

Penerapan PID Kontrol Untuk Pengendalian Kecepatan Motor DC Stepper Pada Pemposisi Hasil Cetak *Filament* (3D Printing) Di Gulungan Berbahan Daur Ulang

Chandra Andreas Setyo Wibisono , Budhy Setiawan, Indrazno Siradjuddin

Abstrak — 3D Printing merupakan terobosan yang baru di bidang teknologi, yakni mampu mencetak sesuatu hal yang sama persis di dalam software yang kita inginkan, namun terkadang dalam proses pencetakan sendiri sering terjadi kegagalan, dalam penelitian ini cetakan 3D yang gagal akan di daur ulang lagi sehingga mengurangi biaya produksi. Tujuan dari alat ini yaitu mengontrol kecepatan putaran motor dc stepper agar pada proses penggulungan hasil cetak daur ulang filament 3D Printing tidak menumpuk pada satu sisi. Pemposisi hasil cetak gulungan filament 3D Printing ini dirancang dengan menggunakan beberapa komponen dan mekanik yang terdiri dari : Motor DC Stepper, Sensor Rotary Encoder, Sensor Optocoupler, Sensor Obstacle Infrared dan Sistem pengendalian menggunakan Arduino Mega dan Kontrol PID. Sistem ini berfungsi mengontrol kecepatan putar motor dc stepper hasil dari cetakan filament (3D Printing) diharapkan dari motor stepper dapat selaras dengan motor di penggulungan. Hal ini perlu dilakukan agar hasil dari cetakan filament (3D Printing) pada proses penggulungan tidak menumpuk pada satu sisi. Salah satu metode yang digunakan untuk mengontrol kecepatan motor stepper adalah PID. Perancangan kontroler PID menggunakan kurva reaksi Ziegler Nichols menghasilkan nilai $K_p = 1,122$ $K_i = 0,33$ dan $K_d = 0,95$. Hasil yang di dapatkan antara lain rise time (tr) sebesar 0,14 s, time settling (ts) sebesar 55,51 s dan Percent Overshoot (Po) sebesar 38,77%. Respon sistem lebih bagus menggunakan Trial Error dengan rise time (tr) sebesar 1,3 s, settling time (ts) sebesar 18,4 s, peak time (tp) sebesar 2,1 s, dan Percent Overshoot (Po) sebesar 5.8% yang memiliki nilai $K_p = 0,8$, $K_i = 0,42$ dan $K_d = 0,05$. Pengujian sistem dilakukan dengan membandingkan 2 Gain yang berbeda. Proses akhir dari alat daur ulang filament (3D Printer) yaitu berat pada penggulungan sudah mencapai 1 kg.

Kata kunci : PID, Motor DC Stepper, *Filament* (3D Printing)

I. PENDAHULUAN

3D Printing merupakan sebuah teknologi baru di dalam teknologi. 3D Printing adalah printer yang dapat mencetak sesuatu benda dalam bentuk tiga dimensi, bukan berupa gambar ataupun tulisan di atas kertas. Kelebihan dari 3D Printing adalah yakni dapat membuat banyak bentuk rumit. Ini disebabkan kebebasan

gerak *printing* pada bentuk 3D. Pengertian dari 3D Printing dapat berfungsi dalam dunia manufaktur. Ukuran dari 3D Printer *filament* juga harus konstant yaitu berdiameter 1.75mm sampai 2.75mm, tetapi rata rata pengguna 3D Printer menggunakan *filament* yang berukuran 1.75mm. Dengan begitu dirancanglah suatu alat yang dapat meringankan biaya dan mengurangi sampah plastik yaitu proses mendaur ulang sampah plastik menjadi *filament*.

Dalam perencanaan pembuatan alat daur ulang *filament* (3D Printing) yaitu, dengan cara menghancurkan plastik tidak pakai atau hasil dari 3D Printing yang gagal berjenis HDPE ke *Crusher* sehingga menghasilkan biji plastik berukuran 2mm sehingga dapat mengurangi biaya pengeluaran suatu 3D Print. *Filament* merupakan bahan dasar dari alat 3D Printing untuk membuat suatu produk yang berbahan plastik berbentuk tiga dimensi.

Alat pemposisi *filament* (3D Printing) di buat menggunakan motor *stepper* sebagai penggerak ulir dan dapat di atur kecepatannya agar penggulungan rapi dan tidak menumpuk pada satu sisi saja. Untuk menjaga kestabilan produksi penggulungan maka perlu di buat kontrol kecepatan motor *stepper* dengan metode PID pada alat pemposisi hasil cetak *filament* (3D Printing) berbahan daur ulang.

II. TINJAUAN PUSTAKA

2.1 Plastik

Plastik digunakan dalam berbagai hal. Plastik bisa dikelompokkan jadi dua macam yaitu thermosetting dan thermoplastic. Thermoplastic adalah jenis plastik yang bila dipanaskan sampai suhu tertentu, akan meleleh dan hasil lelehan tersebut bisa dibentuk sesuai dengan yang kita inginkan . Berikut ini adalah penjelasan beberapa nomor kode plastik beserta kegunaannya dalam kehidupan sehari-hari.



