan
  4. Lain-lain: administrasi, publikasi, seminar, laporan, lainnya.
- Jadwal Kegiatan  
 Berisi urutan kegiatan pembuatan tugas akhir mulai dari pengajuan proposal sampai dengan pengajuan sidang dalam satuan minggu perkuliahan (menggunakan standar format timeline).

#### DAFTAR PUSTAKA

Daftar pustaka adalah suatu susunan tulisan di akhir sebuah karya ilmiah yang isinya berupa nama penulis, judul tulisan, penerbit, identitas penerbit, dan tahun terbit. Daftar pustaka ini digunakan sebagai sumber atau rujukan seorang penulis dalam berkarya

#### LAMPIRAN

Lampiran merupakan dokumen tambahan yang ditambahkan ke dokumen utama. Lampiran dapat ditemukan dalam surat maupun dalam buku. Lampiran surat dapat berupa teks, seperti dokumen pendukung.

© Hak Cipta milik Politeknik Negeri Jakarta

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



POLITEKNIK  
NEGERI  
JAKARTA



Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

## BAB V

### KESIMPULAN DAN SARAN

#### 1. Kesimpulan

1. Telah berhasil dirancang instalasi instrumentasi untuk pengontrolan pembangkit listrik tenaga uap. Hal tersebut ditandai dengan kontrol yang sudah berjalan sesuai dengan rencana. Instalasi tersebut tentunya memiliki beberapa kekurangan seperti deadband pada katup yang mengakibatkan motor stepper tidak dapat melakukan *micro-adjustments*. Pemasangan dan perancangan tersebut sudah dilakukan dengan jalan kontrol yang baik.
2. Telah dibuat sistem yang dapat melakukan kontrol terhadap tekanan, kecepatan dan safety system. Titik pencapaian didapatkan dengan sistem yang ekonomis dari sisi bahan bakar, efisien dari sisi keluaran generator, dan efisien dari sisi otomasinya. Pengontrolan tersebut dicapai dengan menggunakan scenario kontrol PID dengan nilai  $K_p$  dan nilai  $K_i$  terbaik yakni untuk kontrol tekanan pada *incinerator* sebesar 600 dan 1, sedangkan untuk kontrol kecepatan generator, sebesar 2.5 dan 1. Dengan nilai tersebut didapatkan bahwa sistem sudah mencapai titik optimal dengan keluaran masing-masing sistem mencapai *setpoint* nya masing-masing
3. Didapatkan bahwa 2 buah kontrol PID yang digunakan, memiliki steady-state yang stabil. Analisa tersebut dilakukan dengan menggunakan metode root-locus, yang fungsi transfer nya sudah melewati fungsi *unity feedback*. Semua titik pole-zero plot nya berada di sisi negatif atau sebelah kiri dari bidang, menandakan bahwa sistem sudah stabil.

#### 5.2. Saran

1. Perlunya penanganan backlash dari katup, agar kontrol dapat dijalankan lebih presisi.
2. Dibutuhkan komponen yang dapat mengamankan keluaran dari turbin agar terhindar dari kecelakaan kerja

## DAFTAR PUSTAKA

- Haifeng Fan, “*Design Tips for an Efficient Non-Inverting Buck-Boost Converter*”, *Analog Application Journal*, 2014.
- Muhammad Reza Faisal, “ Rancang Bangun Pengendalian Flow Menggunakan *motorized operated valve (MOV)* Berbasis PLC pada Process Control Plant” Tugas Akhir, D3 Metrologi dan Instrumensi, Institut Teknologi Sepuluh Nopember, 2016
- Roy Franc J. S, “Perancangan Turbin Uap Untuk PLTGU dengan Daya Generator Listrik 80 MW pada Putaran Turbin 3000 RPM”, Skripsi, Program Pendidikan Sarjana Ekstensi, Teknik Mesin, Universitas Sumatera Utara, Medan, 2009.
- Graham C. Goodwin, Stefan F. Graebe, Mario E. Salgado, “Control System Design, Prentice Hall PTR”
- [5] John W. Webb Ronald A. Reis “Programmable Logic Controllers, Fourth Edition, Prentice Hall PTR”
- [6] Silalahi, Dony Araventa.2017. Pengendalian Kecepatan Putar (Rpm) Motor DC dengan Metode PID Berbasis Mikrokontroler Atmega328 Menggunakan Bahasa Pemrograman Code Vision Avr. Tugas Akhir. Medan : Universitas Sumatera Utara
- [7] Susanto, Erwin. Kontrol proporsional integral derivatif (PID) untuk motor dc menggunakan personal computer. Departemen Teknik Elektro, Institut Teknologi Telkom Bandung
- [8] Matthew Scarpio, “Motors For Makers”, 2016

