

RANCANG BANGUN SISTEM KENDALI KECEPATAN MOTOR BLDC
DENGAN KONTROL *PROPORTIONAL, INTEGRAL, AND DERIVATIVE* (PID)
BERBASIS ARDUINO



PROGRAM STUDI D-3 TEKNIK ELEKTRONIKA
JURUSAN TEKNIK ELEKTRO
POLITEKNIK NEGERI UJUNG PANDANG
MAKASSAR
2022

HALAMAN PENGESAHAN

Laporan Tugas Akhir ini dengan judul **Rancang Bangun Sistem Kendali Kecepatan Motor BLDC dengan Kontrol Proportional, Integral, and Derivative (PID) Berbasis Arduino** oleh Iffa Kurnia Satira NIM 32319033 dan Indah Maffira NIM 32319034 telah diterima dan disahkan sebagai salah satu syarat untuk memperoleh gelar Diploma Tiga (D-3) pada Program Studi Teknik Elektronika Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang.

Makassar, September 2022

Menyetujui,

Pembimbing I

Fitriaty Panggerang, S.T., M.T
NIP. 19770906 200912 2 001

Pembimbing II

Dharma Aryani, S.T., M.T., Ph.D.
NIP. 19770208 200312 2 001

Mengetahui,



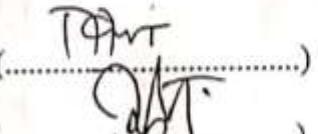
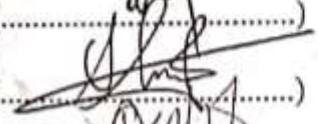
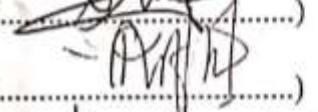
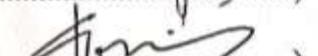
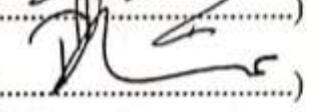
Kartika Dewi, S.T., M.T.
NIP. 19840324 201212 2 003

HALAMAN PENERIMAAN

Pada hari ini, Tim penguji seminar hasil tugas akhir telah menerima hasil seminar hasil tugas akhir oleh mahasiswa Indah Maffira 323 19 034 dan Iffa Kurnia Satira 323 19 033 dengan judul "Rancang Bangun Sistem Kendali Kecepatan Motor BLDC Dengan Kontrol *Proportional, Integral, and Derivative* (PID) Berbasis Arduino".

Makassar, 2022

Tim Penguji Seminar Hasil Tugas Akhir.

- | | | |
|----------------------------------|---------------|--|
| 1. Nur Aminah, S.T.,M.T. | Ketua | (.....
 |
| 2. Kertika Dewi, S.T.,M.T. | Sekretaris | (.....
 |
| 3. Sulaeman, S.T.,M.T. | Anggota | (.....
 |
| 4. Ir. Kifaya, M.T. | Anggota | (.....
 |
| 5. Fitriaty Pangerang, S.T.,M.T. | Pembimbing I | (.....
 |
| 6. Dharma Aryani, S.T.,M.T. | Pembimbing II | (.....
 |

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini :

Nama : Iffa Kurnia Satira

NIM : 32319033

Menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam Laporan Tugas Akhir ini yang berjudul **Rancang Bangun Sistem Kendali Kecepatan Motor BLDC dengan Kontrol *Propotional, Integral, and Derivative (PID)* berbasis Arduino** menggunakan gagasan dan hasil karya sendiri dengan arahan pembimbing, dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi dan instansi manapun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan oleh penulis lain telah disebutkan dalam naskah dan dicantumkan pada daftar pustaka laporan tugas akhir ini.

Jika pernyataan saya tersebut tidak benar, saya siap menanggung resiko yang ditetapkan oleh Politeknik Negeri Ujung Pandang.

Makassar, September 2022



Iffa Kurnia Satira

32319033



SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini :

Nama : Indah Maffira

NIM : 32319034

Menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam Laporan Tugas Akhir ini yang berjudul **Rancang Bangun Sistem Kendali Kecepatan Motor BLDC dengan Kontrol *Propotional, Integral, and Derivative (PID)* berbasis Arduino** menggunakan gagasan dan hasil karya sendiri dengan arahan pembimbing, dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi dan instansi manapun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan oleh penulis lain telah disebutkan dalam naskah dan dicantumkan pada daftar pustaka laporan tugas akhir ini.

Jika pernyataan saya tersebut tidak benar, saya siap menanggung resiko yang ditetapkan oleh Politeknik Negeri Ujung Pandang.

Makassar, September 2022



Indah Maffira

32319034

KATA PENGANTAR

Puji syukur kami panjatkan ke hadirat Allah swt. Karena berkat rahmat dan karunia-Nya, penulisan Laporan Tugas Akhir ini yang berjudul **Rancang Bangun Sistem Kendali Kecepatan Motor BLDC dengan Kontrol Propotional, Integral, and Derivative (PID) berbasis Arduino** dapat diselesaikan dengan cukup baik. Laporan Tugas Akhir ini merupakan hasil penelitian yang dilaksanakan mulai tanggal 26 Januari 2022 sampai dengan Agustus 2022 bertempat di Politeknik Negeri Ujung Pandang.

Dalam penulisan laporan tugas akhir ini tidak sedikit hambatan yang penulis alami. Namun, berkat bantuan berbagai pihak terutama pembimbing, hambatan tersebut dapat teratasi. Sehubungan dengan itu, pada kesempatan dan melalui lembaran ini penulis menyampaikan terima kasih dan penghargaan kepada:

1. Orang tua kami yang selalu setia mendoakan kami dan memberikan dorongan serta motivasi baik moril maupun materil.
2. Bapak Prof. Ir. Muhammad Anshar, M.Si., Ph.D. sebagai Direktur Politeknik Negeri Ujung Pandang.
3. Bapak Ahmad Rizal Sultan, S.T., M.T., Ph.D. sebagai Ketua Jurusan Teknik Elektro.
4. Ibu Kartika Dewi, S.T., M.T. sebagai Ketua Program Studi Teknik Elektronika.
5. Ibu Fitriaty Pangerang, S.T., M.T. sebagai pembimbing I dan Ibu Dharma Aryani, S.T., M.T., Ph.D. sebagai pembimbing II yang telah mencurahkan

waktu dan kesempatannya untuk mengarahkan penulis dalam menyelesaikan Laporan Tugas Akhir ini.

6. Ibu Reski Praminasari, S.T., M.T. selaku Wali Kelas 3B D3 Teknik Elektronika.
7. Bapak/Ibu Dosen Teknik Elektronika Politeknik Negeri Ujung Pandang yang telah membekali ilmu kepada penulis selama mengikuti proses perkuliahan.
8. Teman-teman Kelas 3B Elektronika angkatan 2019 yang telah membantu dalam penyelesaian Laporan Tugas Akhir.
9. Semua pihak yang terkait dalam penulisan Laporan Tugas Akhir.

Penulis menyadari bahwa laporan tugas akhir ini belum sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran yang bersifat membangun demi perbaikan pada masa mendatang. Semoga Laporan Tugas Akhir ini bermanfaat bagi pembacanya.