Gambar 1 Nomer kode plastik

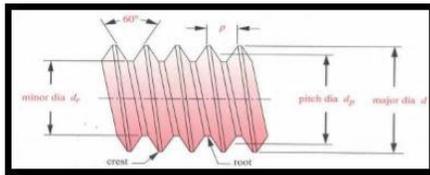
2.2 Mesin Filament Winding

Filament winding merupakan salah satu metoda pembuatan pipa komposit dimana serat digulung-gulung di mandrel yang terlebih dahulu dimasukkann kedalam cairan resin. Proses penggulungan serat dilaksanakan dengan

otomatis, yaitu menggunakan alat yang dilakukan dengan cara berulang-ulang hingga diperoleh jarak dan kepadatan yang sesuai

2.3 *Ulir*

Ulir adalah suatu peralatan yang memakai “baut-mur” contohnya ialah perancangan sebuah mesin yang paling sederhana. Namun dalam penerapan di dunia nyata, keberhasilan ataupun kegagalan peralatan dapat ditentukan oleh pemilihan dan penggunaan sistem sambungan baut-mur



Gambar 2 Terminologi geometri ulir

2.4 *Motor Stepper*

Motor *Stepper* yaitu berbeda dengan motor DC biasa dikarenakan dapat dikendalikan langsung oleh komputer ataupun mikrokontroler. Kelebihan motor ini adalah motor ini bergerak menurut rentetan pulsa yang dimasukan. Ketika beberapa pulsa diterima, maka oleh mikrokontroler maka poros akan berubah melalui step per step dan kelebihan ini membuat motor stepper ideal untuk ketepatan. Kecepatan putaran motor *stepper* (rpm) dapat diekspresikan memakai kecepatan pulsa (pps) sebagai berikut :

$$\omega = \frac{2 \cdot \pi \cdot n}{60} [rps] \dots\dots\dots(1)$$

- ω = Rps (rotasi/rpm)
- n = Step/ Putaran (pulsa)

Untuk rumus torsi bisa dihitung berdasarkan selisih kerja daya motor stepper ini terhadap kecepatan putaran atau dapat dirumuskan sebagai berikut [3] :

$$\tau = \frac{P}{\omega} \dots\dots\dots(2)$$

- τ = Torsi
- P = Daya Kerja Motor
- ω = Kecepatan sudut (rps)

Tabel 1 Urutan Pergantian Step pada Motor *Stepper* Searah dengan Jarum Jam

Step	A	B	A-	B-
1	+	+	-	-
2	-	+	+	-
3	-	-	+	+
4	+	-	-	+

2.5 *Arduino Mega 2560*

Arudino ialah sebuah platform pembuatan purwarupa elektronik yang bersifat open-source, hardware yang berdasarkan pada perangkat keras dan perangkat lunak dalam menciptakan obyek atau lingkungan yang interaktif .



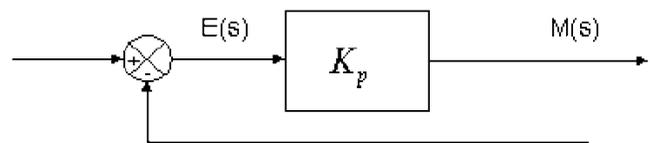
Gambar 3 Arduino Mega 2560

2.6 *Pengontrol PID*

Pada suatu sistem kontrol kita mengenal adanya beberapa macam aksi kontrol, diantaranya yaitu kontrol *proporsional*, kontrol *integral* dan kontrol *derivative*. Masing-masing kontrol ini mempunyai keunggulannya, dimana kontrol *proporsional* memiliki kelebihan *rise time* cepat, kontrol *integral* memiliki kelebihan yang berguna untuk mempersedikit *error*, dan *derivative* kontrol memiliki kelebihan untuk mempersedikit *error* atau meredam *overshot*. Untuk menghasilkan output dengan *risetime* yang cepat dan dengan *error* yang lebih sedikit dapat menyatukan ketiga kontrol ini dan menghasilkan kontrol PID. Pengontrol PID berdasarkan dari meninjau dari sifat yang di atur pada *plant*. Dengan begitu serumitnya apapun *plant*, *plant* tersebut harus di ketahui terlebih dahulu sebelum menemukan nilai PID itu.

2.6.1 *Pengontrol Proporsional*

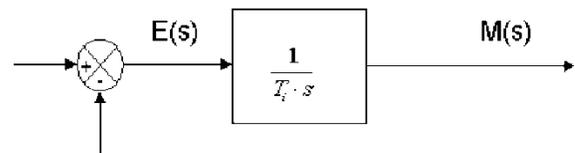
Menurut Ogata Sistem Kontrol Pid (*Proportional-Integral-Derivative Controller*) yaitu kontroler untuk menentukan ketepatan sistem instrumentasi dengan adanya feedback pada sistem tersebut seperti pada gambar di bawah ini.



Gambar 4 Diagram Blok Kontroler Proporsional

2.6.2 *Pengontrol Integral*

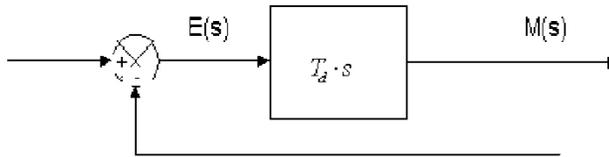
Pengontrol *integral* dimaksudkan untuk meniadakan kesalahan dalam kondisi mantap tanpa mengubah karakteristik-karakteristik *frequency* nol yakni pada kondisi mantap. Kontrol integral digunakan bersama tipe *P* dan *D*, namun pada hal-hal dimana kecepatan respons dan ketidak stabilan bukan merupakan masalah, tipe *P I* adalah cukup. Walaupun, penambahan tipe dari *P* perlu diperhatikan karena efeknya mengurangi kestabilan yakni karena mengakibatkan bertambahnya keterlambatan fasa.