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta







## Lampiran 1

### Source code untuk kontrol tekanan incinerator

```
//LCD config
#include "max6675.h"
#include <Smoothed.h> // Include the library
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <AccelStepper.h>
LiquidCrystal_I2C lcd(0x3f,16,2); //sometimes the adress is not 0x3f. Change to 0x3f if it doesn't work.
AccelStepper myStepper (1,8,9);
#define lamready 7
#define buttonint 2
#define setpointup 3
#define setpointdown 4
//Inputs and outputs
int firing_pin = 3;
int myStepper_pul = 6;
int myStepper_dir = 7;
const int sensorPin = A0;
float analogRaw = 0.0;
float voltageRaw = 0.0;
float realPressure = 0.0;
float real_pressure = 0.0;
//Variables
int last_CH1_state = 0;
bool newData, runallowed = false;
int firing_delay = 7400;
int directionMultiplier = 1; // = 1: positive direction, = -1: negative direction
long receivedSteps = 0; //Number of steps
long receivedSpeed = 0; //Steps / second
long receivedAcceleration = 0; //Steps / second^2
float real_pressure_smoothed = 0;
float readPressure(){
  analogRaw = analogRead(sensorPin);
  voltageRaw = (analogRaw * 5.0 / 1024.0) - 0.42;
  real_pressure = (voltageRaw * 16.0 / 4.62 + 0.20);
  return voltageRaw ;
}
char receivedCommand;
int curPos = 0;
int nexPos = 0;

unsigned long previousMillis = 0;
unsigned long currentMillis = 0;
int temp_read_Delay = 500;
float setpoint = 1.75;
bool pressed_1 = false;
bool pressed_2 = false;
int maxRotate = 6000;
//PID variables
float PID_error = 0;
float previous_error = 0;
float elapsedTime, Time, timePrev;
float PID_value = 0;
//PID constants
float kp = 600 ; float ki= 0.7; float kd = 0;
float PID_p = 0; float PID_i = 0; float PID_d = 0;
int scale = 1000;
unsigned long prevmillis = 0;
Smoothed <float> mySensor;

void setup() {
  //Define the pins
  Serial.begin(9600);
  pinMode (firing_pin,OUTPUT);
  pinMode (lamready, OUTPUT);
  pinMode (buttonint, INPUT);
  pinMode (setpointup, INPUT_PULLUP);
  pinMode (setpointdown, INPUT_PULLUP);
  lcd.init(); //Start the LC communication
  lcd.backlight(); //Turn on backlight for LCD
```

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

myStepper.setMaxSpeed(6000); //SPEED = Steps / second
myStepper.setAcceleration(6000); //ACCELERATION = Steps /(second)^2
myStepper.disableOutputs(); //disable outputs
mySensor.begin(SMOOTHED_AVERAGE, 15);
myStepper.setCurrentPosition (0);
myStepper.moveTo (0);
}

void loop() {
  RunTheMotor(); //function to handle the motor
  analogRaw = analogRead(0);
  voltageRaw = (analogRaw * 5.0 / 1024.0) - 0.92;
  real_pressure = (voltageRaw * 40.0 / 4.62) + 0.58;
  mySensor.add(real_pressure);

  currentMillis = millis();
  if(currentMillis - previousMillis >= temp_read_Delay){
    previousMillis += temp_read_Delay;
    real_pressure_smoothed = mySensor.get();
    PID_error = setpoint - real_pressure_smoothed;