Makassar, September 2022

Penulis

DAFTAR ISI

KATA PENGANTAR.....	6
DAFTAR ISI.....	8
DAFTAR GAMBAR.....	10
DAFTAR TABEL	11
ABSTRAK	12
BAB I PENDAHULUAN	13
1.1 Latar Belakang	13
1.2 Rumusan Masalah	14
1.3 Ruang Lingkup	14
1.4 Tujuan Penelitian	14
1.5 Manfaat Penelitian	15
BAB II TINJAUAN PUSTAKA.....	16
2.1 Studi Pendahuluan	16
2.2 Motor BLDC (Brushless Direct Current)	18
2.2.1 Konstruksi Motor BLDC	19
2.2.2 Prinsip Kerja Motor BLDC	20
2.3 Sistem Kendali PID (<i>Propotional, Integral, and Derivative</i>).....	22
2.3.1 Karakteristik Sistem Kendali	24
2.3.2 Kendali <i>Propotional</i>	24
2.3.3 Kendali <i>Integral</i>	25
2.3.4 Kendali <i>Derivative</i>	26
2.4 Arduino UNO.....	28
2.5 ESC (Electric Speed Control).....	29
2.6 LM317 DC To DC Converter	30
2.7 Sensor Infrared FC-51	31
2.8 LCD I2C	32

2.9 Potensiometer.....	34
BAB III METODE PERANCANGAN.....	Error! Bookmark not defined.
3.1 Tempat dan Waktu Perancangan	Error! Bookmark not defined.
3.2 Alat dan Bahan	Error! Bookmark not defined.
3.3 Tahap Penelitian.....	Error! Bookmark not defined.
3.3.1 Studi Literatur	Error! Bookmark not defined.
3.3.2 Identifikasi Masalah	Error! Bookmark not defined.
3.4 Perancangan Sistem Perangkat Keras (Hardware).....	Error! Bookmark not defined.
3.4.1 Pembuatan Diagram Blok	Error! Bookmark not defined.
3.4.2 Perancangan Rangkaian	Error! Bookmark not defined.
3.4.3 Flowchart	Error! Bookmark not defined.
BAB IV HASIL DAN PEMBAHASAN	Error! Bookmark not defined.
4.1 Hasil Perancangan.....	Error! Bookmark not defined.
4.2 Spesifikasi Alat	Error! Bookmark not defined.
4.3 Pengujian Alat	Error! Bookmark not defined.
4.3.1 Pengujian Sensor Infrared FC-51	Error! Bookmark not defined.
4.3.1 Pengujian Motor BLDC	Error! Bookmark not defined.
4.4 Hasil Analisis Data	Error! Bookmark not defined.
4.4.1 Hasil Analisis Kurva S.....	Error! Bookmark not defined.
4.4.2 Hasil Analisi dengan PID	Error! Bookmark not defined.
4.4.3 Hasil Analisis tanpa PID	Error! Bookmark not defined.
BAB V KESIMPULAN DAN SARAN	36
5.1 Kesimpulan	36
5.2 Saran	36
DAFTAR PUSTAKA	37
LAMPIRAN	39

DAFTAR GAMBAR

Gambar 2.1 Konstruksi Motor BLDC.....	19
Gambar 2.2 Wiring Diagram Motor BLDC.....	21
Gambar 2.3 Diagram Blok Kontrol PID	23
Gambar 2.4 Blok Diagram Kendali <i>Proportional</i>	25
Gambar 2.5 Diagram Blok Kendali Integral	26
Gambar 2.6 Diagram Blok Kendali <i>Derivative</i>	27
Gambar 2.7 Arduino UNO.....	28
Gambar 2.8 ESC tipe 30A Motor BLDC	30
Gambar 2.9 Regulator Tegangan Variabel LM317 DC To DC Converter	31
Gambar 2.10 Sensor Infrared FC-51.....	32
Gambar 2.11 LCD 12C	33
Gambar 2.12 Struktur, Bentuk, dan Simbol Potensiometer	35

Gambar 3.1 Diagram Blok Proses Perancangan Alat **Error! Bookmark not defined.**

Gambar 3.2 Diagram Blok Sistem Kendali.....**Error! Bookmark not defined.**

Gambar 3.3 Skematik Rangkaian.....**Error! Bookmark not defined.**

Gambar 3.4 (a) Flowchart perangkat (b) Flowchart sistem kendali.....**Error! Bookmark not defined.**

Gambar 4.1 Hasil Perancangan Alat**Error! Bookmark not defined.**

Gambar 4.2 Pengujian Sensor Infrared FC-51**Error! Bookmark not defined.**

Gambar 4.3 Pengujian Motor BLDC**Error! Bookmark not defined.**

Gambar 4.4 Kurva S**Error! Bookmark not defined.**

Gambar 4.5 Respon Kurva S**Error! Bookmark not defined.**

Gambar 4.6 Grafik dengan parameter PID**Error! Bookmark not defined.**

Gambar 4.7 Grafik tanpa PID**Error! Bookmark not defined.**

Gambar 4.8 Grafik perbandingan tanpa PID dan dengan PID **Error! Bookmark not defined.**

DAFTAR TABEL

Tabel 2.1 Spesifikasi Arduino UNO	29
Tabel 3.1 Daftar Alat.....	Error! Bookmark not defined.
Tabel 3.2 Daftar Bahan	Error! Bookmark not defined.
Tabel 4.1 Pengujian sensor infrared FC-51.....	Error! Bookmark not defined.
Tabel 4.2 Karakteristik Motor BLDC	Error! Bookmark not defined.
Tabel 4.3 Nilai parameter K _p , t _i , t _d	Error! Bookmark not defined.
Tabel 4.4 Hasil Analisis dengan PID	Error! Bookmark not defined.
Tabel 4.5 Hasil Analisis tanpa PID	Error! Bookmark not defined.
Tabel 4.6 Perbandingan data dengan PID dan tanpa PID	Error! Bookmark not defined.
Tabel 4.7 Hasil data metode <i>trial error</i>	Error! Bookmark not defined.

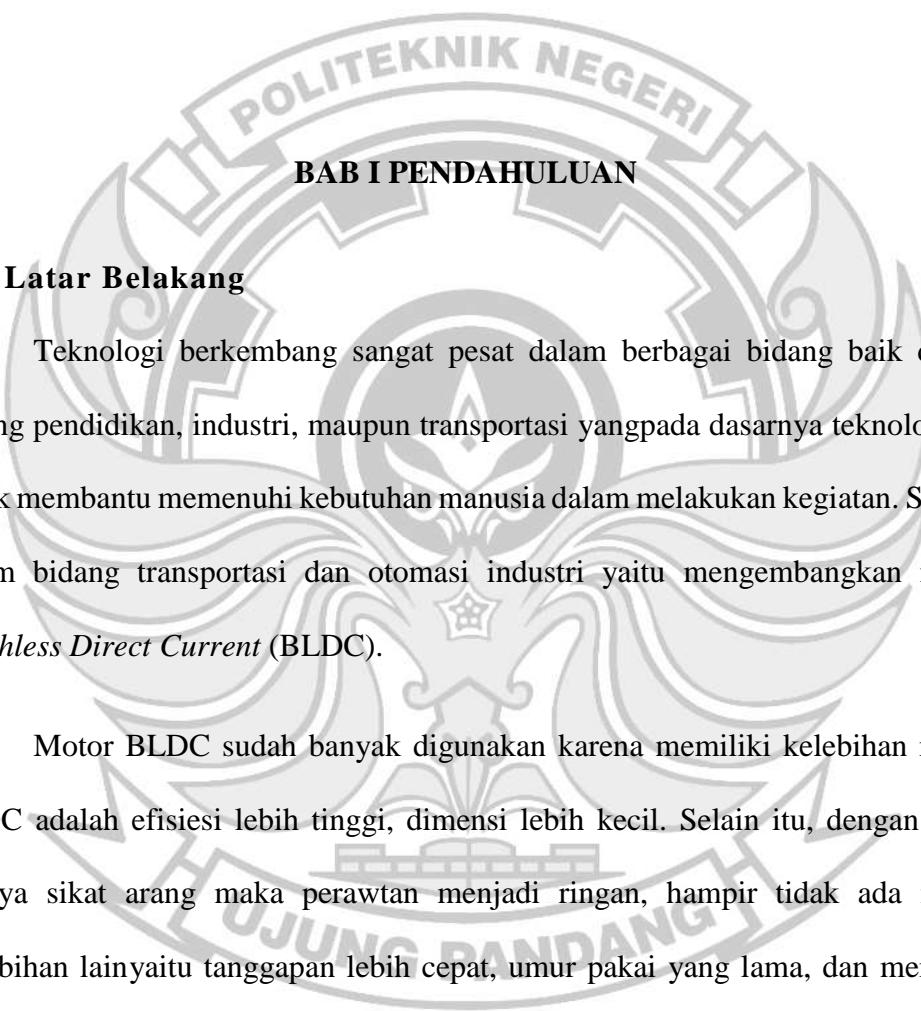
**RANCANG BANGUN SISTEM KENDALI KECEPATAN MOTOR BLDC
DENGAN KONTROL *PROPORTIONAL, INTEGRAL, AND DERIVATIVE*
(PID) BERBASIS ARDUINO**

ABSTRAK

Iffa Kurnia Satira / Indah Maffira, 2022. **Rancang Bangun Sistem Kendali Kecepatan Motor BLDC dengan Kontrol *Propotional, Integral, and Derivative* (PID) berbasis Arduino**, Laporan Tugas Akhir, Program Studi Teknik Elektronika Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang, Pembimbing: Fitriaty Pangerang, S.T., M.T. dan Dharma Aryani, S.T.,M.T.,Ph.D.