Gambar 5 Blok Diagram Besaran Error dengan Pengontrol Integral

2.6.3 Pengontrol Derivative

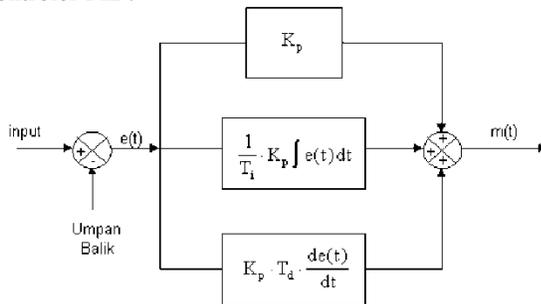
Output dari pengontrol *Derivative* mempunyai karakteristik seperti kontrol differensial. Perubahan yang cepat pada input kontrol, akan berefek juga pada perubahan yang sangat besar dan cepat.



Gambar 6 Blok Diagram Pengontrol Derivative

2.6.4 Pengontrol PID

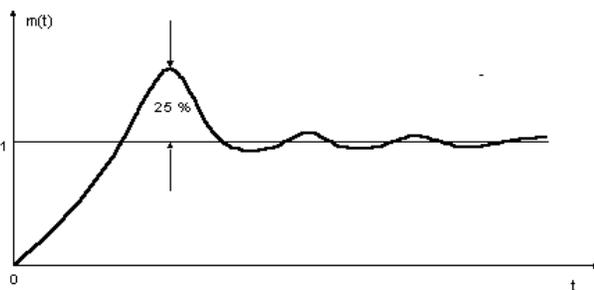
Setiap plus dan minus dari P, I dan D kontrol mampu menutupi dengan menjadikan ketiganya secara parallel sehingga menjadi pengontrol *proposional* ditambah *integral* ditambah *derivative* (pengontrol PID). Elemen-elemen pengontrol P I D secara keseluruhan berguna untuk mempercepat tanggapan *plant*, menghilangkan *offset* dan menghasilkan perubahan awal yang besar. Gambar 7 menunjukkan blok diagram kontroler PID.



Gambar 7 Blok Diagram Kontroler PID Analog

2.6.5 Metode Ziegler-Nichols

Metode dari Ziegler-Nichols pertama kali diperkenalkan pada tahun 1942. Ziegler-Nichols memperkenalkan dua metode yaitu osilasi dan kurva. Kedua metode diperuntukan menghasilkan tanggapan sistem dengan overshoot maksimum sebesar 25%. Gambar 8 memperlihatkan kurva dengan lonjakan 25%



Gambar 8 Respons yang Memperlihatkan 25% Lonjakan Maksimum

2.7 Sensor Rotary Encoder

Rotary Encoder adalah instrument yang menghasilkan keluaran dalam bentuk sinyal digital. Prinsip kerja dari

instrument ini adalah menghasilkan pulsa berupa gelombang kotak dari pembacaan sensor yang mendeteksi cahaya dari LED yang melewati lubang-lubang pada piringan yang berputar satu poros .

2.8 Sensor Optocoupler

Sensor *Optocoupler* ialah modul sensor yang memiliki sumber cahaya yang berpasangan dengan penerima. Sumber cahaya dan penerima terbungkus dalam satu komponen. Sumber cahaya biasanya berupa LED atau *infrared* LED dan penerima biasanya berupa phototransistor.

2.8 Sensor Obstacle Infrared

Sensor *Obstacle Infrared* atau Phototransistor adalah komponen yang bekerja dengan memanfaatkan cahaya yang menerangi pada basis transistor. Phototransistor terdapat jenis NPN dan PNP. Transistor ini berpengaruh pada sensitivitas terhadap cahaya. Semakin besar intensitas cahaya maka semakin besar juga arus. Sensor cahaya yang digunakan ini terdiri dari sumber cahaya berupa LED dan Phototransistor yang memberikan respon pada perubahan intensitas cahaya

III. METODE PENELITIAN

Dalam bab ini akan menjelaskan mengenai perancangan keseluruhan sistem yang terdiri dari tiga tahapan yaitu perancangan mekanik, perancangan elektrik dan perancangan kontrol.

3.1 Perancangan Mekanik

Alat pemosisi hasil cetak *filament* (3D Printing) ini menggunakan motor *stepper* sebagai aktuator pengarah dari hasil cetakan *filament* (3D Printing) ke alat *winding*. Ulir atau screw digunakan untuk mengubah suatu gerakan angular untuk menjadi suatu gerakan linear. Dalam perancangan alat ini terdiri dari :

- Ulir / screw dengan Panjang 15 cm
- Motor *stepper* sebagai aktuator

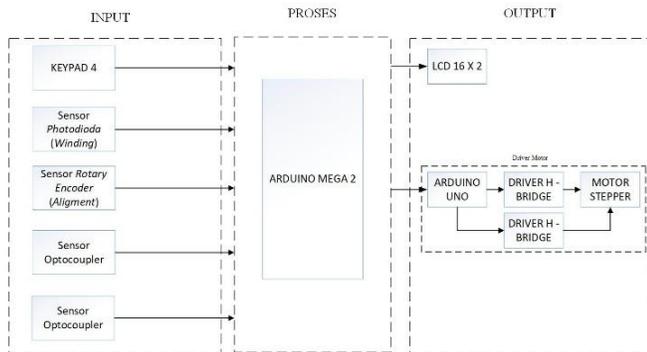
Berikut adalah design dari mekanik.