    PID_p = kp * PID_error; //Calculate the P value
    PID_i = PID_i + (ki * PID_error); //Calculate the I value
    timePrev = Time; // the previous time is stored before the actual time read
    Time = millis(); // actual time read
    elapsedTime = (Time - timePrev) / 1000;
    PID_d = kd*((PID_error - previous_error)/elapsedTime); //Calculate the D value
    PID_value = (PID_p + PID_i + PID_d); //Calculate total PID value

    //Print the values on the LCD
    if (prevmillis >= millis){
      prevmillis = 250;
    }
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Set: ");
    lcd.setCursor(5,0);
    lcd.print(setpoint);
    lcd.setCursor(0,1);
    lcd.print("Pressure: ");
    lcd.setCursor(11,1);
    lcd.print(real_pressure_smoothed);

    Serial.println(real_pressure_smoothed);
    previous_error = PID_error; //Remember to store the previous error.
    newData= true;
  }
  //
  // Serial.println ("pid error");
  // Serial.println (PID_error);
  //
  // Serial.println ("D");
  // Serial.println (PID_d);

  //If the zero cross interruption was detected we create the 100us firing pulse
  if (newData)
  // Serial.println("PID_value");
  // Serial.println(PID_value);
  {
    if (PID_value > maxRotate){
      nexPos = maxRotate;
    }else if (PID_value <(-1 * maxRotate)){
      nexPos = -1 * maxRotate;
    }else{
      nexPos= PID_value;
    }
    //value for the steps
    //Run the function
  }
  RotateRelative();
  // Serial.println(readPressure());
  // delay(1000);

  if(real_pressure_smoothed >= 1.7){

```



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

digitalWrite (lamready, HIGH);
}

if (digitalRead(setpointup) == LOW)
{
  setpoint = setpoint + 1;
}

if (digitalRead(setpointdown) == LOW)
{
  setpoint = setpoint - 1;
}
}

void RotateRelative() {
// Serial.println("curPos");
// Serial.println(curPos);

  if (nexPos > curPos){
    receivedSteps = nexPos - curPos;
    curPos = nexPos;
  }else if (nexPos < curPos){
    receivedSteps = -nexPos - curPos;
    curPos = nexPos;
  }else{
    return 0;
  }

  // Serial.println("received Steps");
  // Serial.println(receivedSteps);
  // Serial.println("nexPos");
  // Serial.println(nexPos);
  runallowed = true; //allow running - this allows entering the RunTheMotor() function.
  myStepper.setMaxSpeed(6000); //set speed
  myStepper.move(receivedSteps); //set relative distance and direction

  // Serial.println("STEPS");
  // Serial.println(receivedSteps);
}

void RunTheMotor() //function for the motor
{
  if (runallowed == true)
  {
    myStepper.enableOutputs(); //enable pins
    myStepper.run(); //step the motor (this will step the motor by 1 step at each loop)
  }
  else //program enters this part if the runallowed is FALSE, we do not do anything
  {
    myStepper.disableOutputs(); //disable outputs
    return 0;
  }
}
}

