

TRAINER PRAKTIKUM PENGENDALI KECEPATAN MOTOR DC DENGAN PID BERBASIS ARDUINO

Fedrik Fajar Alanro; Heru Supriyono
Teknik Elektro, Teknik, Universitas Muhammadiyah Surakarta

Abstrak

Motor dc merupakan suatu komponen yang berguna mengubah energi listrik menjadi energi kinetik. Kecepatan suatu motor saat berputar biasanya terjadi penurunan sehingga menghasilkan nilai kecepatan yang tidak *konstan*. Didalam praktikum terutama DTK (Dasar Teknik Kendali) belum terdapat trainer yang membantu dalam menyelesaikan masalah pengendalian motor DC. Dengan begitu pada penelitian ini memiliki tujuan untuk membuat trainer agar mahasiswa mudah dalam mempelajari tentang pengendalian motor dc. Dalam hal ini proses pengendalian motor dc menggunakan metode PID (*Proporsional Integral Derivative*) dengan mengatur nilai dari k_p , k_i , dan k_d secara acak, lalu untuk set poin menyesuaikan dari kecepatan yang di inginkan. Komponen yang dibutuhkan untuk trainer ini terdiri dari *Arduino UNO*, *keypad*, *LCD*, *Driver L298N*, Motor dc, dan Jumper. Sedangkan aplikasi yang diperlukan *Arduino* dan *Matlab*. Didalam penelitian ini didapatkan hasil yang bagus dengan nilai $k_p=0,04$, nilai $k_i=0,05$, dan nilai $k_d=0,004$ memiliki nilai *overshoot* yang rendah yaitu 0,73 dan *settling time* bernilai 10,66. Dengan nilai *overshoot* yang mendekati 0 dan nilai *settling time* yang rendah membuktikan bahwa motor bergerak dengan baik dan stabil.

Kata Kunci: motor dc, kecepatan, arduino, nilai, matlab.

Abstract

DC motor is a useful component to convert electrical energy into kinetic energy. The speed of a motor when rotating usually decreases resulting in a non-constant speed value. In practicum, especially DTK (Basic Control Techniques) there is no trainer who helps in solving DC motor control problems. That way this research has the goal of making trainers so that students can easily learn about controlling dc motors. In this case the process of controlling the dc motor uses the PID (Proportional Integral Derivative) method by adjusting the values of k_p , k_i , and k_d randomly, then for the set points adjusting the desired speed. The components needed for this trainer consist of *Arduino UNO*, *keypad*, *LCD*, *L298N driver*, dc motor, and jumper. While the application needed *Arduino* and *Matlab*. In this study, good results were obtained with a value of $k_p = 0.04$, a value of $k_i = 0.05$, and a value of $k_d = 0.004$ which has a low overshoot value of 0.73 and a settling time value of 10.66. With an overshoot value close to 0 and a low settling time value, it proves that the motor moves well and is stable.

Keywords: dc motor, speed, arduino, value, matlab.

1. PENDAHULUAN

Motor DC merupakan merupakan suatu komponen mesin yang mengubah energi listrik menjadi energi kinetik. Penggunaan motor DC dalam dunia industri maupun local sangat umum digunakan untuk piranti pendukung dalam kelancaran alat agar berjalan dengan normal dan sesuai dengan kebutuhan yang di inginkan. Sistem pengontrolan kecepatan motor DC sekarang ini banyak dilakukan. Karena dalam pemakaian motor DC sangat dibutuhkan diatur kecepatannya. Motor DC itu sendiri memiliki beberapa metode untuk mengatur kecepatannya. Beberapa penelitian telah dilakukan dalam hal mengendalikan motor DC itu sendiri. Dari metode Fuzzy logic, PID dan PWM. Dalam pengoperasian motor DC tersebut terdapat acuan yang meliputi kecepatan motor, tegangan keluaran motor dan arus keluaran motor. Beberapa acuan tersebut memiliki nilai yang sangat penting dalam proses pengontrolan motor DC tersebut. Dengan mengukur nilai tersebut pengguna bisa mengetahui apa yang dibutuhkan dalam penggunaan motor DC tersebut (Dio & Aswardi, 2020).

PID (*Proporsional Integral Derivative*) yaitu kontroler untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem tersebut. Sistem kendali ini memiliki 3 aksi yaitu *Proporsional*, *Integral*, *Derivative*. Masing-masing aksi kontrol ini memiliki keunggulan tertentu. Tindakan kontrol proporsional memiliki keunggulan dari risetime yang cepat, tindakan kontrol integral memiliki keunggulan untuk mengurangi kesalahan, dan tindakan kontrol diferensial memiliki keunggulan untuk mengurangi *overshoot* / *undershoot*. Oleh karena itu, untuk menghasilkan keluaran output dengan risetime yang tinggi dan error yang kecil kita dapat menggabungkan ketiga aksi kontrol ini menjadi aksi control PID.

Lab Teknik Elektro UMS merupakan suatu ruangan untuk membantu dalam mempelajari berbagai kegiatan praktikum Teknik Elektro. Terutama pada praktikum DTK (Dasar Teknik Kendali) mempelajari tentang pengendalian kecepatan atau sudut suatu mesin listrik. Namun, memiliki kendala tidak memiliki trainer yang menunjang jalannya praktikum sehingga praktikan tidak dapat mengetahui secara langsung kerja alat yang di teliti hanya sekedar melihat simulasi dari laptop/komputer.

Berdasarkan permasalahan di atas penulis membuat alat Pengendali kecepatan motor DC dengan PID berbasis Arduino. Penelitian ini diharapkan dapat mengendalikan kecepatan motor DC agar sesuai dengan nilai yang di inginkan, terutama untuk media praktikum dengan bantuan software tambahan yaitu matlab maka bisa mengetahui nilai dari *rise time*, *settling time*, dan juga *overshoot*. Mempermudah praktikan dalam menemukan solusi pada motor DC.