Gambar 9 Design Mekanik Pemosisi Filament Tampak Depan

3.2 Blok Diagram Sistem Pemposisi Filament

Di dalam perancangan dan pembuatan suatu alat, terdapat komponen - komponen utama ataupun pendukung sehingga perlu digambarkan dengan blok diagram.



Gambar 10 Diagram Blok Sistem Pemposisi Filament

Dari Gambar 10 dapat dijelaskan fungsi dari masing-masing

1. Keypad

Digunakan sebagai perangkat interaksi antara pengguna untuk melakukan pengaturan nilai set point Motor Stepper dalam RPM.

2. Sensor *Obstacle Infrared* (Winding)

Sensor kecepatan motor *winding* menggunakan sensor *Obstacle Infrared* untuk men-sensing kenaikan atau penurunan kecepatan.

3. Sensor *Rotary Encoder* (Alignment)

Sensor kecepatan motor *stepper* menggunakan *rotary encoder* sebagai pembacaan nilai RPM dari motor *stepperg* dan nantinya akan di kuatkan ke dalam kontrol PID.

4. Sensor *Optocoupler*

Sensor *Optocoupler* sebagai pembatas kanan atau kiri dari pemposisi hasil *filament* (3D Printing) nantinya jika salah satu *optocoupler* aktif akan memutar motor searah jarum jam atau berlawanan jarum jam.

5. LCD 16x2

Digunakan untuk menampilkan *Gain* kecepatan motor *stepper* yang diberikan pada awal proses, serta menampilkan kecepatan putar motor dc *stepper* yang sedang berlangsung atau sedang menjalankan proses pemposisi hasil cetak *filament* (3D Printing).

6. Mikrokontroler Arduino Mega 2560

Sebagai mikrokontroler yang mendapatkan data dari sensor *rotary encoder* (Winding) dan *rotary encoder* (Alignment) yang nantinya akan mengeluarkan *output* untuk mengatur kecepatan dari motor *stepper* dengan menggunakan metode PID.

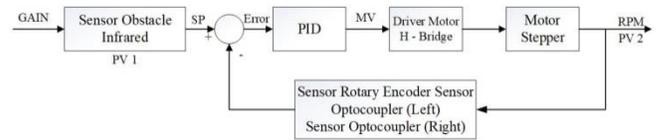
7. Driver Stepper Motor

Untuk mengatur dan menguatkan sinyal dari mikrokontroler menuju motor *stepper*.

8. Motor Stepper

Sebagai aktuator yang dikontrol dalam proses pemposisi hasil cetak *filament* (3D Printing) berbahan daur ulang.

3.3 Blok Diagram Kontrol Pemposisi Filament



Gambar 11 Blok Diagram Kontrol Pemposisi Filament

1. Sensor *Obstacle Infrared* (Winding)

Digunakan untuk men-sensing kecepatan spooling pada motor DC (Winding) dan nantinya akan di baca juga oleh *Rotary Encoder* (Alignment) sehingga pergeseran dari pemposisi *filament* akan selaras dengan motor *Winding*.

2. PID Kontrol

Digunakan untuk mengatur kecepatan pergeseran pemposisi *filament* (3D Printing) dengan *feedback* sensor *Rotary Encoder* (Alignment) dan sensor *optocoupler*.

3. Driver Motor

Digunakan untuk pengaturan step yang masuk ke motor *stepper* melalui *output gain* mikrokontroler *Arduino Mega*.

4. Motor Stepper

Sebagai aktuator yang menghasilkan gerak putar pemposisi.

5. Sensor *Rotary Encoder* (Alignment)

Sebagai pembacaan nilai RPM dari motor dc (Winding) dan nantinya akan di kuatkan ke dalam sistem kontrol PID.

6. Sensor *Optocoupler*

Sebagai pembatas kiri dan kanan dari pemposisi *filament* (3D Printing).

3.4 Spesifikasi

Spesifikasi hardware dalam perancangan alat ini terdiri dari spesifikasi elektronik dan spesifikasi mekanik.

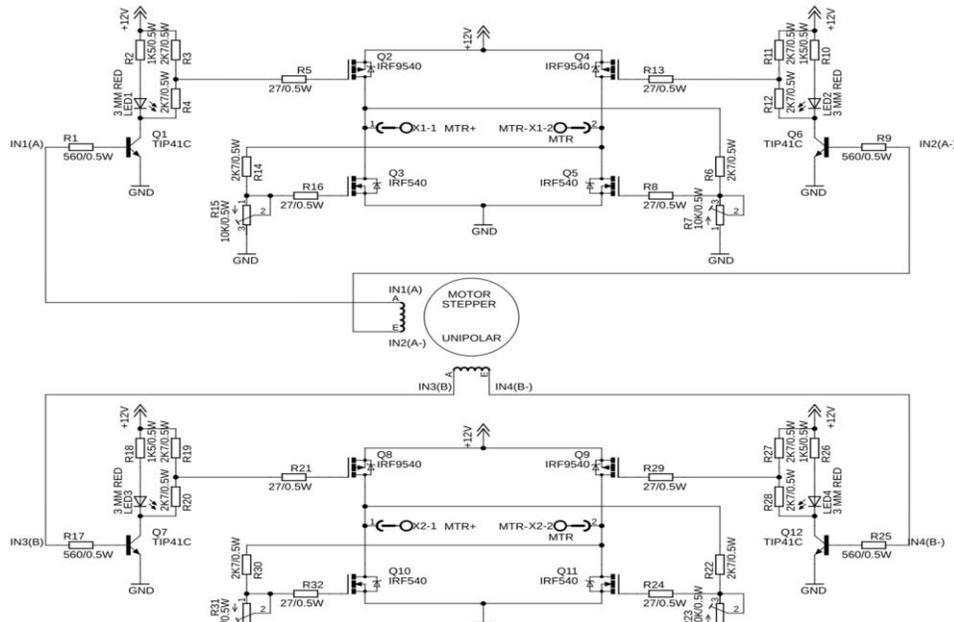
- **Spesifikasi Mekanik**

- 1) Dimensi Alat Pemposisi *Filament*
 Panjang : 15 cm
 Lebar : 10 cm
 Tinggi : 10 cm
- 2) Dimensi *Box Control*
 Panjang : 25 cm
 Lebar : 9 cm
 Tinggi : 35 cm