Perancangan Tugas Akhir ini bertujuan untuk membuat suatu sistem yang berfungsi mengendalikan kecepatan motor BLDC. Motor BLDC merupakan jenis motor listrik tanpa sikat yang memiliki keunggulan efisiensi tinggi, hemat biaya perawatan serta tidak mengeluarkan suara bising terlalu tinggi karena komutasinya dilakukan secara elektronik tanpa brush (sikat). Dalam penggunaan motor BLDC, diperlukan sistem kontrol untuk mengatur kecepatannya agar mencapai kecepatan yang steady state atau stabil, sehingga lebih efisien dalam penghematan energi. Pada penelitian ini menggunakan sistem kontrol *Propotional, Integral, and Derivative* (PID) dengan memanfaatkan mikrokontroler Arduino Uno. Metode yang digunakan untuk mendapatkan nilai PID adalah *Ziegler-Nichols* Tipe 1. Berdasarkan hasil penelitian menunjukkan bahwa kecepatan motor BLDC dengan kontrol PID mendapatkan parameter *rise time*, *overshoot*, *settling time*, dan *error steady state* yang lebih kecil dibandingkan dengan mengontrol motor BLDC tanpa PID.

Kata kunci : Motor BLDC, Arduino Uno, PID



BAB I PENDAHULUAN

1.1 Latar Belakang

Teknologi berkembang sangat pesat dalam berbagai bidang baik dalam bidang pendidikan, industri, maupun transportasi yang pada dasarnya teknologi ini untuk membantu memenuhi kebutuhan manusia dalam melakukan kegiatan. Seperti dalam bidang transportasi dan otomasi industri yaitu mengembangkan motor *Brushless Direct Current* (BLDC).

Motor BLDC sudah banyak digunakan karena memiliki kelebihan motor BLDC adalah efisiensi lebih tinggi, dimensi lebih kecil. Selain itu, dengan tidak adanya sikat arang maka perawatan menjadi ringan, hampir tidak ada noise. Kelebihan lainnya yaitu tanggapan lebih cepat, umur pakai yang lama, dan memiliki kecepatan yang tinggi. Namun motor BLDC masih memiliki beberapa kekurangan, yaitu ketika motor bekerja dan menerima beban maka akan menyebabkan penurunan kecepatanputaran pada motor BLDC.

Pengendalian kecepatan motor BLDC perlu dilakukan agar respon yang dihasilkan sesuai dengan yang diharapkan. Terdapat beberapa metode untuk mengembalikan respon salah satunya yaitu menggunakan kontrol *Proportional Integral Derivative* (PID). Dengan adanya kontrol PID diharapkan kestabilan kecepatan putaran motor BLDC dapat mencapai *set point*.[Ardiansyah,MD., & Rohman, F.(2019)]

1.2 Rumusan Masalah

Agar lebih terperinci dan lebih mudah dipahami, masalah dirumuskan sebagai berikut:

1. Bagaimana merancang sistem kontrol PID menggunakan Arduino Uno untuk motor BLDC
2. Bagaimana menguji coba sistem kendali kecepatan yang dibuat sehingga sesuai dengan karakteristik kendali menggunakan Arduino Uno untuk motor BLDC.

1.3 Ruang Lingkup

Tugas akhir ini membahas mengenai sistem kendali kecepatan motor BLDC dengan kontrol PID berbasis Arduino Uno yang berfungsi untuk menstabilkan putaran motor BLDC berdasarkan beberapa parameter guna membantu penghematan energi dalam penggunaannya dalam kegiatan sehari-hari.

1.4 Tujuan Penelitian

Berdasarkan rumusan masalah tersebut diatas, tujuan perancangan ini adalah untuk:

- 1). Merancang modul kendali dengan kontrol PID untuk motor BLDC menggunakan Arduino Uno.
- 2). Menguji coba modul kendali yang dibuat sehingga sesuai dengan karakter kendali pada motor BLDC menggunakan Arduino Uno.

1.5 Manfaat Penelitian

Manfaat yang diharapkan dalam pembuatan Tugas Akhir ini yakni

- 1). Dengan kontrol PID dalam mengendalikan kecepatan motor BLDC dapat menghemat energi bagi pengguna motor.
- 2). Dengan kontrol PID mampu dalam menstabilkan kecepatan motor BLDC.
- 3). Dengan adanya sistem kendali yang stabil menjadikan penggunaan motor BLDC lebih efektif.

BAB II TINJAUAN PUSTAKA

2.1 Studi Pendahuluan

Dalam tinjauan pustaka ini diterangkan beberapa hasil penelitian sebelumnya yang terkait dengan sistem kendali motor BLDC. Sistem kendali motor BLDC ini dapat dilakukan menggunakan berbagai metode seperti beberapa penelitian sebagai berikut.

Penelitian yang dilakukan oleh Muh. Ikhsan Hasan dan Hardy Poerwanto AS dengan judul “Rancang Bangun Modul Sistem Kendali Proporsional, Integral, Defrensial (PID) berbasis Mikrokontroler ini”, bertujuan menampilkan respon PID untuk menunjang kelancaran praktikum sistem kendali khususnya praktikum Sistem Kendali Proporsional, Integral, dan Differensial (PID) Perancangan alat ini dilakukan dengan menggunakan mikrokontroler ATMega 8535 sebagai pengolah datanya. Sumber arusnya menggunakan adaptor dengan tegangan sebesar 12V kemudian diturunkan tegangannya menggunakan regulator menjadi 5V.[2]

Selanjutnya penelitian yang dilakukan oleh Bernard Daniel Simanjuntak dengan judul “Analisis Perbandingan Pengaturan Kecepatan Motor Brushless DC dengan Menggunakan Kontrol PI dan PID”, yang dalam penelitian ini dilakukan

perbandingan simulasi motor Brushless DC dengan menggunakan kontrol PI dan PID. Parameter PID (Proportional-Integral-Derivative) diperoleh dengan melakukan simulasi pada rangkaian open loop motor Brushless DC dan menggunakan metode tuning Ziegler Nichols. Kecepatan motor Brushless DC diatur pada beberapa variasi kecepatan dan variasi beban torsi.[3]

Dalam penelitian “Kontrol Kecepatan Motor BLDC dengan PID” oleh Firefly Santi Triwijaya, metode optimasi yang digunakan guna meningkatkan performa dalam pengaturan kecepatan motor adalah dengan PID-firefly. Dimana PID (Proportional–Integral– Derivative controller) adalah salah satu metode kontrol konvensional yang banyak dipakai pada dunia industri. Dimana dalam PID terdapat 3 (tiga) parameter yang harus ditentukan secara tepat dalam penentuan nilainya. Tiga parameter tersebut adalah K_p yaitu konstanta proporsional yang berlaku sebagai penguat (gain), K_i adalah konstanta integral yang berfungsi guna memperbaiki respon steady state, dan K_d adalah konstanta derivatif dapat meminimalkan efek overshoot guna mencapai kondisi steady- state. Dengan algoritma firefly akan didapatkan parameter nilai K_p , K_i , dan K_d yang kemudian digunakan sebagai inputan PID. [4]

Pada penelitian Roedy Kristiyono tentang “Sistem Kendali Kecepatan Motor BLDC Menggunakan Algoritma Hybrid PID Fuzzy” menyatakan bahwa seiring dengan perkembangan sistem berbasis pengetahuan, penalaan konstanta PID ini dapat ditentukan dengan menganalisis tanggapan suatu sistem, hasil analisis ini dibentuk dalam sejumlah aturan. Dengan mengkombinasikan aturan, pengaturan ini menjadi sebuah sistem fuzzy sebagai salah satu sistem berbasis pengetahuan

sehingga konstanta PID dapat ditala secara waktu nyata. Untuk mencapai hal tersebut sistem kendali dibentuk menjadi sistem kendali dua aras. Aras pertama adalah sistem kendali PID konvensional. Aras kedua adalah sistem fuzzy yang menala konstanta PID secara waktu nyata. [5]