```





## Lampiran 2

### Source Code untuk kontrol kecepatan generator

```
//LCD config

#include <Smoothed.h> // Include the library
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <AccelStepper.h>
LiquidCrystal_I2C lcd(0x27,20,4); //sometimes the adress is not 0x3f. Change to 0x3f if it doesn't work.
AccelStepper myStepper (1,8,9);
AccelStepper myStepper1 (1,6,7);
int firing_pin = 3;
int myStepper_pul = 6;
int myStepper_dir = 7;
const int sensorPin = A0;
float analogRaw = 0.0;
float voltageRaw = 0.0;
float realPressure = 0.0;

//Variables
int last_CH1_state = 0;
bool newData, runallowed = false;
int firing_delay = 7400;
int directionMultiplier = 1; // = 1: positive direction, = -1: negative direction
long receivedSteps = 0; //Number of steps
long receivedSpeed = 0; //Steps / second
long receivedAcceleration = 0; //Steps / second^2
float real_pressure = 0.0;
float real_tacho_smoothed = 0;
char receivedCommand;
int curPos = 0;
int nexPos = 0;
float real_pressure_smoothed = 0;

float readPressure(){
  analogRaw = analogRead(sensorPin);
  voltageRaw = (analogRaw * 5.0 / 1024.0) - 0.92;

  Serial.println((voltageRaw * 16.0 / 4.62) + 0.20);
  return (voltageRaw * 16.0 / 4.62) + 0.20;
}
const byte PulsesPerRevolution = 2;
const unsigned long ZeroTimeout = 100000;
const byte numReadings = 2;

volatile unsigned long LastTimeWeMeasured;
volatile unsigned long PeriodBetweenPulses = ZeroTimeout+1000; // Stores the period between pulses in microseconds.
// It has a big number so it doesn't start with 0 which would be interpreted as a high frequency.
volatile unsigned long PeriodAverage = ZeroTimeout+1000; // Stores the period between pulses in microseconds in total, if
we are taking multiple pulses.
// It has a big number so it doesn't start with 0 which would be interpreted as a high frequency.
unsigned long FrequencyRaw; // Calculated frequency, based on the period. This has a lot of extra decimals without the
decimal point.
unsigned long FrequencyReal; // Frequency without decimals.
unsigned long RPM; // Raw RPM without any processing.
unsigned int PulseCounter = 1; // Counts the amount of pulse readings we took so we can average multiple pulses before
calculating the period.

unsigned long PeriodSum; // Stores the summation of all the periods to do the average.

unsigned long LastTimeCycleMeasure = LastTimeWeMeasured; // Stores the last time we measure a pulse in that cycle.
// We need a variable with a value that is not going to be affected by the interrupt
// because we are going to do math and functions that are going to mess up if the values
// changes in the middle of the cycle.
unsigned long CurrentMicros = micros(); // Stores the micros in that cycle.
// We need a variable with a value that is not going to be affected by the interrupt
// because we are going to do math and functions that are going to mess up if the values
// changes in the middle of the cycle.

unsigned int AmountOfReadings = 1;

unsigned int ZeroDebouncingExtra; // Stores the extra value added to the ZeroTimeout to debounce it.
// The ZeroTimeout needs debouncing so when the value is close to the threshold it
// doesn't jump from 0 to the value. This extra value changes the threshold a little
```

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta



### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

// when we show a 0.

// Variables for smoothing tachometer:
unsigned long readings[numReadings]; // The input.
unsigned long readIndex; // The index of the current reading.
unsigned long total; // The running total.
unsigned long average; // The RPM value after applying the smoothing.

unsigned long prevmillis = 0;
unsigned long previousMillis = 0;
unsigned long currentMillis = 0;
int temp_read_Delay = 500;
int real_temperature = 0;
int setpoint = 5000;
bool pressed_1 = false;
bool pressed_2 = false;
int maxRotate = 3500;

const int minSample = 5;
int sampleCount = 0;
const float trigger = 7.0;
const float rangeTrigger = 0.1;
bool startTrigger = false;
bool startTrigger1 = false;
bool firstRun = true;

const int minSample1 = 5;
int sampleCount1 = 0;
const float trigger1 = 5.0;
const float rangeTrigger1 = 0.1;

//PID variables
float PID_error = 0;
float previous_error = 0;
float elapsedTime, Time, timePrev;
int PID_value = 0;
//PID constants
int kp = 15; int ki = 0.5; int kd = 0;
int PID_p = 0; int PID_i = 0; int PID_d = 0;

Smoothed <float> mySensor;
Smoothed <float> mySensor1;
void Pulse_Event() // The interrupt runs this to calculate the period between pulses:
{
  PeriodBetweenPulses = micros() - LastTimeWeMeasured; // Current "micros" minus the old "micros" when the last pulse
  happens.
  // This will result with the period (microseconds) between both pulses.
  // The way is made, the overflow of the "micros" is not going to cause any issue.