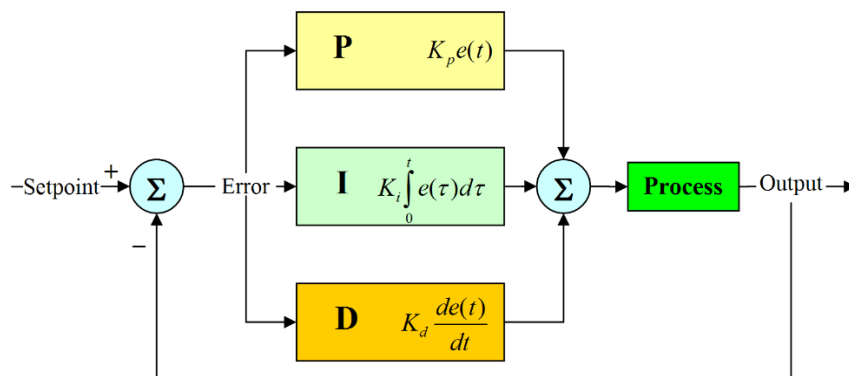
2. METODE

Pada penelitian ini, ada beberapa metode yang digunakan. Pertama melakukan literatur teori yang berkaitan dengan trainer praktikum pengendali kecepatan motor DC dengan pid berbasis arduino yang dilakukan dengan mencari dari jurnal dan situs web yang bersangkutan dengan hal yang dibahas. Kedua melakukan bimbingan pada dosen pembimbing terkait penelitian yang akan di lakukan. Ketiga dengan melakukan perancangan alat yang akan diteliti.

2.1. Alat dan Bahan

2.1.1 PID (*Proporsional Integral Derivative*)

PID adalah kontroller untuk mengendalikan kecepatan atau sudut dengan adanya karakteristik umpan balik pada sistem yang akan di kendalikan. Sistem kerjanya pengontrol PID akan memberikan aksi pada inputan berupa k_p , k_i , k_d , dan set poin sesuai nilai yang dimasukkan untuk menggerakkan motor agar bergerak sesuai dengan set poin. K_p adalah *konstanta Proporsional* yang berfungsi sebagai penguat saat motor ingin berputar, efek pada K_p yang tidak sesuai yaitu menimbulkan *overshoot*. K_i adalah *konstanta Integral* berfungsi mengontrol osilasi atau respon *steady-state* saat motor bergerak. K_d adalah *konstanta Derivative* berfungsi sebagai penahan agar motor bergerak tetap pada set poin. Set poin adalah output kecepatan yang ingin dicapai. Rumus PID dapat dilihat pada gambar 1.



Gambar 1 Rumus PID

Rumus yang digunakan sesuai dengan rumus PID pada umumnya. Fungsi dari P sebagai penguat saat motor ingin berputar. Fungsi I sebagai memperbaiki osilasi agar gelombang yang dihasilkan tidak terjadi osilasi yang terlalu besar. Fungsi D sebagai mengurangi hasil *overshoot* yang terjadi saat motor sedang bergerak agar tetap berada di set poin. Rumus yang sudah didapatkan lalu akan diubah menjadi script agar dapat dibaca oleh arduino. Pada rumus tersebut dapat dijabarkan bahwa sistem umpan balik terletak pada hasil outputnya. Umpan balik yaitu sistem yang dilakukan untuk mempertahankan keadaan tertentu dari sistem yang dikendalikan. Artinya sistem umpan balik ini bekerja untuk mempertahankan kecepatan pada motor agar tetap

atau bertahan sesuai dengan set poin yang diinputkan. Dengan adanya karakteristik umpan balik ini motor dapat bergerak dengan baik dan stabil.

2.1.2 Arduino Uno

Gambar 2 Arduino UNO. Arduino UNO adalah papan mikrokontroler yang berbasis Atmega 328. Arduino di masa sekarang sangat dibutuhkan karena merupakan alat yang sederhana tapi memiliki fungsi yang luas, seperti mengembangkan pengatur suhu, pengendalian peralatan pintar, dan juga sebagai pengendali motor DC. Arduino UNO memiliki 14 pin yang terdapat 6 pin sebagai output PWM. Pin PWM terdiri dari pin 3,5,6,9,10, dan 14. Pin PWM ini berfungsi sebagai merubah tegangan yang masuk menjadi pulsa atau gelombang. Selain pin PWM terdapat juga 6 pin inputan analog. Pada pin 2 dan 3 merupakan pin interupsi eksternal, pin ini digunakan untuk mengintruksi agar output menghasilkan sebuah nilai.



Gambar 2 Arduino UNO

2.1.3 Motor dc

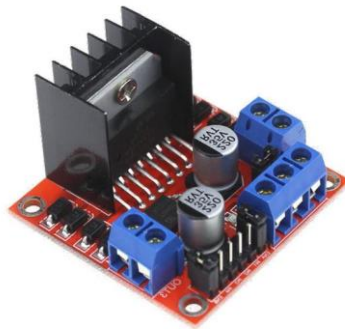
Gambar 3 Motor DC. Motor DC adalah sebuah motor yang memerlukan arus searah pada kumparan untuk dirubah menjadi energi kinetik. Kumparan yang ada pada motor DC ini berfungsi sebagai penghasil putaran. Hasil putaran pada motor dc disebut dengan RPM (*Revolution Per Minute*). Motor DC memiliki torsi yang besar dan dapat dikendalikan. Pada kehidupan sehari-hari motor DC digunakan pada Bor listrik, vibrator HP, dan juga kipas angin. Untuk sebuah motor DC yang digunakan pada penelitian ini memiliki kecepatan maximum sebesar 1700 RPM dengan tegangan 12 volt dan ratio 1 : 4,4. Panjang motor 31 mm, panjang Gear 17 mm, diameter Terluar 25 mm, dan berat 100 gram.