Adapun pada penelitian Rafli Fajar Anugrah tentang “Kontrol Motor Brushless DC Menggunakan Six Step Comutation dengan Kontrol PID (Proportional Integral Derivative)” menyatakan bahwa penelitian menggunakan rangkaian inverter Six Step Comutation dan rangkaian PID untuk pengendali kecepatan motor Brushless DC sebagai speed controller. Penelitian yang dilakukan yaitu dengan membuat rangkaian pada PSIM dengan menggunakan motor BLDC. Sumber dari motor BLDC menggunakan tegangan DC kemudian dihubungkan ke rangkaian inverter six step comutation yang sudah terhubung oleh rangkain sistem PID, sehingga tegangan DC akan berubah AC pada keluaran Driver control yang kemudian sebagai input pada motor BLDC. Rangkain motor BLDC akan diberi beban torsi yang berbeda untuk mengetahui perubahan kecepatan pada motor BLDC. Saat pembebanan tersebut rangkaian PID sudah terhubung oleh sensor kecepatan yang akan mendeteksi perubahan kecepatan pada motor BLDC yang kemudian akan menambah tegangan DC untuk menyesuaikan kecepatan motor sesuai set point. [6]

2.2 Motor BLDC (Brushless Direct Current)

Motor BLDC (*Brushless Direct Current*) merupakan motor listrik yang dikembangkan dari motor DC namun pada prinsip kerjanya motor BLDC termasuk jenis motor sinkron. Hal ini berarti medan magnetik yang dihasilkan oleh stator dan medan magnetik yang dihasilkan oleh rotor berputar pada frekuensi yang sama.

Motor BLDC tidak mengalami slip yang biasanya terjadi pada motor induksi. Motor BLDC terdapat dalam konfigurasi 1 fasa, 2 fasa, dan 3 fasa. Berdasarkan tipe stator yang memiliki jumlah kumparan yang sama terdapat dua tipe dari kumparan stator motor BLDC yaitu trapezoidal dan sinusoidal perbedaan ini dibuat berdasarkan dari hubungan kumparan stator yang menghasilkan tipe GGL balik yang berbeda.

Seperti namanya motor BLDC (*Brushless Direct Current*) tidak menggunakan sikat untuk melakukan komutasi tetapi menggunakan komutasi elektris, untuk dapat memutar motor BLDC maka kumparan stator dari motor harus diberi tegangan sesuai urutan komutasi, supaya dapat melakukan hal tersebut maka posisi rotor harus dapat diketahui. Posisi rotor ini dapat diketahui dengan menggunakan sensor *Hall effect* atau dengan teknik sensorless yaitu dengan mendekripsi GGL balik pada kumparan stator. Pada umumnya motor BLDC memiliki tiga sensor *Hall effect* yang terpasang dekat stator dan memiliki kontroler elektronik dalam motor untuk mengatur komutasi daya yang mengalir dalam kumparan stator.

2.2.1 Konstruksi Motor BLDC

Konstruksi motor Brushless DC terdiri dari 3 buah komponen utama yaitu rotor dalam bentuk magnet permanen, stator dalam bentuk kumparan, dan hall effect yang berfungsi dalam proses komutasi yang ditunjukkan pada Gambar 2.1.



Gambar 2.1 Konstruksi Motor BLDC

Hall effect seperti yang ditunjukkan pada Gambar 2.1 merupakan komponen elektronika daya yang digunakan sebagai sensor untuk mendeteksi perubahan posisi pada medan magnet yang dihasilkan kumparan stator, juga sebagai komutasi elektronik pada motor Brushless DC. Pada motor Brushless DC diperlukan 3 buah sensor Hall dan atau encoder. Pada sensor Hall, timing komutasi ditentukan dengan cara mendeteksi medan magnet rotor dengan menggunakan 3 buah sensor hall untuk mendapat 6 kombinasi timing yang berbeda, sedangkan pada encoder, timing komutasi ditentukan dengan cara menghitung jumlah pola yang ada pada encoder.

2.2.2 Prinsip Kerja Motor BLDC

Motor Brushless DC atau motor BLDC adalah salah satu jenis motor sinkron magnet permanen yang disuplai oleh sumber listrik DC pada kontrolnya, dan membutuhkan sumber listrik AC tiga fasa untuk menggerakan bagian rotor motornya. Sumber listrik AC 3 fasa dibutuhkan karena motor sinkron magnet permanen ini memiliki 3 buah koil pada stator. Akibat arus yang melewati kumparan pada stator timbul medan magnet (B) :

$$B = \frac{\mu Nl}{2i}$$

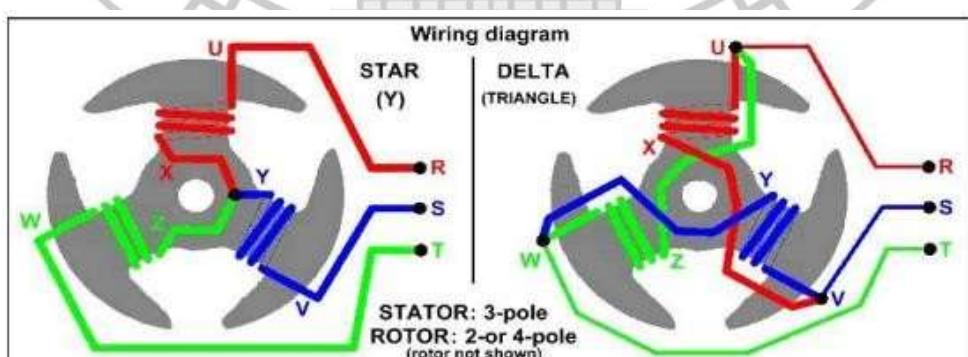
Dimana N merupakan jumlah lilitan, i merupakan arus, l merupakan panjang lilitan dan μ merupakan permeabilitas bahan.

Karena arus yang diberikan merupakan arus AC, maka nilai medan magnet dan polarisasi setiap kumparan akan berubah-ubah. Akibat dari adanya perubahan polarisasi dan besar medan magnet tiap kumparan maka terjadilah medan putar stator yang besarnya :

$$N_S = \frac{120f}{p}$$

Dimana f merupakan frekuensi tegangan input yang dinyatakan dalam Hz, p merupakan jumlah kutub (pole) pada rotor. Motor BLDC bekerja berdasarkan adanya perbedaan polaritas antara kumparan medan pada stator dengan polaritas magnet permanen pada rotor. [6]

Pada Gambar 2.2, sumber tegangan arus bolak-balik 3 fasa dihubungkan dengan masing-masing kumparan stator pada motor. Hal ini akan menimbulkan adanya medan magnet pada masing-masing kumparan. Karena dihubungkan dengan sumber tegangan arus bolak-balik, maka polaritas medan magnet yang timbul juga akan berubah-ubah mengikuti perubahan polaritas tegangan sumber.



Gambar 2.2 Wiring Diagram Motor BLDC

Ketika kumparan stator U diberikan tegangan negatif maka akan timbul medan magnet dengan polaritas negatif sedangkan kumparan stator V dan W yang diberikan tegangan positif akan memiliki polaritas positif. Akibat adanya perbedaan polaritas antara medan magnet stator dan magnet rotor, sisi positif magnet rotor akan berputar mengikuti medan magnet stator V dan W. Karena sumber tegangan yang digunakan adalah tegangan arus bolak-balik, medan magnet stator U, V, dan W akan berubah-ubah polaritasnya dan besarnya mengikuti perubahan polaritas sumber tegangan. Ketika U dan V memiliki medan magnet negatif karena mendapatkan tegangan negatif dan W memiliki medan magnet positif karena tegangan positif, magnet permanen rotor akan berputar menuju ke polaritas yang bersesuaian yakni bagian negatif akan berputar menuju medan magnet stator W dan sebaliknya bagian positif akan berputar menuju medan magnet stator U dan V. Selanjutnya ketika V memiliki medan magnet negatif dan U serta W memiliki medan magnet positif, bagian positif magnet permanen akan berputar menuju V dan bagian negatif akan menuju U dari kumparan W. Karena sumber tegangan arus bolak-balik yang digunakan berlangsung secara kontinu, maka proses perubahan polaritas tegangan pada stator ini akan terjadi secara terus menerus sehingga menciptakan medan putar magnet stator dan magnet permanen rotor akan berputar mengikuti medan putar magnet stator ini. Hal inilah yang menyebabkan rotor pada BLDC dapat berputar.[6]