  LastTimeWeMeasured = micros(); // Stores the current micros so the next time we have a pulse we would have something to
  compare with.

  if(PulseCounter >= AmountOfReadings) // If counter for amount of readings reach the set limit:
  {
    PeriodAverage = PeriodSum / AmountOfReadings;
    PulseCounter = 1;
    PeriodSum = PeriodBetweenPulses;
    int RemapedAmountOfReadings = map(PeriodBetweenPulses, 40000, 5000, 1, 10); // Remap the period range to the
    reading range.

    RemapedAmountOfReadings = constrain(RemapedAmountOfReadings, 1, 10); // Constrain the value so it doesn't go below
    or above the limits.
    AmountOfReadings = RemapedAmountOfReadings; // Set amount of readings as the remaped value.
  }
  else
  {
    PulseCounter++; // Increase the counter for amount of readings by 1.
    PeriodSum = PeriodSum + PeriodBetweenPulses; // Add the periods so later we can average.
  }
} // End of Pulse_Event.

```





### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

void setup() {
  //Define the pins
  Serial.begin(9600);
  attachInterrupt(digitalPinToInterrupt(2), Pulse_Event, RISING); // Enable interruption pin 2 when going from LOW to
  HIGH.

  delay(1000); // We sometimes take several readings of the period to average. Since we don't have any readings
  // stored we need a high enough value in micros() so if divided is not going to give negative values.
  // The delay allows the micros() to be high enough for the first few cycles.
  lcd.init(); //Start the LC communication
  lcd.backlight(); //Turn on backlight for LCD
  myStepper.setMaxSpeed(1000); //SPEED = Steps / second
  myStepper.setAcceleration(1000); //ACCELERATION = Steps / (second)^2
  myStepper.disableOutputs(); //disable outputs
  mySensor.begin(SMOOTHED_AVERAGE, 12);
  mySensor1.begin(SMOOTHED_AVERAGE, 18);
  myStepper1.setCurrentPosition(0);
  myStepper.setCurrentPosition(0);
}

void loop() {
  //-----
  if (real_pressure_smoothed > trigger - rangeTrigger && real_pressure_smoothed < trigger + rangeTrigger){
    sampleCount += 1;
  }

  if (real_pressure_smoothed > trigger1 - rangeTrigger1 && real_pressure_smoothed < trigger1 + rangeTrigger1){
    sampleCount1 += 1;
  }

  if (sampleCount > minSample){
    startTrigger = true;
    sampleCount = 0;
  }

  if (sampleCount1 > minSample){
    if (firstRun == true){
    }else{
    startTrigger1 = true;
    }
    sampleCount1 = 0;
  }

  RotateMotor();

  analogRaw = analogRead(0);
  voltageRaw = (analogRaw * 5.0 / 1024.0) - 0.92;
  real_pressure = (voltageRaw * 16.0 / 4.62) + 0.54;
  mySensor1.add(real_pressure);

  currentMillis = millis(); //Save the value of time before the loop

  real_pressure_smoothed = mySensor.get();

  //-----
  LastTimeCycleMeasure = LastTimeWeMeasured; // Store the LastTimeWeMeasured in a variable.
  CurrentMicros = micros(); // Store the micros() in a variable.

  if (CurrentMicros < LastTimeCycleMeasure)
  {
    LastTimeCycleMeasure = CurrentMicros;
  }

  FrequencyRaw = 10000000000 / PeriodAverage; // Calculate the frequency using the period between pulses.

  if (PeriodBetweenPulses > ZeroTimeout - ZeroDebouncingExtra || CurrentMicros - LastTimeCycleMeasure > ZeroTimeout -
  ZeroDebouncingExtra)
  {
    FrequencyRaw = 0;
    ZeroDebouncingExtra = 2000;
  }
}

```





### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

else
{
  ZeroDebouncingExtra = 0;
}

FrequencyReal = FrequencyRaw / 10000; // Get frequency without decimals.

