

Pengaturan Kecepatan Motor DC dengan Kontrol Proporsional Integral Derifatif (PID) Berbasis LabView

Motor DC Speed Adjustment By Propotional Integral Derivative (PID) Based on LabView

Alifa Restu Janwar Wiriawan, Andry Irawan

Program Studi Teknik Elektro, Fakultas Teknik dan Ilmu Komputer
Universitas Komputer Indonesia Jl. Dipati ukur No 112, Bandung

Email : ai29326@email.unikom.ac.id

Abstrak – Motor DC banyak digunakan di industri kecil dan besar. Kecepatan motor dc sering tidak stabil akibat gangguan dari luar maupun perubahan parameter dari fabrikasinya sehingga perlu dilakukan rancangan kontroller. Pengaturan dan monitoring kecepatan putar motor DC dalam sebuah sistem proses sangat penting perannya dalam implementasi industri. Pengaturan dan monitoring kecepatan putar motor DC ini menggunakan perangkat antarmuka komputer dimana dalam industri hal ini diperlukan untuk memudahkan operator dalam mengatur dan memonitor kecepatan motor. Agar diperoleh pengontrol yang terbaik, maka dilakukan tuning parameter pengontrol *Proporsional Integral Derifatif* (PID). Dalam tuning ini kita dapat mengetahui nilai dari proporsional gain (Kp), waktu integral (Ti) dan waktu derivatif (Td). Pengontrol PID akan memberikan aksi kepada kontrol motor DC berdasarkan error yang diperoleh, nilai putaran motor DC yang diinginkan disebut dengan set point. Software LabVIEW digunakan sebagai pemonitor, kendali kecepatan motor.

Kata kunci : LabView, Motor DC, Arduino, Ouptocoupler, Komputer.

Abstract - DC motors are widely used in small and large industries. DC motor speed is often unstable due to outside interference and changes in the parameters of the fabrication so it is necessary to design a controller. Motor DC speed adjustment and monitoring is a crucial system as it is implemented in industrial. This motor DC speed adjustment and monitoring using computer interface where in industrial this system will support operator for adjusting and monitoring motor speed. For acquiring best control parameters, tuning is needed for acquiring best Proportional Integral Derivative (PID) value. This tuning is used for find the best proportional gain, time integral, derivative time. PID controller will give a better control respond to the DC Motor based on the error, the DC motor rotation speed needed is called Setpoint. The labview software used as an interface of monitor and control.

Keyword : LabView, Motor DC, Arduino, Ouptocoupler, Computer.

I. PENDAHULUAN

Motor DC telah dikenal lama sejak teori gaya Lorentz dan induksi elektromagnetik ditemukan. Motor DC sering digunakan karena kemudahan dalam aplikasinya sehingga dipakai pada berbagai macam keperluan, mulai dari peralatan industri, rumah tangga, hingga didapati pada mainan anak-anak maupun piranti pendukung dalam sistem instrumen elektronik. Namun pada kenyataannya, kecepatan putar motor DC sulit untuk dikendalikan dikarenakan lajunya yang tidak stabil.[1]

Untuk mengatasi hal ini maka diperlukan suatu perancangan sistem kontrol kecepatan motor DC agar motor DC tersebut bergerak sesuai dengan kecepatan yang diinginkan. Yakni kontroller

Proportional Integral Derivatif (PID) yaitu kontrol yang terdiri dari konfigurasi standar Kp, Ki, dan Kd yang nilainya ditentukan / setting agar mendapatkan hasil atau kecepatan yang diinginkan yaitu kecepatan dengan stabilitas yang baik dengan tingkat error dan *overshoot* (melampaui) yang kecil. Dalam penelitian ini, sistem kontrol PID yang akan dibahas menggunakan prinsip kerja *feedback encoder* dengan optocoupler berbasis Mikrokontroler ATmega8535 yang akan diaplikasikan pada motor DC tanpa beban. Pembahasan ini dilakukan dengan mengamati perubahan pada kontrol PID yang ada pada program labview yang disetting pada keypad serta waktu terbaik (Time Sampling) yang diperoleh guna untuk mendapatkan

kestabilan dari kecepatan motor DC yang diinginkan.

A. Latar Belakang

Dalam pengoperasian motor DC, kecepatan putaran motor tidak konstan. Kecepatan maksimum motor terjadi ketika motor tidak dibebani. Kecepatan motor jauh berkurang karena adanya beban pada motor DC, hal ini mengindikasikan bahwa kecepatan putaran diperlukan adalah kecepatan dimana motor DC sedang mendapat beban penuh. Rentang waktu dimana motor berputar tanpa beban cukup stabil dengan set point yang ditentukan. Jika kecepatan motor pada waktu tersebut dapat diturunkan, konsumsi energi listrik menjadi berkurang. Pengaturan kecepatan dapat dilakukan dengan mengubah nilai set point. Untuk mendapatkan kecepatan putaran yang diharapkan digunakan sistem kontrol dengan pengendali PID. Pengendali PID mudah untuk diterapkan pada sistem arduino pada mikrokontroler.[2] Arduino merupakan sistem yang lebih kompleks dalam pengaturan kecepatan motor dan akan menjaga kecepatan motor DC pada posisi kecepatan set pointnya walaupun beban yang diberikan pada motor DC berubah – ubah.

LabVIEW merupakan software yang khusus digunakan untuk memproses dan visualisasi data dalam bidang akuisisi data, kendali dan instrumentasi, serta otomatisasi industri. Software ini dikembangkan pertama kali oleh perusahaan Nasional Instruments (NI) pada tahun 1986. LabVIEW merupakan singkatan dari Laboratory Virtual Instruments Engineering Workbench.[11] Dalam dunia industri pada umumnya sebuah mesin - mesin yang memproduksi suatu barang menggunakan bantuan teknologi untuk mempermudah menghitung jumlah barang yang dihasilkan dalam skala besar.[3] Hal ini dikarenakan proses perhitungan secara otomatis menggunakan beberapa software seperti LabVIEW lebih mudah dalam mengendalikan dan memonitoring motor. Dalam tugas mata kuliah otomasi industri dibahas mengenai metode pengontrol dan memonitoring putaran pada motor DC menggunakan sensor cahaya atau disebut juga dengan sensor optocoupler dan mengirimkan sinyal ke mikrokontroler arduino yang nantinya akan dibaca oleh LabVIEW sebagai indikator putaran motor DC.