- **Spesifikasi Elektronik**

- 3) *Microcontroller* : *Arudino Mega 2560*
- 4) Tampilan : LCD 16x2
- 5) Sensor *Obstacle Infrared* (Winding)
- 6) Sensor *Rotary Encoder* (Alignment)
- 7) Sensor *Optocoupler* (Right)
- 8) Sensor *Optocoupler* (Left)
- 9) Motor : DC Stepper
- 10) Tegangan Kerja : 220 VAC

Gambar 13 Rangkaian Driver Motor Stepper H-BRIDGE



3.5 Perancangan Driver Motor H-BRIDGE

Pada gambar 13 diatas merupakan rangkaian driver motor H – Bridge DC Stepper yang berfungsi untuk mengatur pergerakan dari stepper motor pada proses pemosisi hasil cetak filament (3D Printing), terdapat 2 buah driver H BRIDGE pada alat pemosisi filament yang akan butuhkan untuk motor stepper 4 kanal, dikarenakan motor 4 kanal memiliki 4 kabel dengan konfigurasi (A,A-) dan (B,B-) yang akan di masukkan ke output dari kedua driver di atas , untuk mengatur pergerakan dari stepper motor diperlukan tegangan maksimum 12 VDC dan arus nominal 0,9 Ampere oleh karena itu digunakan MOSFET IRF9540 ini memiliki kemampuan arus yang masuk drain (ID) hingga -19 A dengan VDS hingga -100 V. Lalu untuk batasan tegangan antara source dan gate (VGS) adalah ± 20 V. MOSFET kanal-p mengalirkan arus ID (hidup) jika tegangan pada gate (VG) lebih negatif daripada tegangan pada source (VS). Sebaliknya, tidak ada arus ID (mati) jika tegangan pada gate (VG) sama atau lebih positif daripada tegangan pada source (VS). Nilai VS terbesar adalah 12 V sehingga dibutuhkan pembagi tegangan pada gate agar pada saat hidup VGS memenuhi syarat. Jika arus pada pembagi tegangan (IPT) sebesar 5 mA dan V adalah VS. Disini 12 VDC diperoleh dari Power Supply 12V.

Transistor TIP41 yang digunakan pada driver motor stepper ini sebagai komponen yang berfungsi men-trigger MOSFET IRF9540 (P), untuk memutar motor stepper. Kaki basis Transistor TIP41 diberi masukan berupa sinyal input dari digital I/O pin ArduinoMega 2560 kemudian dihubungkan dengan resistor 560 Ω, dan kaki Collector Transistor TIP41 dihubungkan dengan R2 sebagai pembatas arus yang melewati LED 3mm.

Rangkaian driver motor stepper H - Bridge ini menggunakan R1 560 Ω (4buah), R2 1K5Ω (4buah) dan R32K7Ω (4buah) dengan perhitungan berikut ini:

- Q1 & Q2 = Transistor TIP41
 $\beta = 10$
 $I_b = \frac{I_c}{\beta} = 10 \text{ mA}$
- R1 = Resistor basis Q1
 $R1 = \frac{V_{out \text{ Arduino}}}{I_b} = 500 \approx 560 \Omega$
 $P_{R1} = V_{R1} \times I_{R1} = I^2 B \cdot R1 = 100 \times 10^{-3} \cdot 560 \Omega = 0.056 \text{ Watt} \approx 0.5 \text{ W}$
- R2 & R9 = LED
 $I_f = 16 \text{ mA}$
 $V_f = 1,8 \text{ V}$
 $R2 \ \& \ R9 = \frac{24 - 1,8 \text{ v}}{16 \text{ mA}} = \frac{22,2}{16} = 1,3875 \approx 1K5 \Omega$
- R3 & R8
 $R3 \ \& \ R8 = \frac{V_s - V_{ce}}{I_{R2}} = 2,3K \approx 2K7 \Omega$

IV. HASIL DAN ANALISA

4.1 Pengujian Sensor Obstacle Infrared

Pengujian *sensor Obstacle Infrared* dilakukan dengan cara mengkalibrasi dengan kecepatan dari motor dc (*Winding*) agar pembacaan sensor sesuai dengan *tacho meter*.

Tabel 2 Hasil Pengujian Sesnro *Obstacle Infrared*

Duty Cycle (%)	Obstacle Infrared (rpm)	Tacho Meter (rpm)	Error (%)
0	0	0	0
10	25.6	25.8	0.775194
20	54.9	56	1.964286
30	73	73.2	0.273224
40	82.48	82.7	0.266022
50	87.22	87.7	0.54732
60	91.01	91.4	0.426696
70	92.9	91	2.08791
80	94.8	95.3	0.524659
90	96.7	96.8	0.103306
100	98.59	99.4	0.814889

Hasil pengujian sensor *rotary encoder* membuktikan bahwa terdapat perbedaan hasil sensor dengan *tachometer*. Rata-rata dari *error* setiap pembacaan sensor terhadap perubahan adalah 0.19 %. Ini menunjukkan bahwa *error* tersebut masih dapat ditoleransi dan *rotary encoder* tersebut layak digunakan.