// Calculate the RPM:
RPM = FrequencyRaw / PulsesPerRevolution * 60; // Frequency divided by amount of pulses per revolution multiply by
// 60 seconds to get minutes.
RPM = RPM / 10000; // Remove the decimals.

total = total - readings[readIndex]; // Advance to the next position in the array.
readings[readIndex] = RPM; // Takes the value that we are going to smooth.
total = total + readings[readIndex]; // Add the reading to the total.
readIndex = readIndex + 1; // Advance to the next position in the array.

if (readIndex >= numReadings) // If we're at the end of the array:
{
  readIndex = 0; // Reset array index.
}

// Calculate the average:
average = total / numReadings; // The average value it's the smoothed result.

// Print information on the serial monitor.
// Comment this section if you have a display and you don't need to monitor the values on the serial monitor.
// This is because disabling this section would make the loop run faster.

RunTheMotor(); //function to handle the motor

currentMillis = millis(); //Save the value of time before the loop
/* We create this if so we will read the temperature and change values each "temp_read_Delay"
 * value. Change that value above iv you want. The MAX6675 read is slow. Tha will affect the
 * PID control. I've tried reading the temp each 100ms but it didn't work. With 500ms worked ok.*/
if (currentMillis - previousMillis >= temp_read_Delay){
  previousMillis += temp_read_Delay; //Increase the previous time for next loop
  mySensor.add(average);
  real_tacho_smoothed = mySensor.get();
  PID_error = real_tacho_smoothed - setpoint; //Calculate the pid ERROR

// if(PID_error > 30) //integral constant will only affect errors below 30°C
// {PID_i = 0;}

  PID_p = kp * PID_error; //Calculate the P value
  PID_i = PID_i + (ki * PID_error); //Calculate the I value
  timePrev = Time; // the previous time is stored before the actual time read
  Time = millis(); // actual time read
  elapsedTime = (Time - timePrev) / 1000;
  PID_d = kd * ((PID_error - previous_error) / elapsedTime); //Calculate the D value
  PID_value = PID_p + PID_i + PID_d; //Calculate total PID value

//Printe the values on the LCD
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Set: ");
  lcd.setCursor(5,0);
  lcd.print(setpoint);
  lcd.setCursor(0,1);
  lcd.print("Real RPM: ");
  lcd.setCursor(11,1);
  lcd.print(real_tacho_smoothed);
  Serial.println(real_tacho_smoothed);
  previous_error = PID_error; //Remember to store the previous error.
  newData= true;
}

//If the zero cross interruption was detected we create the 100us firing pulse
if (newData)
// Serial.println("PID_value");
// Serial.println(PID_value);
{

```



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengemukakan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

```

if (PID_value > maxRotate){
  nexPos = maxRotate;
}else if (PID_value <(-1 * maxRotate)){
  nexPos = -1 * maxRotate;
}else{
  nexPos= PID_value;
}
//value for the steps
//Run the function
}
RotateRelative();
}

void RotateRelative()
{
  //We move X steps from the current position of the stepper motor in a given direction.
  //The direction is determined by the multiplier (+1 or -1)

  // Serial.println("curPos");
  // Serial.println(curPos);

  if (nexPos > curPos){
    receivedSteps = nexPos - curPos;
    curPos = nexPos;
  }else if (nexPos < curPos){
    receivedSteps = nexPos - curPos;
    curPos = nexPos;
  }else{
    return 0;
  }

  // Serial.println("received Steps");
  // Serial.println(receivedSteps);
  // Serial.println("nexPos");
  // Serial.println(nexPos);

  runallowed = true; //allow running - this allows entering the RunTheMotor() function.
  myStepper.setMaxSpeed(1000); //set speed
  myStepper.move(receivedSteps); //set relative distance and direction
}

void RunTheMotor() //function for the motor
{
  if (runallowed == true)
  {
    myStepper.enableOutputs(); //enable pins
    myStepper.run(); //step the motor (this will step the motor by 1 step at each loop)
  }
  else //program enters this part if the runallowed is FALSE, we do not do anything
  {
    myStepper.disableOutputs(); //disable outputs
    return;
  }
}

void RotateMotor(){
  if (startTrigger == true){
    myStepper1.setMaxSpeed(2000);
    myStepper1.moveTo(maxRotate);
    if (myStepper1.distanceToGo() == 0){
      startTrigger = false;
      firstRun = false;
    }
  }
  else if (startTrigger1 == true){
    myStepper1.setMaxSpeed(2000);
    myStepper1.moveTo(maxRotate * -1);
    if (myStepper1.distanceToGo() == 0){
      startTrigger1 = false;
    }
  }
  myStepper1.enableOutputs();
  myStepper1.run();
}

```