B. Tinjauan State of Art

Pengaturan adalah suatu kegiatan yang sering dilakukan untuk mengendalikan nilai suatu

variable. Pengaturan dapat dilakukan dengan berbagai macam cara dari lup terbuka hingga tertutup.. Pada laporan ini menyajikan perkembangan system kendali lup terbuka untuk pengendalian kecepatan motor DC dengan berbasis mikrokontroler Atmega 8535. Sistem yang dikendalikan adalah sebuah motor DC. Dan yang dikendalikan adalah mengatur kecepatan motor DC dengan teknik Pulse Width Modulation (PWM). Penelitian ini menggunakan PC untuk memberikan perintah ke mikrokontroler yang berfungsi untuk menaikkan atau menurunkan putaran motor DC, sehingga mikrokontroler mengeluarkan sinyal *Pulse Width Modulation* (PWM) ke driver untuk menguatkan arus. Sinyal PWM yang sudah dikuatkan arusnya digunakan untuk menggerakkan motor DC.[4&8]

Pengontrolan PID pada umumnya yang digunakan adalah metode trial-error, untuk menentukan parameter yang tepat untuk PID, namun hasil yang didapat tidak membuat kontroler PID optimal. Belakangan ini sudah banyak penelitian untuk mengoptimasi kontroler PID, salah satunya dengan metode cerdas. Untuk itu pada penelitian ini akan digunakan metode cerdas berbasis *Cuckoo Search Algorithm* (CSA), untuk mengoptimasi dan menentukan parameter yang tepat dari PID. CSA adalah salah satu metode cerdas yang terinspirasi dari perilaku burung cuckoo dalam menempatkan telurnya disarang burung lain yang dia pilih secara acak, konsep inilah yang di adaptasi dan diterapkan menjadi algoritma cerdas untuk menyelesaikan masalah optimasi. Dari hasil yang diperoleh metode CSA dapat dengan baik menala parameter PID, sehingga *overshoot* yang dihasilkan tidak ada dan *settling time* sangat cepat. Pada penelitian ini juga akan dibahas dan dibandingkan kontroler tanpa PID, dengan PID trial-error dan dengan PID CSA.[5&9]

Disamping pengontrolan kecepatan putar, sistem control motor dc juga mengatur arah putar rotor, searah jarum jam atau berlawanan dengan arah jarum jam. Salah satu sistem control motor dc adalah menggunakan Modulasi Lebar Pulsa (PWM) sebagai pemicu pada driver control seperti transistor H-Bridge. Penelitian ini bertujuan untuk merancang sistem control motor dc menggunakan suhu udara sebagai pemicu.[6]

Motor arus searah merupakan salah satu motor listrik yang sering digunakan oleh industri – industri terutama industri yang membutuhkan kecepatan putaran yang konstan. Motor arus searah mempunyai pengaturan yang sangat mudah dilakukan dalam berbagai kecepatan dan beban

yang bervariasi. Pengaturan kecepatan pada motor arus searah dapat dilakukan dengan metode Ward Leonard. Pengaturan putaran motor ini dilakukan dengan mengubah-ubah tegangan terminal yang memiliki daerah pengaturan yang luas. Dalam penelitian ini dilakukan dengan beberapa variasi tegangan dari yang terkecil 20 volt sampai yang terbesar 220 volt untuk mendapatkan hasil putaran yang juga akan bervariasi. Nilai kecepatan putaran tertinggi yang didapat sebesar 1850 rpm ketika diberi tegangan 220 volt dan kecepatan terendah sebesar 750 rpm ketika diberi tegangan 20 volt.[7]

Alat yang digunakan berbeda dengan empat studi literatur di atas, dengan menggunakan sistem pengendali *Proportional Integral Derivative* (PID) yaitu untuk mengatur motor DC agar dapat berputar stabil dan sesuai dengan nilai set point yang telah ditentukan.

C. Tujuan

Adapun tujuan penelitian ini yaitu untuk mengetahui fungsi parameter KP (Konstanta Proporsional), mengetahui fungsi Parameter KI (Konstanta Integral), mengetahui fungsi Parameter KD (Konstanta Derivatif), dan merancang serta membuat alat untuk mengontrol dan memonitoring motor dc menggunakan arduino dan komputer dengan metode PID pada program LabView.

D. Sistematika Pembahasan

Makalah ini diorganisasikan sebagai berikut. bagian 2 akan menjelaskan mengenai pengaturan

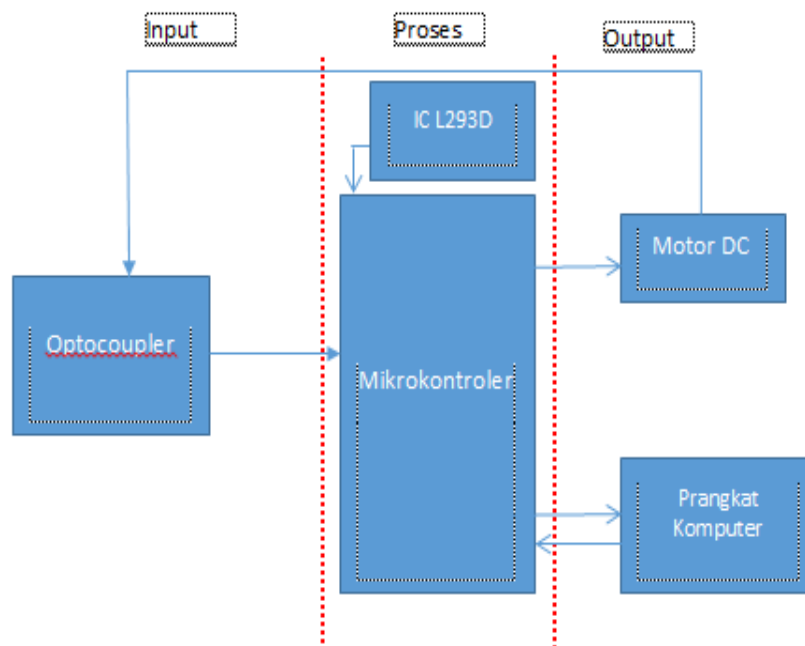
kecepatan motor DC dengan kontrol PID berbasis LabView. Bagian 3 akan menyajikan hasil pengujian dan analisa. Adapun kesimpulan dan saran akan disajikan pada bagian 4.

II. METODOLOGI

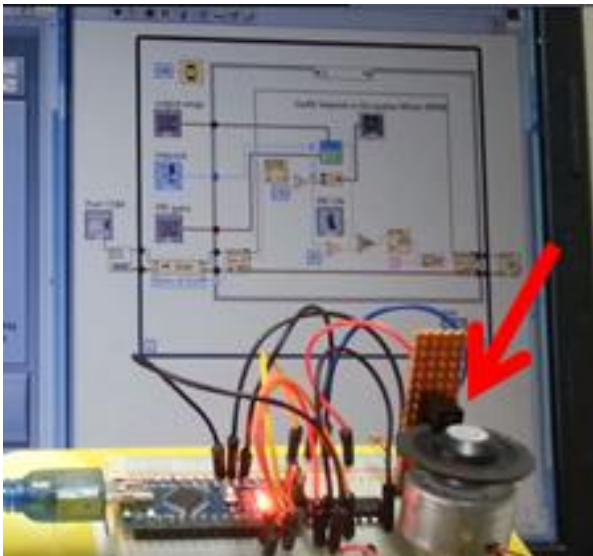
Metode penelitian ini meliputi perancangan perangkat keras (hardware), kendali PID, dan metode pengujian.