2.3 Sistem Kendali PID (*Propotional, Integral, and Derivative*)

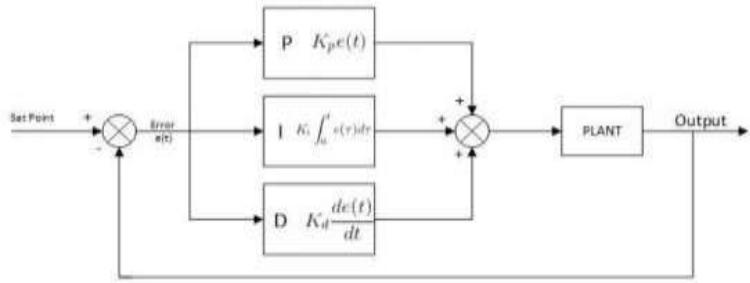
Keberadaan kendali dalam sebuah sistem kendali mempunyai kontribusi yang besar terhadap perilaku sistem. Pada prinsipnya hal itu disebabkan oleh tidak

dapat diubahnya komponen penyusun sistem tersebut. Artinya, karakteristik plant harus diterima sebagaimana adanya sehingga perubahan perilaku sistem hanya dapat dilakukan melalui penambahan suatu sub-sistem yang disebut kendali.

Menurut Pakpahan (1992) bahwa Sistem kendali ialah suatu proses pengaturan atau pengendalian terhadap satu atau beberapa besaran (variabel parameter), sehingga berada pada suatu harga atau dalam suatu rangkaian harga (range) tertentu. Sedangkan menurut Tarmukan (1995) bahwa "Sistem pengendalian ialah suatu sistem yang memperoleh pengendali pada besaran fisiknya melalui pengendali masuknya".[2]

Sistem kendali PID merupakan pengendali yang umum digunakan dalam berbagai macam proses industri. Popularitas kendali PID disebabkan khusus karena performansinya yang baik dalam jangkauan yang lebar dari berbagai kondisi operasi dan khususnya dalam kesederhanaan fungsi PID, yang memungkinkan *engineer* untuk mengoperasikannya secara simpel dan langsung. Untuk mengimplementasikan pengendali PID, tiga parameter harus ditentukan pada proses yang dikendalikan yang meliputi proporsional, integral dan diffrensial gain.[2]

Kontroler PID merupakan kontroler untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Pengontrol PID adalah pengontrol konvensional yang banyak dipakai dalam dunia industri.



Gambar 2.3 Diagram Blok Kontrol PID

Sebuah kontroler PID secara kontinu menghitung nilai kesalahan sebagai beda antara set point yang diinginkan dan variabel proses terukur. Kontroler mencoba untuk meminimalkan nilai kesalahan dengan penyetelan variabel kontrol.[2]

2.3.1 Karakteristik Sistem Kendali

Karakteristik utama yang harus dimiliki oleh sistem kendali adalah meliputi sebagai berikut:

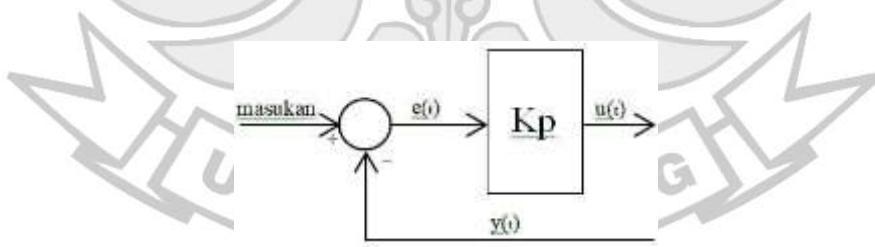
1. Ketelitian ; Ketelitian sistem pengendali diperbaiki dengan menggunakan sistem kendali seperti integral. Dengan penggunaan umpan balik (feedback) ketelitian sistem semakin diperbaiki.
2. Kestabilan ; Sistem dikatakan stabil jika keluarannya tetap pada nilai tertentu dalam jangka waktu yang ditetapkan setelah diberi masukan.
3. Sensitivitas ; Sensitivitas merupakan ukuran kepekaan keluaran sistem terhadap perubahan harga komponen-komponennya, dan juga kondisi lingkungannya. Sistem yang baik hanya akan terpengaruh oleh perubahan masukan dan bukan terhadap sinyal yang tidak diinginkan.
4. Kehandalan ; Kehandalan merupakan kecukupan performance suatu sistem atau alat dalam mencapai kepuasan kerja dalam waktu tertentu.

2.3.2 Kendali Propotional

Karakteristik aksi pengontrolan proporsional adalah mengurangi wakiu naik, menambah overshoot, dan mengurangi kesalahan dalam keadaan tunak.

Pengendali proporsional memiliki keluaran yang sebanding dengan besarnya sinyal kesalahan (selisih antara besaran yang di inginkan dengan harga aktualnya). Secara lebih sederhana dapat dikatakan bahwa keluaran kendali proporsional merupakan perkalian antara konstanta proporsional dengan masukannya. Perubahan pada sinyal masukan akan segera menyebabkan sistem secara langsung mengeluarkan output sinyal sebesar konstanta pengalinya.[2]

Gambar 2.4 menunjukkan blok diagram yang menggambarkan hubungan antara besaran setting, besaran aktual dengan besaran keluaran kendali proporsional. Sinyal kesalahan (error) merupakan selisih antara besaran setting dengan besaran aktualnya. Selisih ini akan mempengaruhi kendali, untuk mengeluarkan sinyal positif (mempercepat pencapaian harga setting) atau negatif (memperlambat tercapainya harga yang diinginkan).



Gambar 2.4 Blok Diagram Kendali *Propotional*

Keterangan : Kp adalah konstanta proporsional.

Kelebihan kendali proporsional yaitu memiliki respon yang cepat. Namun demikian, harga variabel kendalinya tidak pernah mencapai harga yang sama dengan harga masukan acuannya sehingga terjadi *steady state error*.

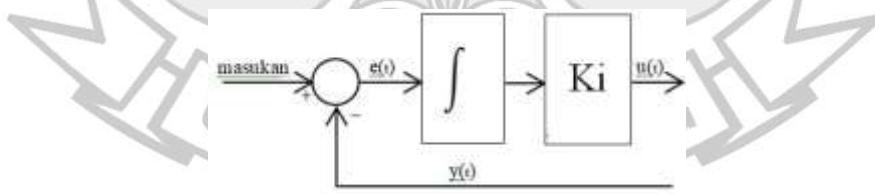
Dalam rangkaian kendali proporsional menggunakan operasional complifier (Op-Amp) yang berfungsi untuk membangkitkan sinyal proporsional Sinyal input (Vin) dihasilkan oleh set point Untuk memperoleh tegangan keluaran (Vout). [2]

2.3.3 Kendali Integral

Kendali integral berfungsi menghasilkan respon sistem yang memiliki kesalahan keadaan stabil nol. Jika sebuah plan tidak memiliki unsur integrator (16) kendali proporsional tidak akan mampu menjamin keluaran sistem dengan kesalahan keadaan stabilnya nol. Dengan pengontrol integral respon sistem dapat diperbaiki, yaitu mempunyai kesalahan keadaan stabilnya nol.

Pada kendali integral, harga keluaran kendali C(s) dipengaruhi oleh perubahan yang sebanding dengan sinyal kesalahan R(s). Hubungan antara keluaran kontroler C(s) dengan sinyal kesalahan R(9) adalah:

$$\frac{de(t)}{dt} = Ki(t) \quad \text{atau} \quad C(s) = Ki \int_0^t E(s)dt$$



Gambar 2.5 Diagram Blok Kendali Integral

$$\frac{C(s)}{R(s)} = \frac{Ki}{s}$$

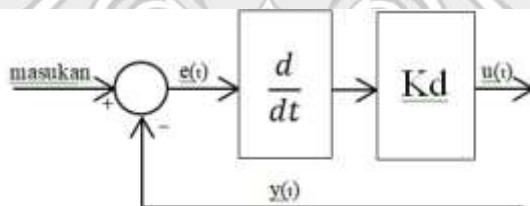
Kelebihan kendali integral yaitu mampu mengeliminasi kesalahan steady state yang terjadi pada kendali proporsional Namun, kendali integral memiliki respon yang

lambat disebabkan laju perubahan keluaran tergantung pada konstanta waktu integralnya.