Gambar 3 Motor DC

2.1.4 Driver L298N

Gambar 4 Driver L298N. Driver L298N adalah modul untuk mengontrol kecepatan dan sudut motor DC. Driver ini sering digunakan dan juga dihubungkan dengan arduino karena merupakan *driver* yang terbaik untuk mengendalikan motor DC. Konstruksi yang digunakan yaitu *H-Bridge* digunakan sebagai mengendalikan beban induktif pada kumparan yang ada pada motor listrik. Untuk pin yang ada pada driver ini terdiri dari pin In, pin Out, pin *Enable*, pin 12v, pin 5v, pin *ground*, dan pin *jumper*. Pin enable digunakan untuk meningkatkan kecepatan motor DC saat sedang berputar. Pin *jumper* digunakan sebagai pin tambahan jika motor DC kekurangan kecepatan.



Gambar 4 Driver L298N

2.1.5 Keypad 4x4

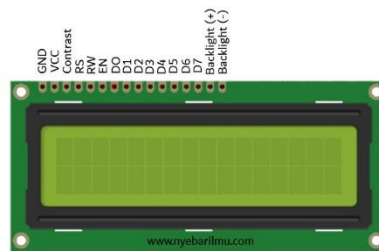
Gambar 5 Keypad. Keypad adalah perangkat listrik yang digunakan untuk memasukan sebuah inputan yang harus di lakukan dengan interaksi manusia. Keypad biasanya berisi angka ataupun huruf yang harus ditekan atau interaksi dengan manusia. HMI (*Human Machine Interface*) penyebutan keypad yang yang memiliki fungsi sebagai interface perangkat listrik dengan manusia. Keypad 4x4 memiliki konstruksi yang simpel dan hemat penggunaan pada mikrokontroler. Disebut hemat karena memiliki jumlah key atau tombol yang sedikit.



Gambar 5 Keypad 4x4

2.1.6 LCD 12 x 6

Gambar 6 LCD 16x2. *Liquid Crystal Display* (LCD) adalah perangkat yang digunakan manusia sebagai media reflektif. LCD biasanya ditemukan pada TV, Monitor, laptop, dan juga HP. LCD merupakan komponen penting pada sebuah perangkat listrik karena dengan adanya LCD maka manusia lebih mudah untuk melihat hasil atau output dengan jelas. LCD memiliki dua jenis, yaitu LCD berwarna dan LCD tidak berwarna. LCD tidak berwarna memiliki piksel yang sedikit, oleh karena itu piksel yang dihasilkan LCD untuk yang berwarna memiliki ribuan piksel. LCD merupakan teknologi digital yang memiliki struktur molekul polar dan bias membentuk gambar dengan menyinari *Liquid Crystal* dan filter warna yang berada di antara dua elektroda transparan.



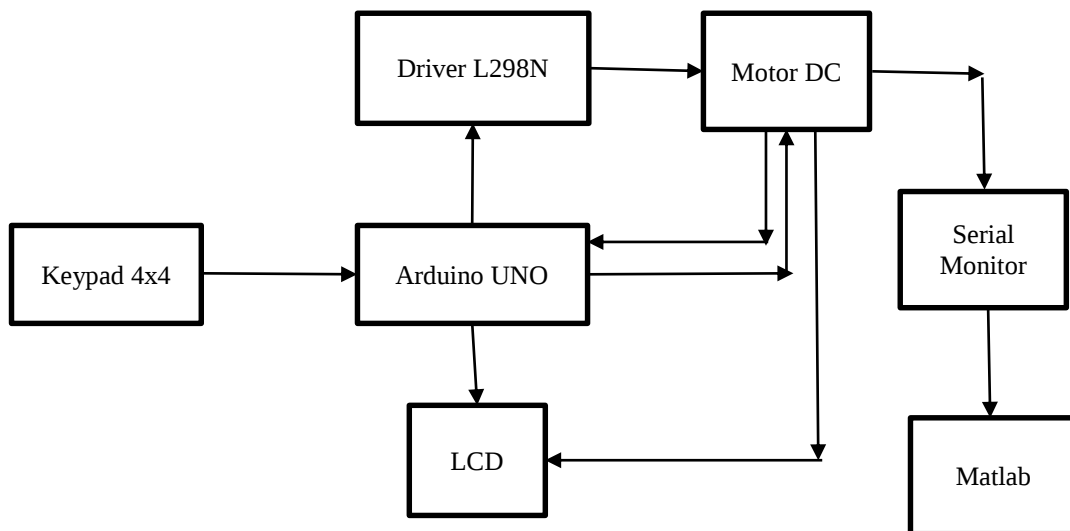
Gambar 6 LCD 16x2

2.1.7 Matlab

Matlab merupakan sebuah *software* yang biasanya banyak digunakan pada bidang teknik. Karena matlab merupakan aplikasi yang bisa memecahkan banyak masalah Algoritma, analisis data, simulasi, visualisasi dan juga bisa membuat sebuah hasil akhir atau kesimpulan. Pada penelitian ini matlab digunakan untuk memecahkan masalah hasil akhir yang digunakan untuk data akhir atau hasil akhir. data yang akan dicari seperti *overshoot*, *settlingTime*, dan *RiseTime*.

2.2. Blok Diagram

Perancangan trainer praktikum pengendali kecepatan motor DC dengan PID berbasis *arduino* dibuat dengan blok diagram agar dapat mempermudah penulis dalam melakukan perancangan alat. Blok diagram dapat dilihat pada gambar 7.