A. Perancangan Perangkat Keras

Secara garis besar skema alat kontrol motor DC dapat dilihat seperti pada **Gambar 1**. Pada **Gambar 1** dapat diketahui bahwa sensor optocoupler ini digunakan sebagai input, sensor ini akan menghubungkan berdasarkan cahaya optik karena optocoupler ini terbagi 2 bagian yaitu pertama bagian transmitter yang berfungsi sebagai pengirim cahaya optik dan bagian yang kedua receiver yang berfungsi sebagai pendeteksi sumber cahaya. Dimana nantinya sensor optocoupler ini berfungsi membaca piringan yang terdapat pada motor DC yang memiliki 4 buah lubang sehingga akan terdapat celah dan penghalang, ketika terhalang piringan tersebut maka output akan terbuka, dan sebaliknya apabila tidak terhalang maka output akan short. Dengan cara kerja tersebut sinar inframerah akan putus-putus dan menimbulkan pulsa-pulsa listrik, pulsa-pulsa listrik tersebut selanjutnya akan diolah dan diproses oleh mikrokontroler arduino nano, untuk lebih jelasnya dapat dilihat pada **Gambar 2**.

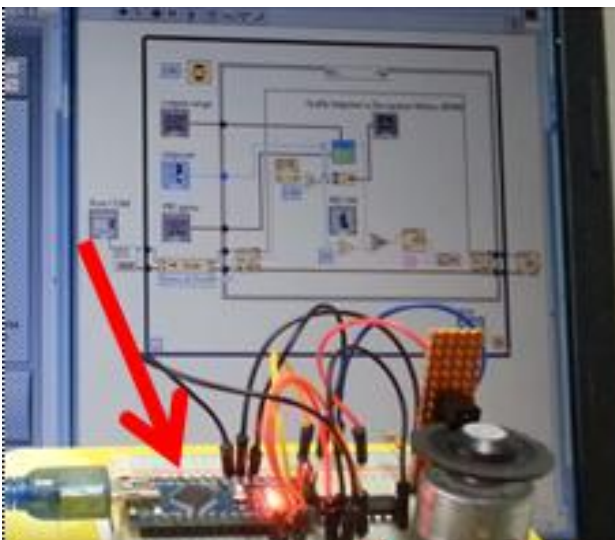


Gambar 1 Block Diagram



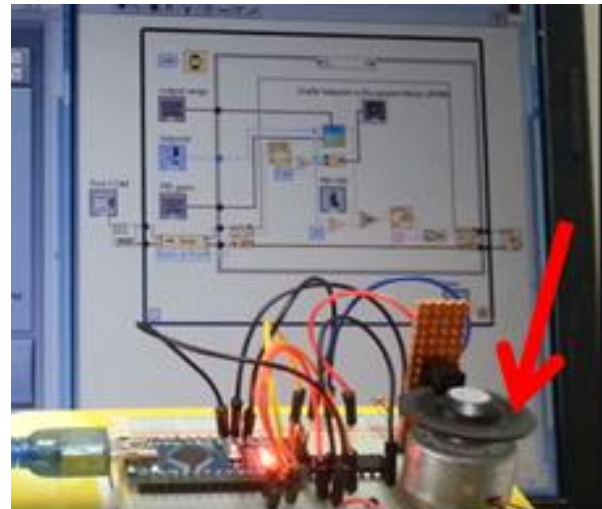
Gambar 2 Sensor Optocoupler

Pada perancangan alat ini menggunakan mikrokontroler, dan mikrokontroler yang digunakan adalah Arduino Nano yaitu digunakan untuk menghitung RPM motor DC sekaligus mengirim nilai RPM kepada program LabView dan pengiriman sinyal PWM kepada driver. Arduino nano juga akan mengolah data dari sensor rotari dan kecepatan motor, melakukan proses logika PID, Mikrokontroler Arduino Nano dapat dilihat pada **Gambar 3**.



Gambar 3 Mikrokontroler Arduino Nano

Motor DC yang digunakan dalam penelitian ini merupakan motor DC 5-12 VDC dengan RPM maksimal tanpa beban 1500 yang merupakan motor DC magnet permanen. Motor DC adalah motor listrik yang memerlukan suplai tegangan arus searah (DC) pada kumparan medan untuk diubah menjadi energi gerak mekanik, untuk lebih jelasnya Motor DC dapat dilihat pada **Gambar 4**.