Selain itu, hasil sensor yang tertampil pada LCD tidak dapat menampilkan nilai yang pasti, selalu berubah-ubah namun pada *range* tertentu. Hasil pada *tachometer* juga demikian. Hal ini yang menyebabkan terjadinya *error* namun tidak terlalu besar.

4.2 Pengujian Sensor Rotary Encoder

Pengujian *sensor Rotary Encoder* dilakukan dengan cara mengkalibrasi dengan kecepatan dari motor *stepper* agar pembacaan sensor sesuai dengan *tachometer*.

Tabel 3 Hasil Pengujian Sensor *Rotary Encoder*

MV	Rotary Encoder (rpm)	Tacho Meter (rpm)	Error (%)
0	0	0	0
20	10.3	10	0.3
30	14.85	14.8	0.33784
40	19.6	19.8	1.010101
50	24.5	24.8	1.209677
60	30.47	30	1.56667
70	36.14	35	3.25714
80	40.7	39.9	2.00501
90	45.73	44.9	1.84855
100	50.44	49.7	1.48893

Hasil pengujian sensor *rotary encoder* membuktikan bahwa terdapat perbedaan hasil sensor dengan *tachometer*. Rata-rata dari *error* setiap pembacaan sensor terhadap perubahan adalah 1,25%. Hal ini menunjukkan bahwa *error* tersebut masih dapat ditoleransi dan *rotary encoder* tersebut

layak digunakan walaupun dalam pembacaan terdapat perubahan nilai (± 5 rpm).

Selain itu, hasil sensor yang tertampil pada LCD tidak dapat menampilkan nilai yang pasti, selalu berubah-ubah namun pada *range* tertentu. Hasil pada *tachometer* juga demikian. Hal ini yang menyebabkan terjadinya *error* namun tidak terlalu besar.

4.3 Pengujian Sensor Optocoupler

Pengujian sensor *Optocoupler* dilakukan dengan menggunakan *Volt Meter* untuk mengetahui logika tegangan output pada sensor *optocoupler*. Berikut adalah tampilan logika tegangan pada *Volt meter*.



Gambar 14 Hasil Pengujian Sensor *Optocoupler* dengan *Volt Meter* Terhalang Pembatas



Gambar 15 Hasil Pengujian Sensor *Optocoupler* dengan *Volt Meter* Tidak Terhalang Pembatas

4.4 Pengujian Driver Motor

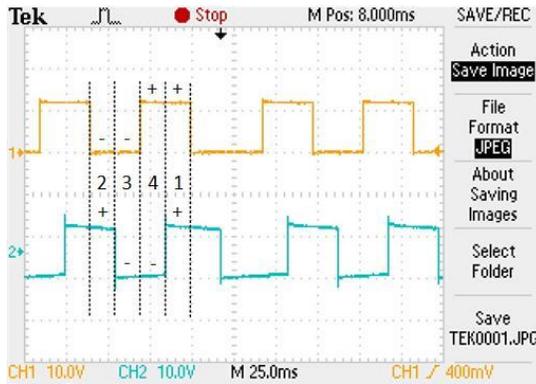
- Pengujian Step Driver *H-Bridge*

Pengujian driver motor *stepper* dilakukan dengan menggunakan *oscilloscope* untuk melihat gelombang dari driver motor *stepper H – Bridge* dengan nilai MV 20.

Tabel 4 Urutan Pergantian Step

Step	A	B	A-	B-
1	+	+	-	-
2	-	+	+	-
3	-	-	+	+
4	+	-	-	+

SEARAH JARUM JAM



Gambar 16 Hasil Pengujian Step Phasa A dan B berputar Searah Jarum Jam

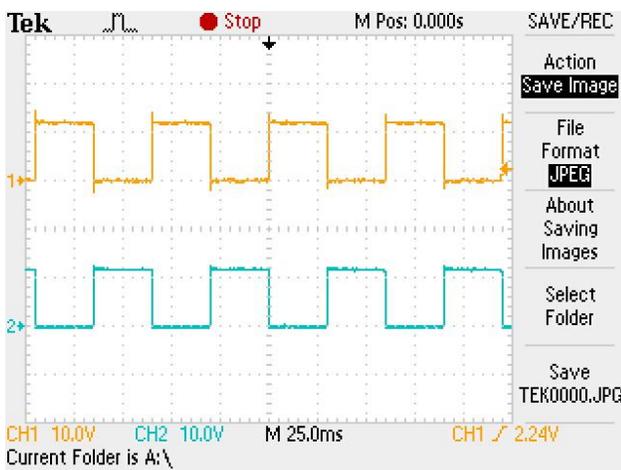
Dari hasil pengujian *driver* motor *stepper* menggunakan *oscilloscope* dapat dilihat bahwa pergeseran step sesuai dengan tabel IV Sehingga dapat disimpulkan bahwa *driver* motor *stepper* berkerja sesuai dengan semestinya.

- *Pengujian MV pada Driver H-Bridge*

Pengujian *driver* motor *stepper* dilakukan dengan menggunakan *oscilloscope* dan mengubah nilai MV dari 0 sampai dengan 100 untuk melihat hasil gelombang daru *driver* motor *stepper* H – Bridge.