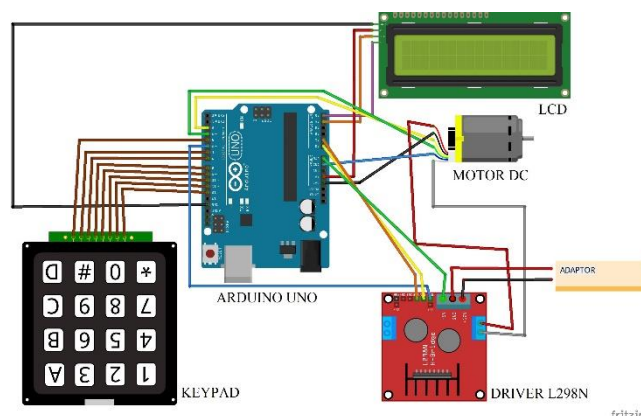


Gambar 7 Blok Diagram

Keypad sebagai inputan untuk memasukkan nilai k_p , k_i , k_d , dan set poin. Inputan akan diproses kedalam arduino agar *arduino* dapat menampilkan inputan kedalam *LCD* dan ketika inputan sudah dimasukan semuanya proses selanjutnya *arduino* akan mengontrol driver dan motor DC, ketika motor DC berputar nilai kecepatan akan ditampilkan ke dalam *LCD*. Pada *arduino* dan motor DC terjadi umpan balik yang digunakan agar kecepatan motor tetap stabil sesuai set rpm yang dimasukkan. Selain *LCD*, kecepatan motor dc dapat dilihat pada serial monitor yang nanti datanya akan digunakan untuk dimasukkan ke dalam matlab.

2.3. Konfigurasi Rangkaian

Konfigurasi digambarkan melalui aplikasi Fritzing. Komponen yang digunakan yaitu *Arduino UNO*, Motor DC, *Driver L298N*, *LCD*, *Keypad*. Berikut konfigurasi rangkaian dapat dilihat pada gambar 8.



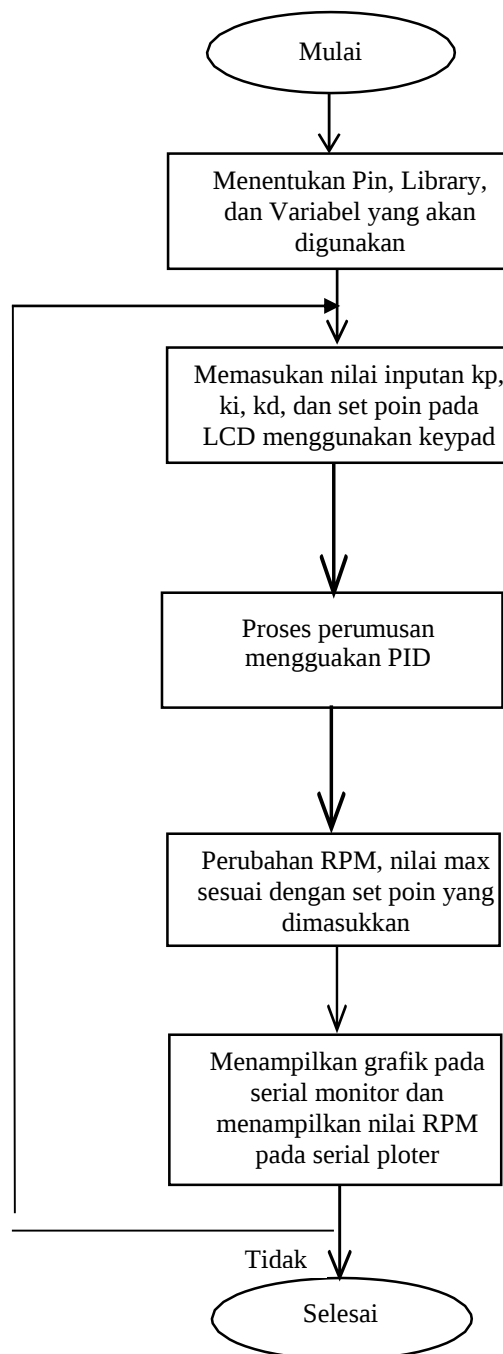
Gambar 8 Konfigurasi Rangkaian

Dalam konfigurasi ini, keypad berada pada pin 4,6,7,8,9,10,11.dan 12. Tegangan yang digunakan 5 v yang terhubung dengan VCC LCD dan VCC motor DC, lalu ground arduino tersambung dengan ground motor DC, ground LCD,dan ground Driver. Pada LCD terdapat SCL

dan SDA yang tersambung pada pin A5 dan A4. Pada driver IN 1 dan IN 2 tersambung pada pin A0 dan A1, lalu bagian Enabler tersambung pada pin 5. Bagian VCC driver tersambung ke dalam pin Vin arduino. Pada Motor DC pada bagian encoder A dan encoder B tersambung pada pin 2 dan pin 3. Pada bagian out 1 dan out 2 motor DC terhubung pada output *driver L298N*.

2.4. Diagram Alir Program

Perancangan script program yang digunakan untuk menjalankan sistemika alat Trainer Praktikum Pengendali kecepatan Motor DC dengan PID berbasis Arduino agar mempermudah penulis dalam menyusunnya. Kemudian diagram alir ini digunakan sebagai pedoman penulis dalam pembuatan program. Diagram alir dapat dilihat pada Gambar 9.