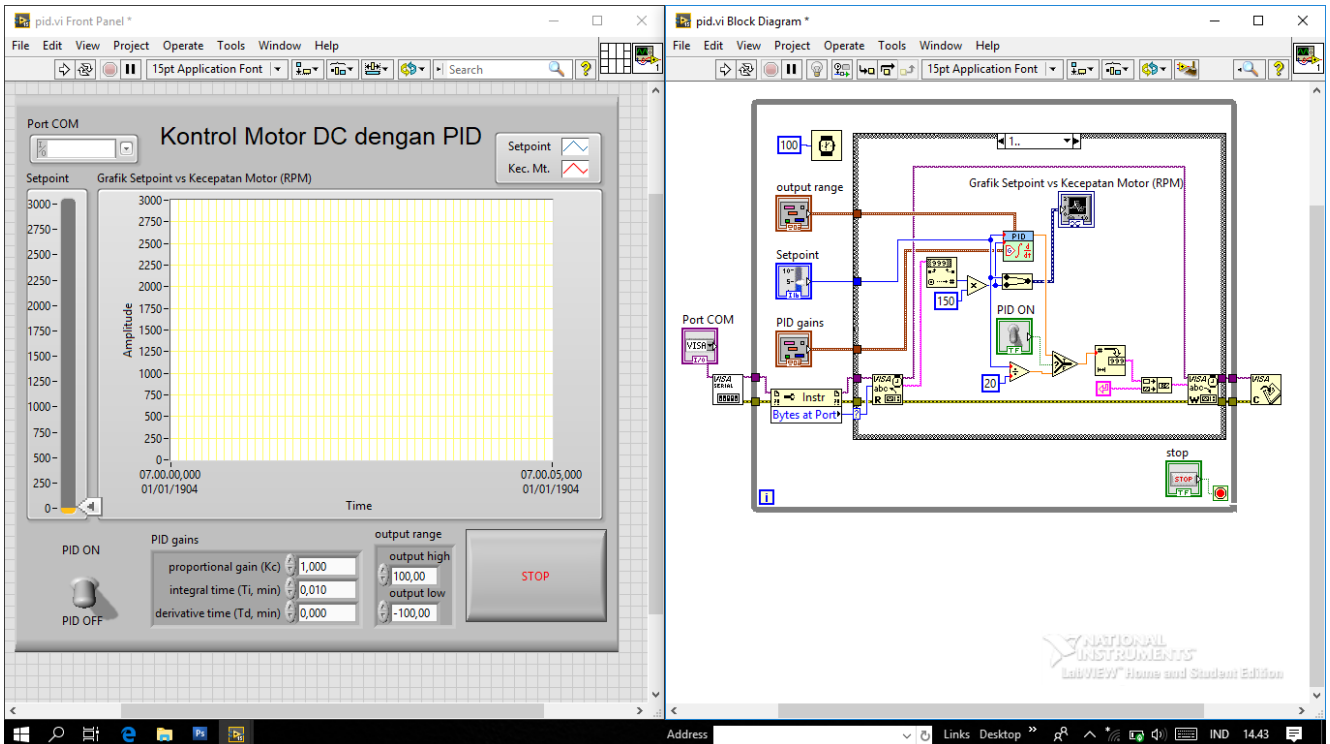


Gambar 4 Motor DC

Driver L293D digunakan sebagai pengontrol motor DC dengan inputan sinyal direction dan PWM dari Arduino Nano. Motor DC tidak dapat langsung oleh Arduino Nano karena kebutuhan daya listrik yang cukup besar pada motor DC sedangkan keluaran Arduino Nano sangat kecil maka dari itu dibutuhkan IC Driver Motor DC H-Bridge IC L293D dengan IC ini dapat digunakan sekaligus untuk mengontrol 2 unit motor DC secara PWM maupun TTI. Dengan IC L293D yang didesain khusus sebagai driver motor DC H-Bridge. Sinyal direction digunakan untuk mengontrol arah putaran motor DC sedangkan PWM digunakan untuk mengontrol kecepatan putar motor DC dengan level 0 - 255. Motor sebagai penggerak atau aktuator. Untuk lebih jelas IC L293D dapat dilihat pada **Gambar 5**.



Gambar 5 IC Driver Motor L293D

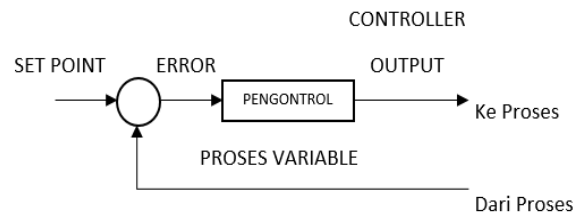


Gambar 6 Tampilan dan Source Code pada Labview

Sedangkan perangkat komputer berguna untuk menampilkan grafik dan inputan nilai KP, KD, KI yaitu nilai yang digunakan untuk mengatur putaran motor DC yang putarannya dideteksi oleh sensor optocoupler berputar dengan stabil. Lebih jelasnya tampilan pada perangkat komputer dapat dilihat pada Gambar 6.

B. Pengontrolan PID

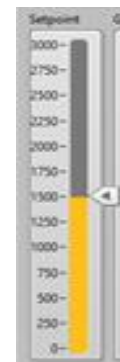
Sistem Kontrol PID merupakan kontroler untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut (*Feedback*). Sistem kontrol PID terdiri dari tiga buah cara pengaturan yaitu kontrol P (*Proportional*), D (*Derivative*) dan I (*Integral*), dengan masing-masing memiliki kelebihan dan kekurangan. Dalam implementasinya masing-masing cara dapat bekerja sendiri maupun gabungan diantaranya. Dalam perancangan sistem kontrol PID yang perlu dilakukan adalah mengatur parameter P, I atau D agar tanggapan sinyal keluaran sistem terhadap masukan tertentu sebagaimana yang diinginkan.[10] Secara umum hubungan antara harga yang diinginkan atau disebut setpoint, harga proses yang terukur (*Process Variable*), sinyal kesalahan (*Error*) dan keluaran pengontrol (*Controller Output*) tampak seperti Gambar 7.



Gambar 7 Blok diagram pengontrol

C. Sistem Pengontrol

Sistem pengontrol yang di terapkan disini yaitu menggunakan pengontrol PID yang di dalamnya terdapat 3 index nilai yang harus di ketahui dengan cara tuning parameter pengontrol PID. Salah satunya yaitu dengan metode osilasi. Pada Gambar 8 terdapat pengontrol setpoint yang berfungsi untuk mengatur kecepatan motor DC.



Gambar 8 Panel set point

D. Tuning Parameter Pengontrol PID

Agar diperoleh respon pengontrol yang terbaik, maka perlu dilakukan tuning parameter pengontrol PID. Dalam tuning, kita mencari nilai-nilai dari *Proportional Gain* (K_p), Waktu *Integral* (T_i) dan waktu *derivative* (T_d). Penalaan parameter *controller* PID selalu didasari atas tinjauan terhadap karakteristik yang diatur (*Plant*). Dengan demikian betapa rumitnya suatu *plant*, perilaku *plant* tersebut harus diketahui terlebih dahulu sebelum penalaan parameter PID itu dilakukan. Karena penyusunan model matematik *plant* tidak mudah, maka dikembangkan suatu metode eksperimental.

Metode ini didasarkan pada reaksi *plant* yang dikenai suatu perubahan. Dengan menggunakan metode itu model matematik perilaku *plant* tidak diperlukan lagi, karena dengan menggunakan data yang berupa kurva keluaran, penalaan kontroler PID telah dapat dilakukan. Penalaan bertujuan untuk mendapatkan kinerja sistem sesuai spesifikasi perancangan. Dua metode pendekatan eksperimen adalah Ziegler-Nichols.

E. Cara Kerja Sistem Kecepatan Motor DC Menggunakan Pengontrolan PID

Pembuatan simulasi kontrol kecepatan motor DC ini menggunakan PID berbasis LabView. Meliputi proses pemrograman yang berfungsi saat *setpoint* (titik acuan) di atur, maka akan mengontrol 1 buah motor DC agar dapat bekerja sesuai dengan kontrol PID yang diinginkan, dan menampilkan data yang berbentuk grafik melalui sebuah simulasi pada Labview.

Cara kerja sistem kecepatan motor DC ini disuplay dari IC driver motor yang telah terhubung dengan mikrokontroler arduino yang sudah terprogram. Pada sebuah motor DC terdapat 1 buah piringan yang memiliki 4 buah lubang yang berfungsi untuk pembacaan pada sebuah sensor optocoupler, yang dimana hasil pembacaan optocoupler akan di tampilkan pada sebuah program LabView dalam bentuk grafik.

F. Bagian – Bagian Dari *Plant* Simulasi

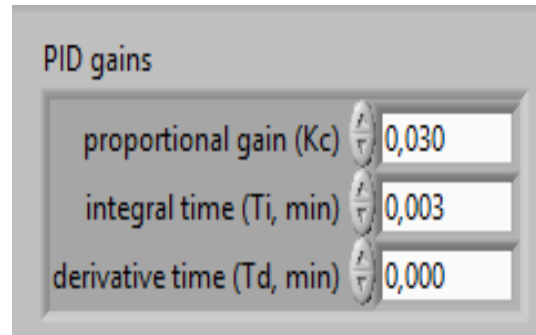
Ada beberapa bagian – bagian *plant* dan fungsi yang digunakan pada simulasi ini adalah sebagai berikut.