Rangkaian kendali integral merupakan rangkaian penguat inverting, tetapi mempunyai tahanan umpan balik berupa kapasitor yang bertujuan agar arus umpan balik tetap besarnya dengan arus input pada semua keadaan. Menggunakan operasional amplifier (Op-Amp) yang berfungsi untuk membangkitkan sinyal integrator.[2]

2.3.4 Kendali Derivative

Keluaran kendali derivatif memiliki sifat seperti halnya suatu operasi derivatif. Perubahan yang mendadak pada masukan kendali, akan mengakibatkan perubahan yang sangat besar dan cepat. Gambar 2.6 menunjukkan blok diagram yang menggambarkan hubungan antara sinyal kesalahan dengan keluaran kontroler.



Gambar 2.6 Diagram Blok Kendali Derivative

Secara matematis alat kendali dinyatakan sebagai berikut :

$$K_d \frac{c[n] - c[n-1]}{T}$$

Keterangan : K_d adalah konstanta diferensial

Karakteristik kendali diferensial sebagai berikut :

1. Kendali ini tidak dapat menghasilkan keluaran bila tidak ada perubahan pada masukannya (berupa sinyal kesalahan).

2. Jika sinyal kesalahan berubah terhadap waktu, maka keluaran yang dihasilkan kontroler tergantung pada nilai T_d dan laju perubahan sinyal kesalahan.
3. Kendali diferensial mempunyai mastu karakter untuk mendahului, sehingga kendali ini dapat menghasilkan koreksi yang signifikan sebelum pembangkit kesalahan menjadi sangat besar. Jadi kendali diferensial dapat mengantisipasi pembangkit kesalahan, memberikan aksi yang bersifat korektif, dan cenderung meningkatkan stabilitas sistem.[2]

Berdasarkan karakteristik kendali tersebut, kendali diferensial umumnya dipakai untuk mempercepat respon awal suatu sistem, tetapi tidak memperkecil kesalahan pada keadaan tunaknya. Kerja kendali diferensial hanyalah efektif pada lingkup yang sempit, yaitu pada periode peralihan.

Kontroler turunan :

$$U(t) = K_d \frac{de(t)}{dt}$$

Rangkaian differensiator adalah rangkaian aplikasi dari rumusan matematika yang dapat diatur (dipengaruhi) dari kerja kapasitor. Menggunakan Operasionalamplifier (Op-amp) yang berfungsi untuk membangkitkan sinyal differensiator.[2]

2.4 Arduino UNO

Arduino UNO berbeda dari semua board Arduino sebelumnya, Arduino UNO tidak menggunakan chip driver FTDI USB-to-serial. Sebaliknya, fitur-fitur Atmega16U2 (Atmega8U2 sampai ke versi R2) diprogram sebagai sebuah pengubah USB ke serial. Revisi 2 dari board Arduino Uno mempunyai sebuah resistor yang menarik garis 8U2 HWB ke ground, yang membuatnya lebih mudah

untuk diletakkan ke dalam DFU mode. Model board Arduino Uno ditampilkan pada Gambar 2.7.



Gambar 2.7 Arduino UNO

Tabel 2.1 Spesifikasi Arduino UNO

Mikrokontroler	ATmega328
Tegangan Operasi	5V
Input Voltage (disarankan)	7-12V
Input Voltage (limit)	6-20V
Pin Digital I/O	14 (6 pin digunakan sebagai output PWM)
Arus DC per pin I/O	20 Ma
Flash Memory	32KB (ATmega328P) 512 byte digunakan oleh Bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
Ukuran	68.6mm x 54.4mm

2.5 ESC (Electric Speed Control)



Gambar 2.8 ESC tipe 30A Motor BLDC

Ini memiliki beberapa fungsi:

1. Perlindungan Suhu: Kontrol kecepatan memiliki heat sink aluminium, yang memungkinkannya bekerja pada 30 A terus menerus. Ketika suhu mencapai 110 derajat celcius, sistem pendekripsi suhu akan mengirimkan sinyal perlindungan dan sistem akan secara otomatis menurunkan 60% keluaran daya untuk melindungi kontrol kecepatan dan motor. Ini akan kembali ke daya normal setelah suhu diturunkan. Untuk melindungi modul aktuator, hilangkan panas selama penggunaan dalam waktu lama.
2. Pencegahan kehilangan sinyal: Mati otomatis ketika tidak ada sinyal yang terdeteksi dari generator sinyal PWM (potensiometer).

Spesifikasi:

- Nilai tegangan: DC 6-12.8V
- Peringkat saat ini: 32A (dalam 5 detik, arus maks tidak dapat melebihi 40A)
- Kekuatan nominal: 360W
- Nilai Daya: 5V 0,5A
- Perlindungan suhu: 110 ° C

- Rentang kecepatan: 8% ~ 98%
- Untuk tiang motor: 2-18 tiang
- Mulai: mulai fleksibel
- Rem: beban mundur
- Sinyal kecepatan: PWM frekuensi tetap 8KHz / 16K

2.6 LM317 DC To DC Converter



Gambar 2.9 Regulator Tegangan Variabel LM317 DC To DC Converter

Regulator tegangan variabel merupakan rangkaian regulator yang memiliki tegangan output dapat diubah-ubah sesuai kebutuhan. Rangkaian regulator tegangan variabel pada saat ini telah tersedia dalam bentuk chip IC regulator tegangan variabel 3 pin. Salah satu contoh regulator tegangan variabel adalah IC LM317.

IC LM317 merupakan chip IC regulator tegangan variable untuk tegangan DC positif. Untuk membuat power supply dengan tegangan output variabel dapat dibuat dengan sederhana apabila menggunakan IC regulator LM317. Spesifikasi Regulator Tegangan Variabel LM317 :

- Arus maksimum 1,5 Ampere.
- Dapat memberikan perubahan output dari 1,2 volt sampai 37 volt DC.
- Dilengkapi dengan proteksi dari hubung singkat (short circuit).
- Dilengkapi dengan proteksi over heating (panas berlebih)

2.7 Sensor Infrared FC-51

Modul sensor infrared FC-51 merupakan suatu rangkaian yang digunakan untuk mendeteksi sinar infra merah pada area kerjanya. Dalam rangkaian sensor infrared FC-51 ini terdapat dua buah komponen infrared yaitu pemancar infrared (IR Transmitter) dan penerima infrared (IR Receiver). Pemancar infrared merupakan sebuah photodiode yang dapat memancarkan sinar infra merah, sendangkan penerima infrared merupakan sebuah dioda khusus yang berfungsi sebagai penerima sinar infra merah. Bagian-bagian sensor infrared FC-51 terlihat pada gambar di bawah.



Gambar 2.10 Sensor Infrared FC-51

Pada saat sumber tengangan dihubungkan ke VCC dan GND, maka lampu indikator modul akan hidup (ON). Cara kerja dari sensor infrared FC-51 ini adalah dengan memancarkan sinar infra merah melalui dioda pemancar infra merah. Jika tidak ada benda yang ada di wilayah pANCARAN infra merah, maka tidak ada media yang dapat memantulkan sinar infra merah tersebut. Penerima infra merah tidak akan mendeteksi apapun. Pada keadaan ini, LED indikator sinyal akan mati (OFF) dan sinyal keluaran akan berlogika HIGH (5 V).

Jarak benda yang dideteksi bisa disesuaikan dengan cara memutar potensio (pengatur jarak) agar dapat mendeteksi benda dengan jarak antara 2 cm hingga 15 cm. Sensor infrared FC-51 ini bekerja dengan tegangan 5 volt DC.

Spesifikasi Modul Sensor Infrared FC-51 :

Output level adalah digital output signal (LOW ketika mendeksi hambatan).

Jarak pendeksi adalah 2 cm samapai dengan 30 cm.

Sudut pendeksi adalah 35° Modul ini menggunakan komparator LM393.