Tabel 5 Hasil Pengujian Menggunakan *Oscilloscope*

MV	V/ div	Time div (ms/div)	Jumlah Div Time	T (ms)	F (hz)
10	10	25	4.8	120	8.3
20	10	25	2.4	60	16.67
30	10	25	1.8	45	22.22
40	10	10	3	30	33.33
50	10	10	2.2	22	45.45
60	10	10	2	20	50
70	10	10	1.7	17	58.82
80	10	5	3	15	66.66
90	10	5	2.6	13	76.92
100	10	5	2.4	12	83.33

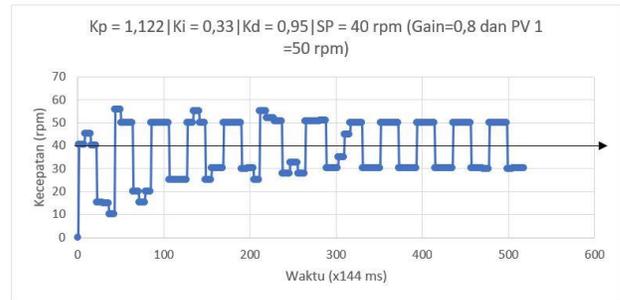


Gambar 17 Hasil Pengujian MV pada *Driver H-Bridge* dengan MV 20

Dari hasil pengujian *Manipulate Value* menggunakan *oscilloscope* dengan mengubah nilai 0 sampai dengan 100 dapat di lihat kecepatan motor *stepper* meningkat sesuai dengan peningkatan frekuensi pergantian step.

4.5 Pengujian Metode Osilasi PID Ziegler-Nichols

Gambar 18 adalah hasil grafik dengan pemberian konstanta $K_p = 1,86$ serta K_i dan K_d masing-masing bernilai 0 serta *setpoint* kecepatan diberi 40 rpm. Dimana konstanta K_p adalah nantinya berubah menjadi K_u apabila telah ditemukan respon sistem beresilasi

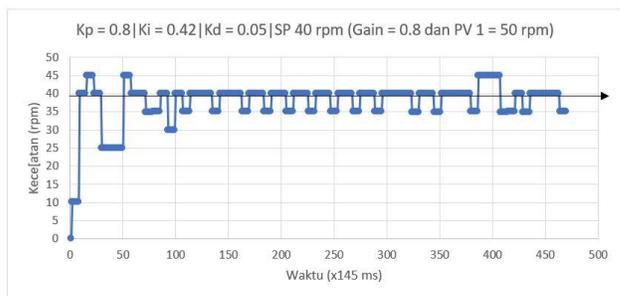


Gambar 18 Respon Pemposisi *Filament* Metode PID Ziegler-Nichols

Gambar dari grafik di atas menunjukkan bahwa kecepatan motor *stepper* dalam proses pemposisi *filament* (3D Printing) dengan kontrol memiliki *maximum peak* (M_p) 55,51 rpm dengan *peak time* (t_p) 1,29 s, kemudian menuju *setpoint* 51 s dengan, 49,2 rpm. Selain itu pada grafik juga menunjukkan bahwa kecepatan putar motor *stepper* dalam proses pemposisi *filament* (3D Printing) pada waktu *steady state* tidak tepat pada *setpoint* 40 rpm, namun selalu beresilasi antara 49,97 hingga 30,04 rpm dengan *Gain* sebesar 0,8 dan PV 1 sebesar 50 rpm sehingga menghasilkan *setpoint* 40 rpm.

4.6 Pengujian Metode PID Perbaikan dari Ziegler-Nichols

Pengujian metode perbaikan dari *Ziegler-Nichols* saat proses pemposisi *filament* (3D Printing) dilakukan dengan mengubah nilai K_p , K_i dan K_d diperoleh melalui cara melihat respon sistem berupa grafik yang kemudian menganalisis metode PID pada respon motor *stepper* dalam proses pemposisi *filament* (3D Printing). Hal ini digunakan untuk menentukan apakah dengan menggunakan metode ini lebih baik atau kurang baik.



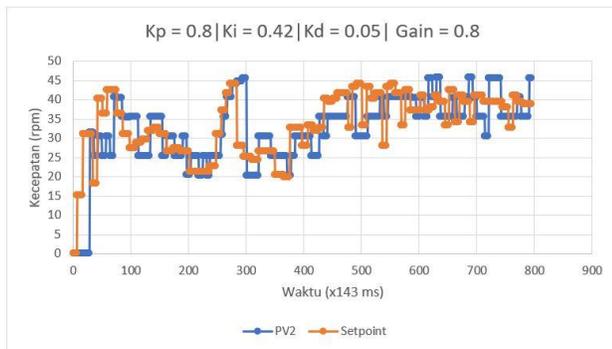
Gambar 19 Respon Pemposisi *Filament* Menggunakan Metode PID Perbaikan dari *Ziegler-Nichols*

Gambar grafik di atas menunjukkan bahwa kecepatan motor *stepper* pada proses pemposisi *filament* (3D Printing)

dengan kontrol beban memiliki *maximum peak* (Mp) 44,93 rpm dengan *peak time* (tp) 2,1 s, kemudian menuju *setpoint* pada 18,4 s dengan kecepatan 39,98 rpm. Selain itu pada grafik juga menunjukkan bahwa kecepatan putar motor *stepper* pada proses pemposisi *filament* (3D Printing) waktu *steady state* tidak tepat pada *setpoint* 40 rpm, namun selalu berosilasi antara 40,2 hingga 34,98 rpm dan *Gain* sebesar 0,8 dan PV 1 sebesar 50 rpm, sehingga dihasilkan *setpoint* 40 rpm.