A. Panel *setpoint*

Panel *setpoint* digunakan untuk menentukan berapa *setpoint* yang di ingin kan dengan cara memindahkan jarum. Pada **Gambar 8** adalah panel *setpoint* yang digunakan memiliki rentang dari 0 s/d -3000.

B. Parameter Pengontrol PID

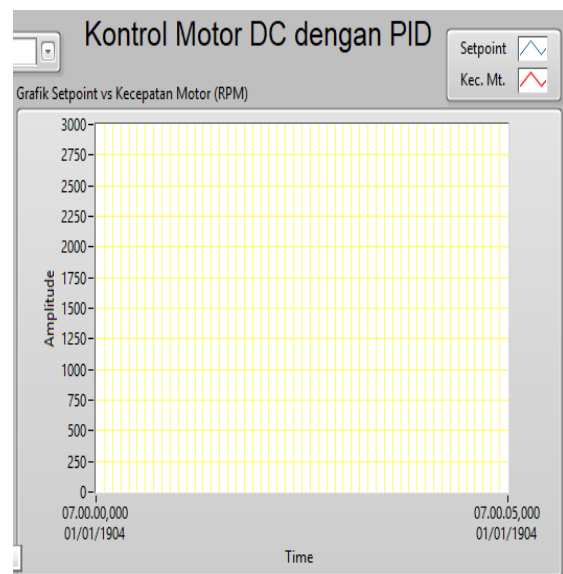
Parameter pengontrol PID digunakan untuk mengontrol nilai P , I , dan D dengan cara memasukan nilai yang sudah diketahui ke dalam bagian pengontrol ini. **Gambar 9** adalah bagian dari pengontrol PID yang sudah dimasukan dengan nilai $P = 0,030$, $I = 0,003$ dan $D = 0,000$.



Gambar 9 Parameter Pengontrol PID

C. Waveform chart

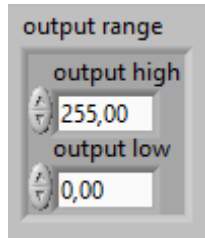
Waveform chart digunakan untuk mengetahui tampilan grafik performa sistem, pada **Gambar 10** adalah tampilan *waveform chart* yang digunakan dalam simulasi ini.



Gambar 10 Tampilan Waveform Chart

D. RPM Motor

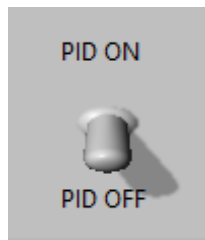
RPM motor digunakan untuk menentukan nilai output *high* dan output *low* pada motor. Pada **Gambar 11** adalah RPM motor yang digunakan dalam simulasi ini dengan RPM motor ini penulis dapat mengetahui keluaran motor.



Gambar 11 RPM Motor

E. Pengontrol PID dan Non PID

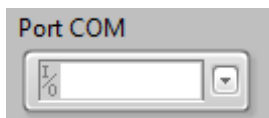
Pengontrol PID dan non PID digunakan untuk menentukan sistem yang akan digunakan pada sistem kecepatan motor DC. Lebih jelasnya dapat dilihat pada Gambar 12.



Gambar 12 Pengontrol PID dan Non PID

F. Koneksi Sistem

Koneksi sistem ini digunakan untuk menghubungkan hardware dengan simulasi pada program LabView. Lebih jelasnya dapat dilihat pada Gambar 13.



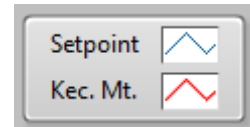
Gambar 13 Koneksi Sistem

G. Tombol Stop dan Indikator Grafik

Tombol stop ini digunakan untuk menghentikan jalannya program pada sebuah program LabView. Lebih jelasnya dapat dilihat pada Gambar 14. Indikator Grafik berfungsi untuk membedakan antara set point (hijau) dengan kecepatan motor (merah). Lebih jelasnya dapat dilihat pada Gambar 15.



Gambar 14 Tombol Stop



Gambar 15 Indikator Grafik

G. Tampilan Simulasi dan Perogram Pengaturan Kecepatan Motor DC

Pada Gambar 6 adalah tampilan utama simulasi dan program keseluruhan sistem dan Gambar 17 adalah bagian kendali PID.

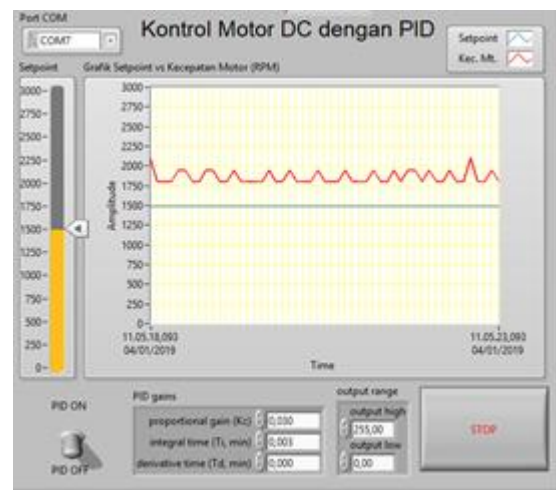
H. Cara Pengoperasian Sistem

Untuk melakukan pengoprasian sistem kecepatan motor DC ini adalah sebagai berikut:

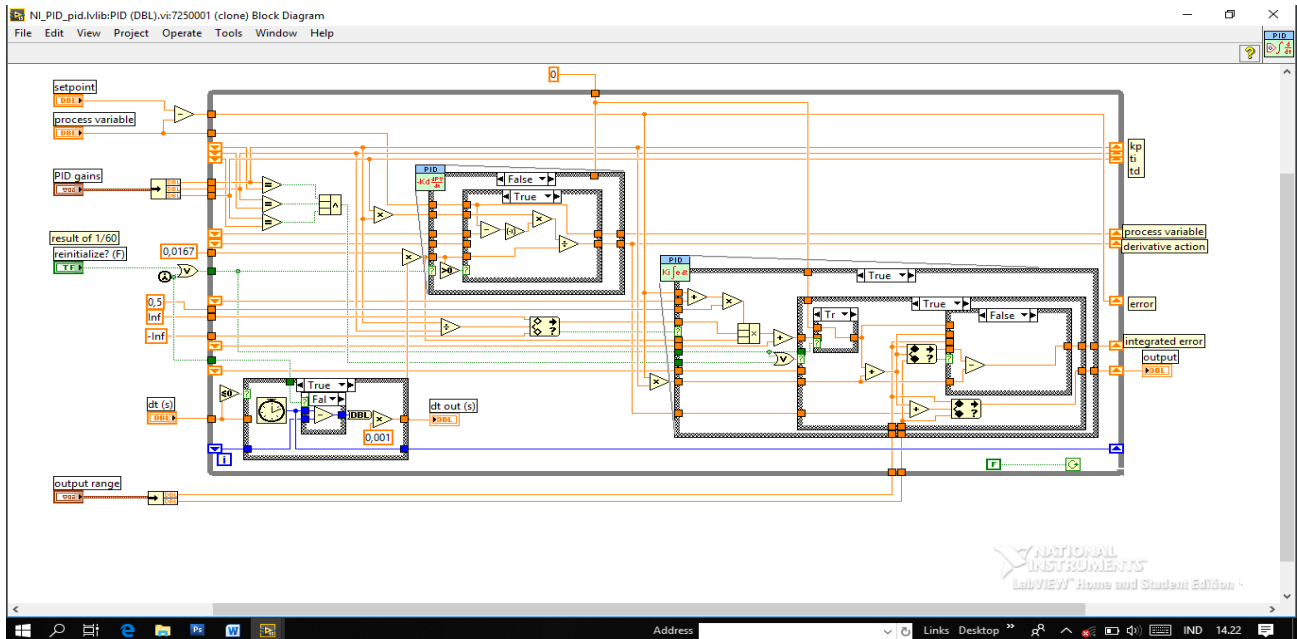
- Pertama download file sistem pada halaman ini.
- Pada file tersebut terdapat 1 folder yang didalamnya terdapat 2 file yaitu file pid.vi dan motor_dc.ino.
- Buka file yang bernama motor_dc.ino akan terlihat seperti pada Gambar 18.
- Buka file yang bernama pid.vi dapat dilihat seperti pada Gambar 6.
- Kemudian isi nilai $P = 0,030$ $I = 0,003$ dan $D = 0,000$.
- Jalankan program.
- Atur *setpoint* untuk mengatur kecepatan motor DC.
- Lihat grafik pada *waveform chart* akan tampil seperti Gambar 16.

Untuk melihat pengoprasian lebih jelas dapat di akses videonya melalui halaman berikut ini.

https://www.youtube.com/watch?v=Iur_epaqFCM



Gambar 16 Grafik waveform chart



Gambar 17 Kendali PID

```

motor_dc | Arduino 1.6.13
File Edit Sketch Tools Help

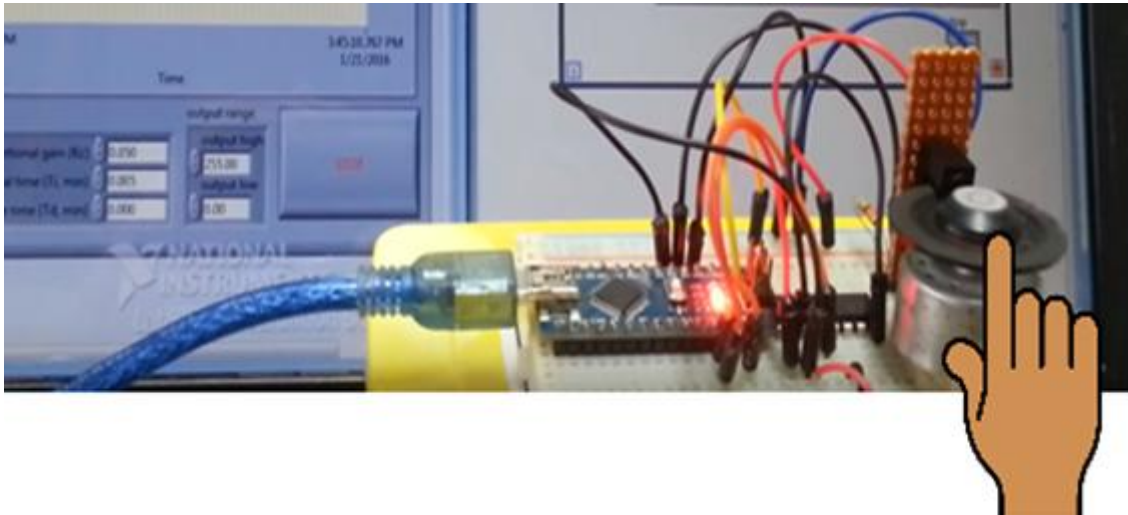
volatile int rpmcount = 0;
int rpm = 0;
unsigned long lastmillis = 0;

void setup() {
  pinMode(3, OUTPUT);
  Serial.begin(9600);
  attachInterrupt(0, rpm_fan, FALLING);
}

void loop(){
  if(Serial.available()){
    int a=Serial.parseInt();
    if(Serial.read()==char(13)) analogWrite(3,a);
  }
  if (millis() - lastmillis == 100){
    detachInterrupt(0);
    Serial.println(rpmcount);
    rpmcount = 0;
    lastmillis = millis();
    attachInterrupt(0, rpm_fan, FALLING);
  }
}

void rpm_fan(){
  rpmcount++;
}
    
```

Gambar 18 Source Code Arduino



Gambar 19 Gangguan yang dilakukan

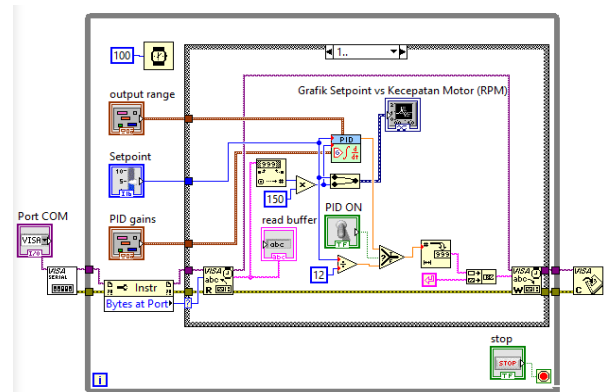
I. Pengujian Gangguan

Pada pengujian saat dilakukan gangguan pada motor DC siap digunakan dan telah diberi supply 5V dan motorpun berputar dengan piringan yang sudah diberi 4 lubang untuk menghasilkan grafik yang dibaca oleh optocoupler pada saat piringan yang terdapat pada motor DC berputar selanjutnya kita lakukan gangguan berupa menekan piringan pada motor DC menggunakan salah satu jari , gangguan dilakukan seperti pada Gambar 19.

Berikut listing program pada software labview dapat dilihat pada Gambar 20.

III. HASIL DAN PEMBAHASAN

Dalam merancang simulasi yang paling penting adalah pengujian dari simulasi tersebut, apakah simulasi memenuhi target yang di tentukan atau tidak. Dengan dicapainya tujuan yang telah ditargetkan maka simulasi tersebut dikatakan baik. Pada bab ini penulis akan memaparkan berupa hasil ujicoba dari simulasi pada saat pengaturan kecepatan motor DC dengan menggunakan PID pada program LabView yang telah dilakukan.