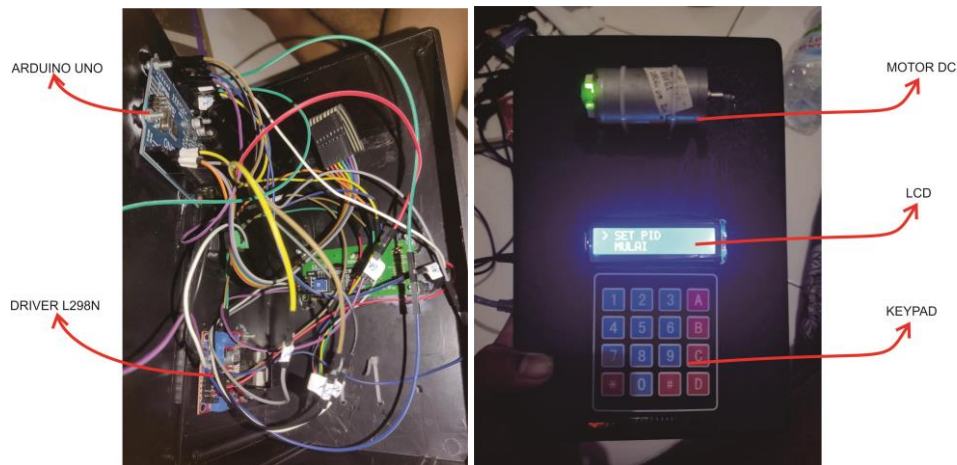


Gambar 9 Diagram alir program

Pada bagian awal program menentukan Pin arduino yang akan digunakan, lalu menentukan library dan juga variable. Setelah itu, pada looping berisikan inputan yang akan dimasukkan dengan menggunakan *keypad* yang dilihat melalui *LCD*. Inputan berisikan nilai *kp*, nilai *kd*, nilai *ki*, dan set poin. Dilanjut dengan membuat program rumus pid yang digunakan untuk menghitung agar nilai kecepatan motor dc dapat bergerak dan stabil. Setelah semua sudah sesuai, grafik dan nilai rpm yang dihasilkan dapat dilihat pada *serial monitor* dan *serial plotter*.

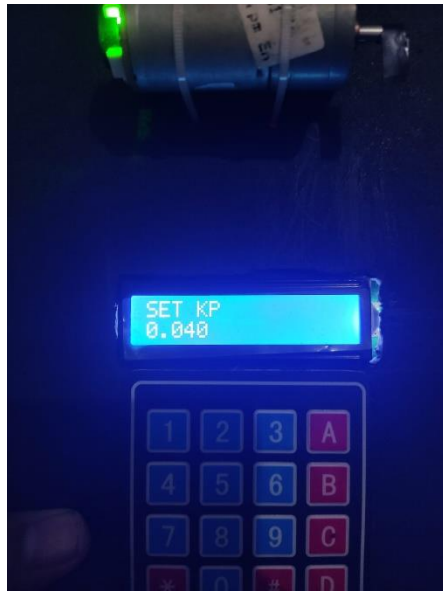
3. HASIL DAN PEMBAHASAN

Penelitian ini menghasilkan trainer praktikum yang dapat digunakan untuk mempermudah praktikan dalam melakukan praktikum. Alat yang digunakan berupa *Arduino UNO*, *keypad*, *LCD*, *Driver L298N*, dan Motor DC. Alat ini digunakan untuk mengendalikan kecepatan motor agar lebih stabil dan bias di sesuaikan dengan yang diinginkan. Motor yang stabil membuktikan bahwa motor bergerak dengan baik sehingga didapatkan rpm yang bagus. Dapat dilihat pada gambar 10.

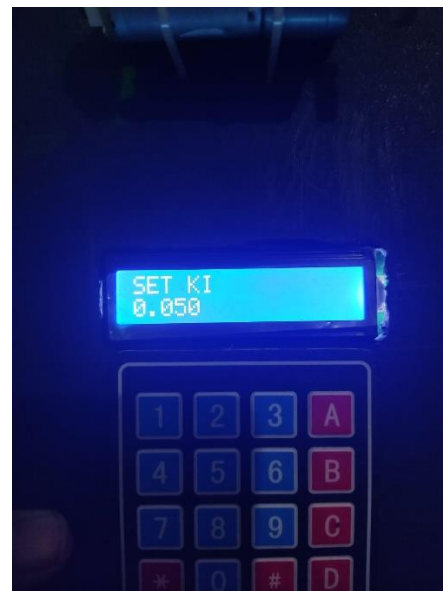


Gambar 10 Trainer Motor DC

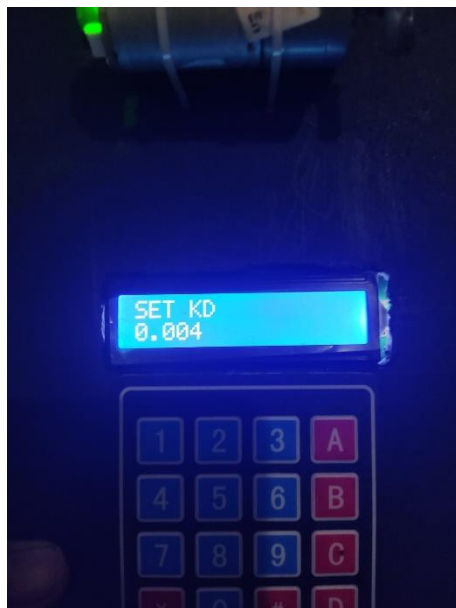
Sistem kerja alat ini untuk mendapatkan hasil output akhir yaitu diawali dengan memasukkan nilai k_p , k_i , k_d , dan set poin yang dilihat pada *LCD* dengan menggunakan keypad, setelah itu jalankan. Nilai rpm yang dihasilkan dapat dilihat pada serial monitor yang ada pada aplikasi arduino. Data kecepatan yang ada pada serial monitor dicopy lalu masuk pada *Microsoft Excel* untuk dipaste. Pada *Microsoft Excel* harus dirapikan secara manual agar data dapat dibaca dengan benar, data yang harus ada ada 3 kolom yaitu kolom 1 berisi penomoran, kolom 2 berisi nilai set poin (contoh : 1500), kolom 3 berisi nilai kecepatan motor. Jumlah data yang diambil bebas. Setelah data yang didapatkan tadi lalu dimasukkan pada aplikasi *Matlab* dengan cara *copy-paste* lagi data pada *excel* ke bagian *Workspace* aplikasi *Matlab*. Pada aplikasi *Matlab* ini juga terdapat script yang digunakan untuk memunculkan data yang dicari dan juga grafik. Setelah semua sudah ada pada *Matlab* lalu klik *run*, maka data hasil akhir akan muncul berupa nilai *Overshoot*, *SettlingTime*, *RiseTime*, dan juga grafik.