4.7 Analisa Data

Berdasarkan hasil pengambilan data yang telah di ambil, dapat di analisa respon dari ssstem PID berkerja sesuai dengan *setpoint*. Perubahan PV 2 terjadi dikarenakan proses penggulangan tidak mencapai titik stabil sehingga menyebabkan kenaikan dan penurunan rpm pembacaan pada sensor *obstacle photodiode* yang terletak pada motor dc (*Winding*) dan itu mengakibatkan kecepatan motor *stepper* sulit mendapatkan titik *steady state*.



Gambar 20 Respon Sistem dengan Gain 0,8 Dengan beban *Filament* (3D Printing)

Berdasarkan grafik di atas *Gain* 0,8 hasil dari pemposisi *filament* (3D Printing) pada penggulangan menghasilkan penggulangan yang lebih rapi dikarenakan kecepatan motor *stepper* (*Aligment*) selaras dengan motor dc (*Winding*).



Gambar 21 Hasil Cetak Pemposisi *Filament* (3D Printing) Pada Penggulangan Dengan Gain 0,8

Berdasarkan percobaan di atas antara *Gain* 0,5 dengan *Gain* 0,8 hasil dari pemposisi *filament* (3D Printing) dengan *Gain* 0,8 menghasilkan penggulangan yang lebih rapi itu di sebabkan kecepatan dari motor *stepper* (*Aligment*) selaras dengan motor dc (*Winding*).

V. PENUTUP

5.1 Kesimpulan

Dari hasil perancangan dan pengujian yang telah dilakukan, dapat diambil kesimpulan bahwa pengaturan kecepatan otomatis telah berhasil dibuat dan bekerja dengan baik. Berikut ini adalah kesimpulan lain yang didapatkan :

1. Berdasarkan perancangan yang telah dilakukan bahwa nilai parameter K_p , K_i , dan K_d didapatkan dengan perhitungan metode *Ziegler Nichols* dengan nilai $K_p = 11,122$, $K_i = 0,33$, $K_d = 0,95$. Memiliki *rise time* (t_r) sebesar 0,14 s, *settling time* (t_s) 51 s, *peak time* (tp) 6,9 s dan *Percent Overshoot* (P_o) sebesar 38,77%.
2. Untuk hasil respon sistem dengan perancangan dirasa kurang baik sehingga diperbaiki dengan *trial error* ditemukan nilai $K_p = 0,8$ dan $K_i = 0,42$ dan $K_d = 0,05$ pada motor *stepper* dalam proses pemposisi *filament* (3D Printing), metode PID *trial error* mampu mempercepat sistem menuju *setpoint* yang ditentukan. Diperoleh *Percent Overshoot* (PO) 12,3 %, *Rise Time* (TR) 1,3 s saat *setpoint* 40 rpm, *Settling Time* (TS) 18,4 s saat 39,98 rpm, *Peak Time* (TP) 2,1 s saat 44,93 rpm.
3. Untuk hasil pengujian sistem dilakukan dengan 2 percobaan *Gain* dan didapatkan *Gain* 0,8 dengan hasil penggulangan yang lebih rapi dikarenakan kecepatan motor *stepper* (*Aligment*) selaras dengan motor dc (*Winding*).

5.2 Saran

1. Dari segi mekanik, sebaiknya ulir pada pemposisi di ganti menggunakan *Lead Screw* agar pergerakan di pemposisi *filament* (3D Printing) lebih berjalan mulus.
2. Dari segi *software*, perlu adanya metode kontrol lain sebagai perbandingan apakah metode baru yang digunakan lebih baik atau kurang baik dibandingkan metode PID (*Proportional Integral Derivative*) yang digunakan saat ini.

DAFTAR PUSTAKA

- [1] M Kambalur, B, DKK. 2016. *3D Printer Filament Extruder For Plastic Recycling*. Department of Electronics and Communication Engineering. B.M.S College Of Engineering.
- [2] Wirno Azhar, D. 2016. *Rancang Bangun Mesin Filament Winding*. Jurusan Teknik Elektro. Universitas Pasundan Bandung.
- [3] Hughes Austin. 2006. *Electric Motors and Drives Third Edition*. Elsevier ltd.
- [4] Nanda Guntu. 2017. *Prototipe Kunci Pintu Menggunakan Motor Stepper Berbasis Arduino Mega 2560 Dengan Perintah Suara Pada Android*. Jurusan Teknik Elektro. Universitas Lampung.
- [5] Ogata, K. 2010, *Modern Control Engineering Fifth Edition*, Prentice Hall : New Jersey.
- [6] S. Pakpahan. 1994. *Kontrol Otomatik Teori dan Penerapan Edisi Kedua*. Jakarta. Erlangga

- [7] Gunterus, Frans. 1994. Falsafah Dasar: Sitem Pengendali Proses.PT. Elex Media Komputindo. Jakarta.
- [8] Bukardi Sukma. 2015. Perancangan Dan Pembuatan Semi Automatic T-Shirt Folding Machine Menggunakan Metode Fuzzy Proportional Derivative.
- [9] S. Pakpahan. 1994. Kontrol Otomatik Teori dan Penerapan Edisi Kedua. Jakarta. Erlangga
- [10] Pallas-Areny, Ramon dan John G. Webster 2000. *Sensors and Signal Conditioning*. New York : Acid Free Paper.
- [11] O., Kasap S. 2000. *Optocouplectronic Devices and Photonics : Principles and Practices*. 2000. New Jersey : Prentice-Hall, Inc.