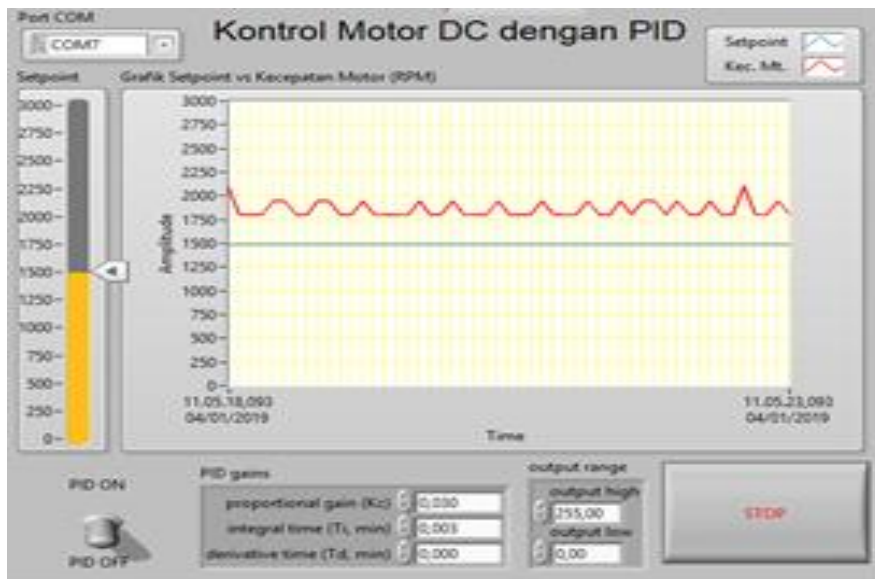
Gambar 20 Listing program pada software labview

A. Kontrol PID (Proporsional, Integral, dan Derivatif)

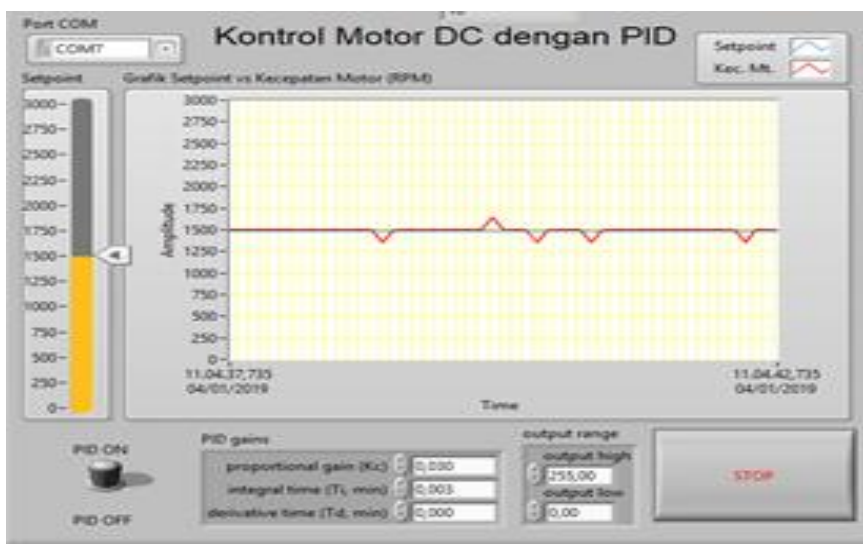
PID ini merupakan kontroler untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Komponen pada PID terdiri dari tiga jenis yaitu Proporsional (Kp) , Integral (Ki), dan Derivatif (Kd). Ketiganya bisa dipakai bersamaan ataupun sendiri-sendiri, tergantung dari respon yang kita inginkan dari suatu plant.

B. Pengujian tanpa PID dan menggunakan PID terhadap Motor DC dengan Feedback Encoder

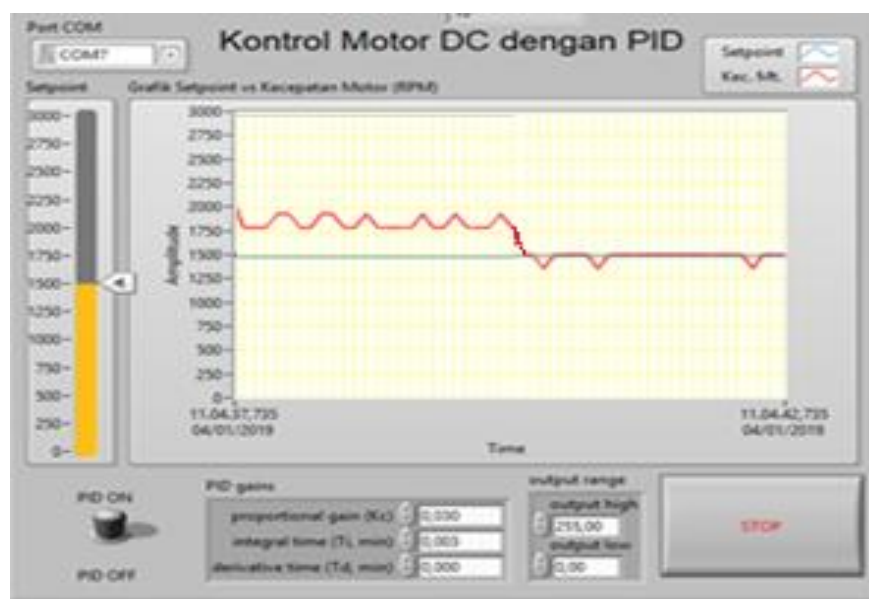
Pengaturan speed / kecepatan motor adalah dengan menerapkan metode pengaturan kecepatan menggunakan PWM (Pulse Width Modulation), yaitu metode dengan merubah nilai duty cycle pada driver motor , dengan begitu tegangan yang dialirkan pada motor dapat diatur , PWM yang digunakan pada penelitian ini menggunakan fitur timer 0 pada mikrokontroller dengan resolusi 8 bit (0 s/d 255) , berikut perhitungan pwm yang digunakan : Pengujian yang dilakukan menerapkan control Proporsional, Integral, dan Derivatif (PID) untuk memperbaiki nilai perbaikan kecepatan terhadap nilai error system yg didapat dari perbedaan nilai set point terhadap present value rpm motor.



Gambar 21 Tampilan pengontrolan Motor DC tanpa PID



Gambar 22 Tampilan pengontrolan Motor DC dengan PID



Gambar 23 Tampilan ketika diberi gangguan

Dari **Gambar 21** diatas terlihat kecepatan putaran pada motor DC yang di tampilkan pada program LabView, kecepatan putaran motor tidak sesuai dengan nilai set point yang telah ditentukan.

Sedangkan pada **Gambar 22** diatas merupakan tampilan pengontrolan motor DC setelah menggunakan metode PID, terlihat lebih kecepatan putaran motor DC mencapai nilai set point yang telah ditentukan dan terlihat lebih stabil.