2.8 LCD I2C

I2C LCD adalah modul LCD yang dikendalikan secara serial sinkron dengan protokol I2C/IIC (*Inter Integrated Circuit*) atau TWI (*Two Wire Interface*). Normalnya, modul LCD dikendalikan secara parallel baik untuk jalur data maupun kontrolnya. Namun, jalur parallel akan memakan banyak pin di sisi kontroller (misal Arduino, Android, komputer, dll). Setidaknya Anda akan membutuhkan 6 atau 7 pin untuk mengendalikan sebuah modul LCD. Dengan demikian untuk sebuah kontroller yang ‘sibuk’ dan harus mengendalikan banyak I/O, menggunakan jalur parallel adalah solusi yang kurang tepat.



Gambar 2.11 LCD 12C

Sebagai contoh, sebuah Arduino Uno memiliki pin digital sebanyak 13 buah. Jika Anda gunakan separuhnya untuk mengendalikan LCD berarti Anda hanya punya alternatif sekitar 6 atau 7 pin untuk mengendalikan perangkat yang lain, misalnya motor DC, sensor cahaya, keypad, dan I/O devices lainnya. Nah,

sekarang tergantung pada sistem Anda, cukup atau tidak jika harus menggunakan 6/7 pin khusus untuk bekerja dengan LCD saja. Jika tidak cukup, Anda dengan mengubah jalur kendali LCD dari parallel ke serial (I2C) menggunakan modul I2C converter, sehingga Anda hanya akan membutuhkan 2 jalur kabel saja (plus satu kabel ground) untuk menghubungi sang LCD.

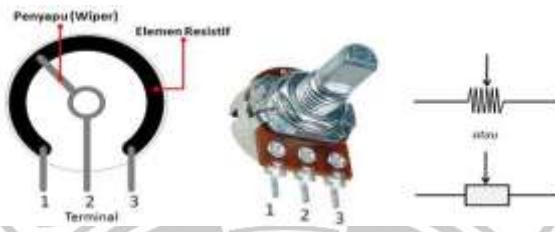
Arduino sendiri sudah mendukung protokol I2C/IIC. Di papan Arduino Uno, port I2C terletak pada pin A4 untuk jalur SDA (Serial Data) dan pin A5 untuk jalur SCL (Serial Clock). Jangan lupa untuk menghubungkan jalur kabel Ground antara Arduino dengan perangkat I2C client. Untuk sisi software, Arduinosudah cukup membantu kita bekerja dengan protokol ini melalui library ‘Wire.h’. Berikutnya, library ini akan dimanfaatkan untuk mengkonversi jalur parallel LCD menjadi jalur serial I2C. Memiliki spesifikasi :

- Power : DC 5V.
- Support LCD 1602 dan 2004 (LCD 16x2, LCD 16x4)
- Kontrol pin : SDA dan SCL.
- Built-in potensio untuk adjust brightness.
- Built-in jumper untuk menon-aktifkan backlight.
- Dimensi : 40mm x 18mm.
- Berat : 20 gram.

2.9 Potensiometer

Potensiometer adalah salah satu jenis Resistor yang nilai resistansinya dapat diatur sesuai dengan kebutuhan rangkaian elektronika ataupun kebutuhan pemakainya. Potensiometer merupakan keluarga resistor yang tergolong dalam

kategori variable resistor. Secara struktur potensiometer terdiri dari 3 kaki terminal dengan sebuah shaft atau tuas yang berfungsi sebagai pengaturnya. Gambar dibawah ini menunjukan struktur internal potensiometer beserta bentuk dan simbolnya.



Gambar 2.12 Struktur, Bentuk, dan Simbol Potensiometer



BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pengujian dan pembahasan analisi data yang telah dilakukan maka dapat disimpulkan bahwa :

1. Perancangan modul kendali proportional, integral, derivative (PID) dan kombinasinya menggunakan Mikrokontroler Arduino Uno sebagai pusat pengelolahan datanya.
2. Berdasarkan hasil analisis kendali kecepatan motor BLDC dengan kontrol PID didapatkan parameter rise time, overshoot, settling time, dan error steady state yang lebih kecil dibandingkan dengan mengontrol motor BLDC tanpa PID.
3. Berdasarkan analisis kendali kecepatan motor BLDC dengan kontrol PID dihasilkan parameter $kp = 0,16$ $kd = 0,03$ dan $ki = 0,005$ yang didapat dengan metode Ziegler-Nichols.

5.2 Saran

Berdasarkan pengujian dan pembahasan yang telah dilakukan maka dapat disarankan bahwa :

1. Untuk percobaan selanjutnya sebaiknya menambahkan beban atau gangguan

pada motor BLDC.

2. Pada penelitian ini parameter yang didapatkan dengan metode *Ziegler- Nichols* yaitu nilai kp, ki, kd, namun pada penelitian selanjutnya dapat diubah dengan menggunakan berbagai metode.

DAFTAR PUSTAKA

- [1] Ardiansyah, M. D., & Rohman, F. (2019). Implementasi Dan Analisis Kendali Kecepatan Motor Bldc 1 Kw Menggunakan Algoritma PID. *JURNAL ELTEK*, 17(2), 81-93.
- [2] HASAN, M., & AS Poerwanto, H. (2012). *Rancang Bangun Sistem Kendali PID Berbasis Mikrokontrol*. Makassar: Politeknik Negeri Ujung Pandang.
- [3] SIMANJUNTAK, B. D. (2019). *Analisis Perbandingan Pengaturan Kecepatan Moto BLDC dengan Menggunakan Kontrol PI dan PID*. Medan: Universitas Sumatera Utara
- [4] TRIWIJAYA, F. S., PRASETYO, Y., & WATI, T. (2021) Kontrol Kecepatan Motor BLDC dengan PID. homepage URL : ejurnal.itats.ac.id/index.php/iptek
- [5] KRISTIYONO, R., WAHYUNGGORO, O., & NUGROHO, P. (2015) Sistem Kendali Kecepatan Motor BLDC Menggunakan Algoritma Hybrid PID Fuzzy. University Research Colloquium 2015

- [6] ANUGRAH, R. F., P.K, I. D., & Dewantara, B. Y. (2020). Kontrol Moto Brushles DC Menggunakan Six Step Comutation dengan Kontro PID (Proportiona Integral Derivative). *Jurnal Tekni Elektr da Kompute Tria Vol. 7 No. 2 Tahun 2020.*
- [7]. Dharmawan, Abe. 2009. Pengendalian Motor Brushless DC Dengan Metode PWM Sinusoidal Menggunakan Atmega 16. Depok: UniversitasIndones
- [8] ALUYO, W., Fitriansyah, A., & SYAHRIAL, S. (2013). Analisis Penalaan Kontrol PID pada Simulasi Kendali Kecepatan Putaran Motor DC Berbeban menggunakan Metode Heuristik. ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika, 1(2), 79.
- [9] Elektro.um.ac.id. (2016, April) Identifikasi Sistem. Diakses pada 24 Agustus 2022, dari https://www.google.com/url?sa=t&source=web&rct=j&url=http://elektro.um.ac.idAOvVaw2UlkQ4S_n9-Gnn
- [10] Kuliah.unpatti.ac.id. (2020, 30 Maret). Sistim Kendali Umpan-Balik. Diakses pada 24 Agustus 2022, dari <https://kuliah.unpatti.ac.id/mod/page/view>.

LAMPIRAN



Gambar 1. Perancangan komponen

```
#include <LiquidCrystal_PCF8574.h>
```

```
#include <Servo.h>
#include <Wire.h>
LiquidCrystal_PCF8574 lcd(0x27);
```

```
Servo motor;
```

```
#define btn1 A1
```

```
#define btn2 A2
```

```
#define btn3 A3
```

```
#define trt A0
```

```
#define sensor 2
```

```
#define pinmotor 9
```

```
float rpm = 0;
```

```
float rpm1 = 0;
```

```
float rpm2 = 0;
```

```
int rpmprev;
```

```
int pid;
```

```
unsigned long millisBefore;
```