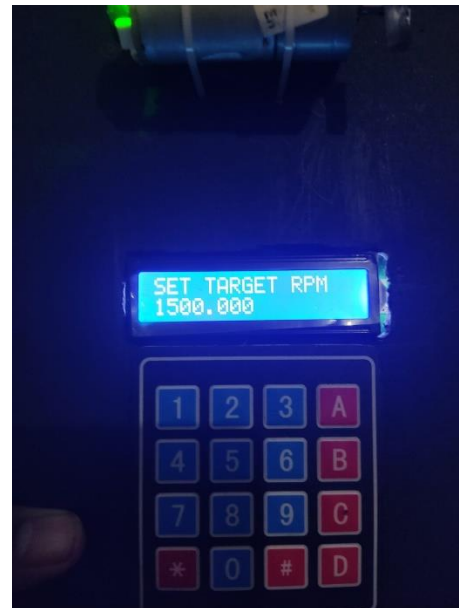
Gambar 11 Set KP



Gambar 12 Set KI

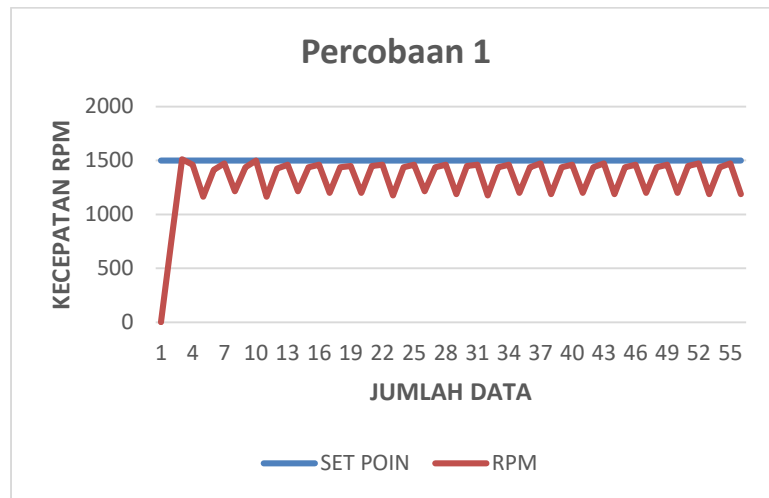


Gambar 13 Set KD



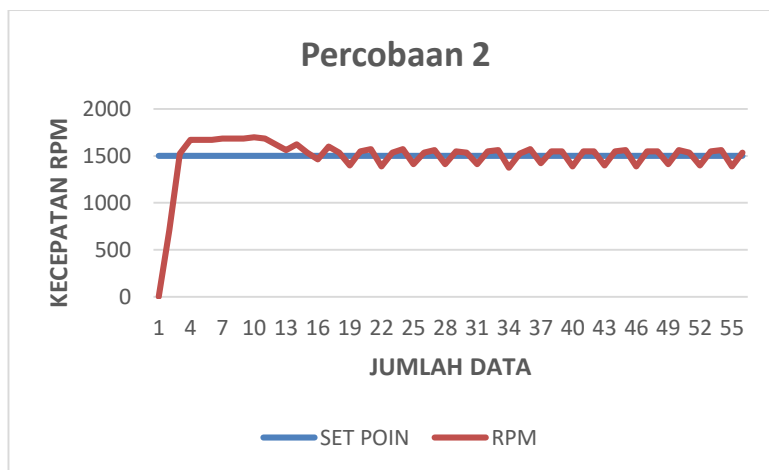
Gambar 14 Set RPM

Pengujian ini dilakukan sebanyak 6 kali dengan nilai k_p , k_i , dan k_d yang berbeda-beda. Output yang dihasilkan berupa grafik kecepatan yang memiliki nilai *overshoot* yang berbeda. Hasil output yang memiliki nilai *overshoot* terkecil di peroleh pada $k_p = 0,04$, $k_i = 0,05$, $k_d = 0,004$ dengan *overshoot* bernilai 0,73, sedangkan untuk hasil output yang memiliki nilai *overshoot* terbesar diperoleh dengan nilai $k_p = 0,1$, $k_i = 0,5$, $k_d = 0,7$ dengan *overshoot* bernilai 13,13. Nilai *overshoot* yang kecil atau mendekati 0 itu membuktikan bahwa motor bergerak dengan baik dan stabil. Dari hal tersebut pengujian ini dilakukan agar mendapatkan nilai yang terbaik. Set inputan dapat dilihat pada gambar 11 sampai gambar 14.



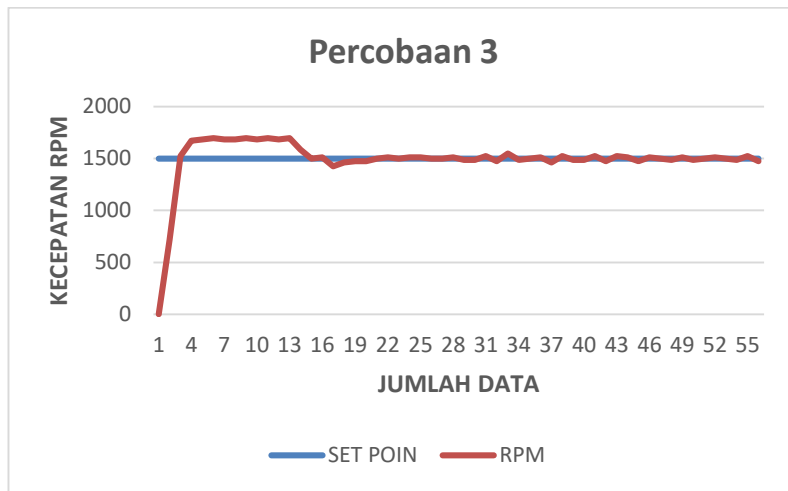
Gambar 15 Grafik perubahan kecepatan motor terhadap waktu, pada setting $k_p=1$, $k_i=0$, dan $k_d=0$

Pada percobaan 1 ini menggunakan inputan $k_p=1$ sedangkan untuk nilai k_i dan k_d bernilai 0. Grafik yang dihasilkan kecepatan motor tidak bisa mencapai nilai set poin dengan stabil masih terjadi kenaikan dan juga penurunan secara berterus-terusan.