C. Pengujian ketika diberi gangguan

Pengujian pengontrol PID dengan memberikan sebuah gangguan dilakukan guna untuk mengetahui hasil data grafik dari performansi sistem PID pada simulasi pengujian pengontrol PID berbasis Labview dengan menggunakan set point yang berbeda agar hasil dari pengujian pengontrol PID tanpa gangguan dan pengujian pengontrol PID dengan memberikan sebuah gangguan dapat dibedakan dengan menggunakan sebuah grafik.

Ketika rangkaian yang sudah menggunakan PID ketika diberi gangguan pada baling-baling yang terdapat pada motor DC grafik yang dihasilkan akan jauh dan nilai set point untuk seketika sebelum akhirnya kembali lagi pada nilai set pointnya lagi atau stabil kembali. Lebih jelasnya dapat di lihat pada **Gambar 23**.

D. Penentuan Parameter PID

Langkah awal untuk pengujian pada simulasi ini yaitu dilakukan untuk mendapatkan nilai parameter Kp, Ki, dan Kd. Berdasarkan teori yang digunakan untuk mendapatkan nilai parameter tersebut dengan mencari nilai Kcr dan Pcr untuk mendapatkan nilai parameter Kp, Ki, dan Kd dengan menggunakan metode osilasi dan didapatkan hasil dari penentuan parameter PID, sebagai berikut:

- a. Atur set point sesuai dengan yang diinginkan. Dalam percobaan di atur nilai set point 1500.
- b. Lalu set nilai Kp 0,001 lalu jalankan program, naikan sedikit demi sedikit nilai Kp-nya hingga tercapai osilasi pada grafik, nilai Kp terakhir sama dengan nilai Kcr.
- c. Hasil yang di dapatkan dari tuning yaitu: $K_p = 0,030$ $K_i = 0,003$, dan $K_d = 0,000$
- d. Lalu setting aknga keluaran motor dengan hasil yang didapatkan output *high* 255 dan output *low* 0
- e. Perhitungan untuk menentukan PID:

$$P = \frac{Kcr}{1.7} \dots\dots\dots(1)$$

$$P = \frac{0.05}{1.7} = 0,030$$

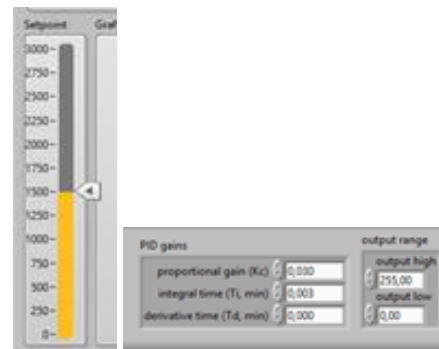
$$I = \frac{Pcr}{2} \dots\dots\dots(2)$$

$$I = \frac{0,006}{2} = 0,003$$

$$D = \frac{0}{8} \dots\dots\dots(3)$$

$$D = \frac{Pcr}{8} = 0,000$$

Setelah didapatkan nilai PID, nilai-nilai tersebut akan dimasukan ke dalam nilai pengontrol setiap parameter pada program yang telah dibuat pada **Gambar 24**.



Gambar 24 (a) Nilai Set point (b) Nilai Kp, Kd, Ki

IV. KESIMPULAN

Setelah dilakukan pengujian pada sistem yang dirancang, dapat ditarik beberapa kesimpulan. Parameter KP (Konstanta Proporsional) berperan dalam mempercepat waktu untuk kecepatan motor meraih set poin yang diinginkan, Parameter KI (Konstanta Integral) digunakan untuk menghilangkan error sehingga sistem menjadi stabil, Parameter KD (Konstanta Derivatif) berperan mengurangi overshoot (melampaui).

Pada pengujian, sistem dapat merespon perubahan arah *setpoint* dengan tingkat keberhasilan 100%. Dengan menggunakan metode PID ini dapat putaran motor DC terlihat lebih stabil dan mencapai nilai set point, yaitu dengan nilai kontrol yang diberikan yaitu $K_p = 0,030$, $K_i = 0,003$, dan $K_d = 0$ didapati kecepatan motor DC yang mulai stabil dengan set point sistem pengendali PID mampu meredam error dan menghasilkan *grafik*.

Adapula saran untuk pengembangan alat ini mengenai mekanik pada baling-baling yang terdapat pada motor DC, lubang yang terdapat pada baling-baling tersebut harus memiliki lubang yang sama karena jika tidak akan berpengaruh terhadap bacaan optocoupler.

DAFTAR PUSTAKA

- [1] Waluyo, Aditya F, dan Syahrial. 2012. Analisis Penalaan Kontrol PID pada Simulasi Kendali Kecepatan Putaran Motor DC Berbeban menggunakan Metode Heuristik. *Jurnal Teknik Elektro Itenas*, Vol. 1, No. 2.
- [2] Aria, M. "PID control of a three-degrees-of-freedom model helicopter." *Majalah Ilmiah UNIKOM* 9.2 (2011): 207-214.
- [3] Afyudin, Rizal, Junartha Halomoan, and Erwin Susanto. "Perancangan Dan Implementasi Scada Pada Mini Plan Sorting Kayu Menggunakan Software Labview." *eProceedings of Engineering* 2.1 (2015).
- [4] Hidayati, Qory. "Pengaturan Kecepatan Motor DC dengan Menggunakan Mikrokontroler Atmega 8535." *JIP (Jurnal Ilmiah Politeknik)* 4.1 (2012).
- [5] Djalal, Muhammad Ruswandi, et al. "Desain optimal kontroler PID motor DC menggunakan cuckoo search algorithm." *SENTIA* 2015 7.1 (2015).
- [6] Prabowo, Brilliant Adhi. "Pemodelan Sistem Kontrol Motor DC dengan Temperatur Udara sebagai Pemicu." *INKOM Journal* 2.1 (2010): 39-43.
- [7] Putra, Dimas Harrind Yudha, and Riswan Dinzi. "Studi Pengaturan Kecepatan Motor DC Shunt dengan Metode Ward Leonard." (2014).
- [8] Nugroho, Yudistiro Ardi. Penerapan Sensor Optocoupler Pada Pengukur Kecepatan Angin Berbasis Mikrokontroler AVR Atmega8535. Diss. Universitas Negeri Semarang, 2011.
- [9] Ali, Muhamad. "Pembelajaran Perancangan Sistem Kontrol PID Dengan Software Matlab." *Jurnal Edukasi@ Elektro* 1.1 (2004): 1-8.
- [10] Wardhana, Danu Wisnu, Arif Wahyudi, and Hendro Nurhadi. "Perancangan Sistem Kontrol PID Untuk Pengendali Sumbu Azimuth Turret Pada Turret-gun Kaliber 20mm." *Jurnal Teknik ITS* 5.2 (2016): A512-A516.
- [11] Utama, Jana. "Electrocardiogram (ECG) dengan Noise Reduction Berbasis Wavelet Menggunakan Pemrograman LabVIEW." *Jurnal Telekomtran* 1, no. 1 (2013).