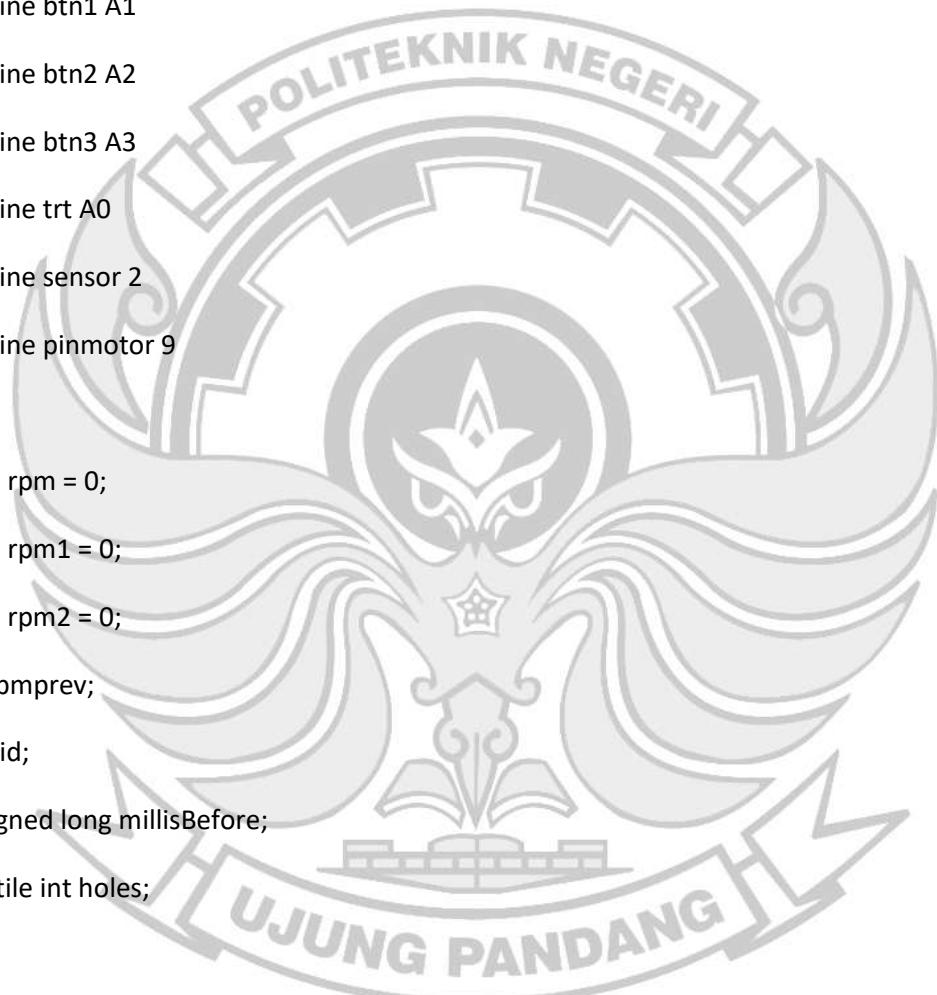
```
volatile int holes;
```

```
double kp = 0.0;
```

```
double ki = 0.0;
```

```
double kd = 0.0;
```

```
unsigned long currentTime, previousTime;
```



```
double elapsedTime;  
  
double error;  
  
double lastError;  
  
double cumError, rateError;  
  
int setPoint=4770;  
  
double out;  
  
int btnVal1 = 0, btnVal2 =0,btnVal3 =0, trtVal =0, starting=0;  
  
int pwm, rotasi;  
  
void setup()  
{  
    Serial.begin(9600);  
  
    lcd.begin(16, 2); lcd.setBacklight(255);  
  
    pinMode(2, INPUT);  
  
    pinMode(btn1,INPUT_PULLUP);  
  
    pinMode(btn2,INPUT_PULLUP);  
  
    pinMode(btn3,INPUT_PULLUP);  
  
    attachInterrupt(digitalPinToInterrupt(2), count, RISING);  
  
    motor.attach(9);  
  
    motor.write(0);  
  
    delay(1000);  
}  
  
void loop()
```

```
{  
    btnVal1=analogRead(btn1);  
    btnVal2=analogRead(btn2);  
    btnVal3=analogRead(btn3);  
    trtVal=analogRead(trt);
```

```
pwm = map(trtVal, 0, 1023, 500, 1000);  
  
if (millis() - millisBefore > 100) {  
    rpm = (holes*1.55)*60;  
    rpm1 = (0.854*rpm1)+(0.0728*rpm)+ (0.0728*rpm2);  
    rpm2 = rpm;  
    holes = 0;  
    millisBefore = millis();  
    rpmprev = rpm1;  
    Serial.println(rpmprev);  
}  
  
if(starting==0){  
    motor.writeMicroseconds(0);  
}  
  
if(starting==1){
```

```
    motor.writeMicroseconds(pwm);  
}
```

```
if(btnVal2<=500){  
    starting=1;  
    delay(100);  
}  
  
}  
  
if(btnVal1<=500){  
    starting=0;  
    delay(100);  
}  
  
lcd.setCursor(0,0);  
lcd.print("S: ");  
lcd.print(trtVal);  
lcd.print("      ");  
lcd.setCursor(0,1);  
lcd.print("RPM: ");  
lcd.print(rpmprev);  
lcd.print("      ");  
}  
  
void count() {  
    holes++;  
}
```



Program 1. Tanpa PID

```
#include <LiquidCrystal_PCF8574.h>  
  
#include <Servo.h>  
  
#include <Wire.h>  
  
LiquidCrystal_PCF8574 lcd(0x27);
```

```
Servo motor;  
  
#define btn1 A1  
  
#define btn2 A2  
  
#define btn3 A3  
  
#define trt A0  
  
#define sensor 2  
  
#define pinmotor 9
```

```
float rpm = 0;  
  
float rpm1 = 0;  
  
float rpm2 = 0;  
  
int rpmprev;  
  
int pid;  
  
unsigned long millisBefore;  
  
volatile int holes;
```

```
double kp = 0.0;  
  
double ki = 0.0;
```

```
double kd = 0.0;

unsigned long currentTime, previousTime;

double elapsedTime;

double error;

double lastError;

double cumError, rateError;

int setPoint=4770;

double out;

int btnVal1 = 0, btnVal2 =0,btnVal3 =0, trtVal =0, starting=0;

int pwm, rotasi;

void setup()

{

    Serial.begin(9600);

    lcd.begin(16, 2); lcd.setBacklight(255);

    pinMode(2, INPUT);

    pinMode(btn1,INPUT_PULLUP);

    pinMode(btn2,INPUT_PULLUP);

    pinMode(btn3,INPUT_PULLUP);

    attachInterrupt(digitalPinToInterrupt(2), count, RISING);

    motor.attach(9);

    motor.write(0);
```

```
delay(1000);

}

void loop()
{
    btnVal1=analogRead(btn1);

    btnVal2=analogRead(btn2);

    btnVal3=analogRead(btn3);

    trtVal=analogRead(trt);

    if (millis() - millisBefore > 100) {

        rpm = (holes*1.55)*60;

        rpm1 = (0.854*rpmprev)+(0.0728*rpm)+(0.0728*rpmprev);

        rpm2 = rpm;

        holes = 0;

        millisBefore = millis();

        rpmprev = rpm1;

        Serial.println(rpmprev);

    }

    currentTime=millis();

    elapsedTime = (double)(currentTime - previousTime);

    if (currentTime - previousTime >= 100){

        error = setPoint - rpm1;

        cumError += error * elapsedTime;

        rateError = (error - lastError)/elapsedTime;
    }
}
```

```
out = ((kp*error)+ (ki*cumError)+ (kd*rateError));
```

```
lastError = error;
```

```
previousTime = currentTime;
```

```
}
```

```
rotasi=850;
```

```
if(starting==0){
```

```
lastError= 0;
```

```
cumError = 0;
```

```
error=0;
```

```
rateError=0;
```

```
motor.writeMicroseconds(0);
```

```
}
```

```
if(starting==1){
```

```
motor.writeMicroseconds(pwm);
```

```
pwm= rotasi + out;
```

```
if(pwm< 510){pwm=510;}
```

```
if(pwm> 1000){pwm=1000;}
```

```
}
```

```
if(btnVal2<=500){
```

```
starting=1;
```

```
delay(100);  
}  
  
if(btnVal1<=500){
```

```
    starting=0;
```

```
    delay(100);
```

```
}
```

```
lcd.setCursor(0,0);
```

```
lcd.print("S: ");
```

```
lcd.print(setPoint);
```

```
lcd.print("      ");
```

```
lcd.setCursor(0,1);
```

```
lcd.print("RPM: ");
```

```
lcd.print(rpmprev);
```

```
lcd.print("      ");
```

```
}
```

```
void count() {
```

```
    holes++;
```

```
}
```



Program 2. Dengan PID