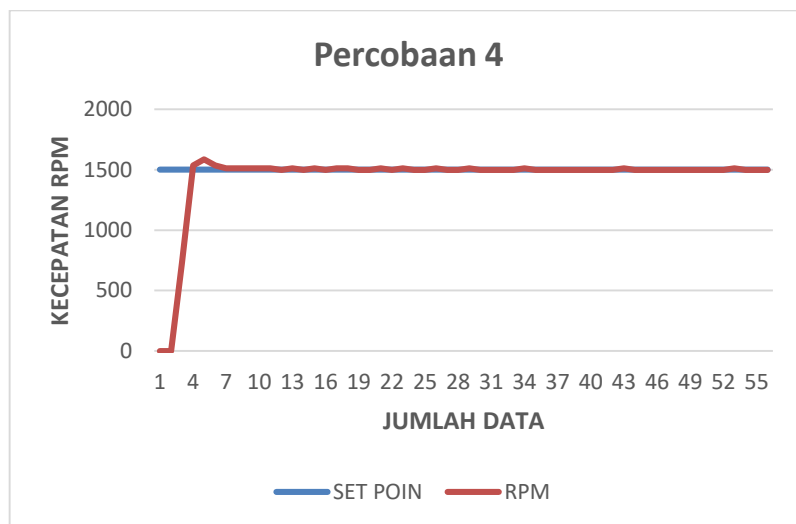
Gambar 16 Grafik perubahan kecepatan motor terhadap waktu, pada setting $k_p=1$, $k_i=0,5$, dan $k_d=0$

Pada percobaan 2 ini menggunakan inputan $k_p=1$, $k_i=0,5$, dan $k_d=0$. Grafik yang dihasilkan kecepatan dapat menyentuh set poin tapi terdapat overshoot yang tinggi dan kecepatan masih tidak stabil.



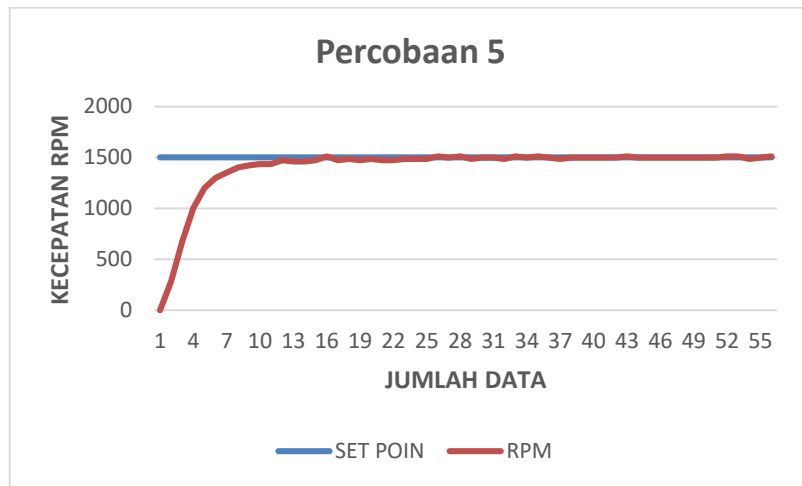
Gambar 17 Grafik perubahan kecepatan motor terhadap waktu, pada setting $kp=0,1$, $ki=0,5$, dan $kd=0,7$

Pada percobaan 3 ini menggunakan inputan $kp=0,1$, $ki=0,5$, $kd=0,7$. Grafik yang dihasilkan masih memiliki *overshoot* yang tinggi namun kecepatan yang dihasilkan sudah dapat bertahan pada set poin walaupun belum stabil.



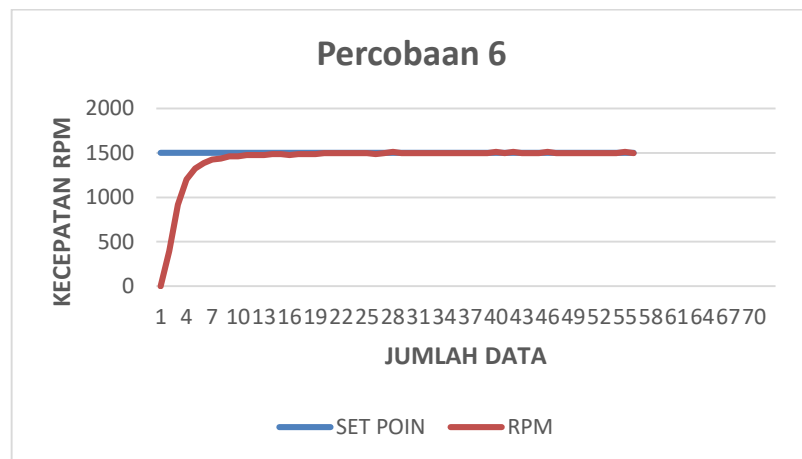
Gambar 18 Grafik perubahan kecepatan motor terhadap waktu, pada setting $kp=0,05$, $ki=0,06$, dan $kd=0,005$

Pada percobaan 4 ini menggunakan inputan $kp=0,05$, $ki=0,06$, $kd=0,005$. Grafik yang dihasilkan sudah bagus kecepatan sudah stabil, tapi masih terdapat nilai *overshoot* sedikit. Sehingga motor masih memiliki lonjakan kecepatan.