```

}

void RotateMotor1(){
  if (startTrigger == true){
    myStepper.setMaxSpeed(800);
    myStepper.moveTo(maxRotate);
    if (myStepper.distanceToGo() == 0){
      startTrigger = false;
      firstRun = false;
    }
  }
  else if (startTrigger1 == true){
    myStepper.setMaxSpeed(800);
    myStepper.moveTo(maxRotate * -1);
    if (myStepper.distanceToGo() == 0){
      startTrigger1 = false;
    }
  }
  myStepper.enableOutputs();
  myStepper.run();
}

```



## © Hak Cipta milik Politeknik Negeri Jakarta

### Hak Cipta :

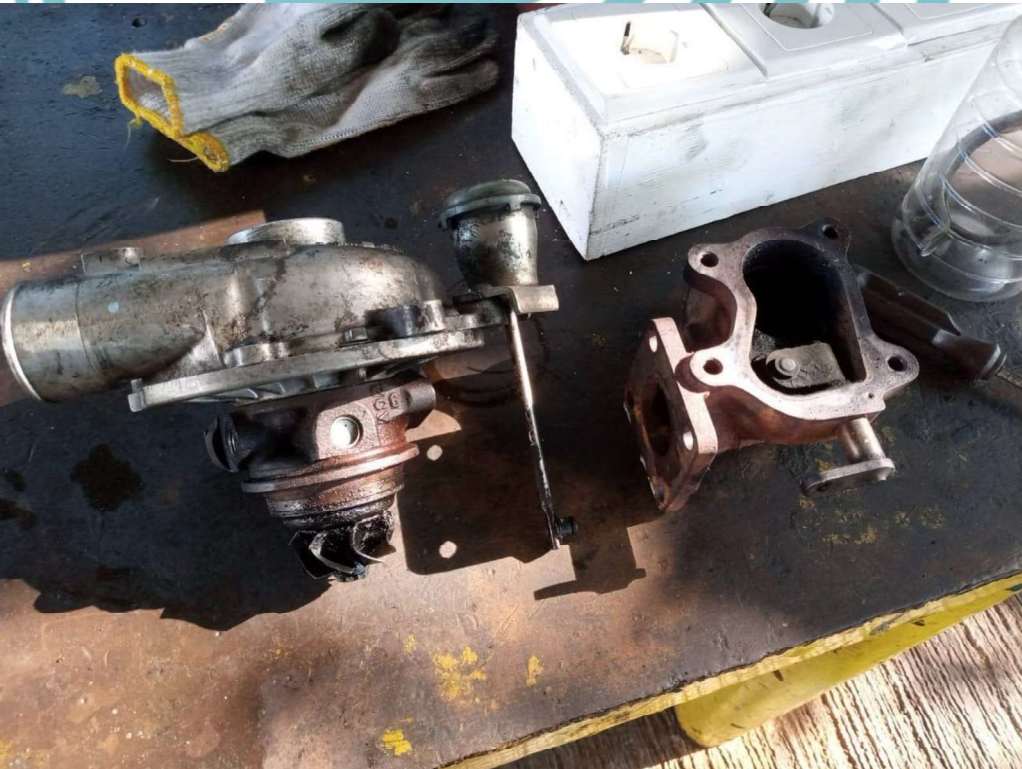
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





## Lampiran 2

### Komponen yang digunakan



© Hak Cipta milik Politeknik Negeri Jakarta

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





### Lampiran 3

#### Penampakan alat secara keseluruhan



© Hak Cipta milik Politeknik Negeri Jakarta

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkannya dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta





## Lampiran 4

### Pemasangan Komponen



© Hak Cipta milik Politeknik Negeri Jakarta

#### Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
  - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
  - b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin Politeknik Negeri Jakarta