Gambar 19 Grafik perubahan kecepatan motor terhadap waktu, pada setting $kp=0,03$, $ki=0,04$, dan $kd=0,005$

Pada percobaan 5 ini menggunakan inputan $kp=0,03$, $ki=0,04$, $kd=0,005$. Grafik yang dihasilkan sudah sangat bagus, tidak terdapat *overshoot* dan juga kecepatan yang dihasilkan stabil. Tapi waktu mulainya terlalu lama atau delay yang dihasilkan ketika motor mulai berputar terlalu besar.



Gambar 20 Grafik perubahan kecepatan motor terhadap waktu, pada setting $kp=0,04$, $ki=0,05$, dan $kd=0,004$

Pada percobaan 6 ini menggunakan inputan $kp=0,04$, $ki=0,05$, $kd=0,004$. Grafik yang dihasilkan sudah sangat bagus mendekati nilai sempurna, karena grafik tidak memiliki *overshoot* dan waktu mulainya juga cepat, maka motor yang berputar akan cepat dan stabil.

Dapat dilihat terdapat perbedaan pada setiap percobaan, pada percobaan pertama dilakukan inputan kp saja karena mencoba menemukan nilai *overshoot* yang baik, lalu pada percobaan kedua menambahkan ki agar mengurangi nilai dari osilasi yang dihasilkan oleh gelombang. Pada percobaan ketiga menambahkan nilai kd agar dapat menstabilkan kecepatan sesuai dengan set poin yang di masukan atau kd biasa disebut dengan rem. Pada percobaan 4 menentukan nilai kp , ki , kd yang lebih baik tapi masih terdapat *overshoot* sedikit. Pada

percobaan kelima sudah tidak terdapat *overshoot* tapi kecepatan yang dihasilkan terlalu lambat. Pada percobaan keenam sudah menghasilkan kecepatan yang terbaik karena pergerakan motor tidak lambat dan juga tidak memiliki *overshoot*.

Tabel 1. Hasil Percobaan alat

NO	kp	ki	kd	Overshoot	Settling Time
1	1	0	0	0,73	NaN
2	1	0,5	0	13,13	NaN
3	0,1	0,5	0,7	13,13	37,13
4	0,05	0,06	0,005	5,6	6,24
5	0,03	0,04	0,005	0,73	14,67
6	0,04	0,05	0,004	0,73	10,66

Alat ini telah diriset secara langsung oleh Asisten Lab Teknik Elektro UMS dan telah digunakan pada Praktikum yang sebenarnya yaitu Praktikum Dasar Teknik Kendali (DTK). Didapatkan hasil yang bagus dan alat berjalan dengan baik.

4. PENUTUP

Dari semua pengujian yang dilakukan dapat disimpulkan sebagai berikut. Dalam pengendali kecepatan motor dc dengan berbasis arduino ini didapatkan nilai nilai *overshoot* yang kecil dan pergerakan awal motor yang cepat membuktikan bahwa motor bergerak dengan kecepatan yang bagus dan stabil. Berdasarkan nilai *overshoot*, *settling Time*, dan *rise Time*, nilai pengendali PID terbaik yaitu $kp = 0,04$, $ki = 0,05$, dan $kd = 0,004$ mendapatkan nilai *overshoot* sebesar 0,73. Alat ini sudah berhasil menyelesaikan permasalahan yang dihadapi oleh Laboratorium Elektro UMS terkait dengan pelaksanaan Praktikum Dasar Teknik Kendali (DTK).

PERSANTUNAN

Kepada Allah SWT yang telah memberikan rahmat dan memberikan kesehatan sehingga dapat mengerjakan dengan lancar, dan kedua kepada orang tua yang telah mendukung secara moral, mental, dan juga material serta dorongan semangat.

DAFTAR PUSTAKA

- Rizky, H. & Subuh, I. H. (2020). Rancang Bangun Kontrol Motor DC Dengan PID Menggunakan Perintah Suara dan Monitoring Berbasis internet Of Things(IOT). *Jurnal Teknik Elektro*, 09, 477-485.
- Reza, M. & Krismadinata. (2020). Kendali Kecepatan Motor DC dengan Kontroller PID dan Antarmuka Visual Basic. *JTEV(Jurnal teknik Elektro dan Vokasional)*, 06, 328-329.
- Dio, P. & Aswardi. (2020). Sistem Kendali Kecepatan Motor DC Berbasis Arduino dengan Metode Open Loop. *JTEV(Jurnal Teknik Elektro dan Vokasional)*, 06, 187-197.
- Marfanri, L., Ifan, W., & Wrastawa, R. (2020). Perancangan balancing Robot Beroda Dua dengan Metode Pengendalian PID Berbasis Arduino Nano. *JJEEE (Jambura Journal Of Electronics Engineering)*, 02, 39-43.
- Muhammad, I. E. & Muldi, Y. (2020). Kendali Kecepatan Motor DC Menggunakan DC Chopper Satu Kuadran Berbasis Kontroller PI. *JTEV (Jurnal Teknik Elektro dan Vokasional)*, 06, 296-305.
- Muhammad & Ali. (2004). Pembelajaran Perancangan Sistem Kontrol PID dengan Software Matlab. *Jurnal Edukasi*, 01, 1-8.
- Radi, B., Noer, S., & Osea, Z. (2015). Pengendali Kecepatan Motor DC Menggunakan perintah Suara Berbasis Perintah Suara Berbasis Mikrokontroller Arduino. *Jurnal Rekayasa dan teknologi Elektro*, 09, 96-107.
- Mila, D. I. P., Alfian, M., & Riky, D. P. (2021). Pengendali Kecepatan Sudut Motor DC Menggunakan Kontrol PID dan Tuning Ziegler Nichols. *Techno*. 23, 09-18.
- Ilham, A., Nanang, I., & T. D. Rachmilda. (2021). Rancang Bangun Pendeteksi Posisi Sudut dan kecepatan sesaat Dengan Menggunakan Rotary Encoder KY-040. *SENTER*. pp, 287-293.