

# Pembacaan G-Code Gerak Extruder menggunakan Servo Motor BLDC untuk Firmware 3D Printer Simetris Bilateral 2x2x2 Meter

Resti Dyah Ayu Retno Palupi<sup>1</sup>, Budhy Setiawan<sup>2</sup>, Indrazno Siradjuddin<sup>3</sup>

e-mail: [restipalupi83@gmail.com](mailto:restipalupi83@gmail.com), [budhy.setiawan@polinema.ac.id](mailto:budhy.setiawan@polinema.ac.id), [indrazno@polinema.ac.id](mailto:indrazno@polinema.ac.id)

<sup>1,3</sup>Jurusan Teknik Elektro, Politeknik Negeri Malang, Jalan Soekarno Hatta No. 9 Malang, Indonesia

## Informasi Artikel

### Riwayat Artikel

Diterima 2 Juli 2024  
Direvisi 23 September 2024  
Diterbitkan 30 September 2024

### Kata kunci:

3D Printer  
G-Code  
Firmware  
BLDC

## ABSTRAK

3D printing mengubah desain digital menjadi objek nyata, tetapi loss step pada motor stepper. Penelitian ini bertujuan meningkatkan kontrol printer 3D skala besar melalui firmware khusus untuk motor servo BLDC pada sumbu X dan Y, guna meningkatkan efisiensi dan kualitas. Motor BLDC yang semakin terjangkau dan berkinerja tinggi mendorong kebutuhan solusi inovatif, karena firmware servo BLDC yang sesuai belum ada. Dengan menggunakan Close-Loop Interrupt Speed, firmware ini dirancang untuk meningkatkan kecepatan dan presisi cetak. Implementasi dan uji coba eksperimental akan memastikan kompatibilitas dan keandalan sistem dalam jangka panjang, khususnya pada mesin 3D printer skala besar. Hasil penelitian diharapkan dapat meningkatkan kontrol pergerakan ekstruder dan ekstrusi material, yang menghasilkan cetakan lebih presisi dan konsisten dalam skala besar.

## ABSTRACT

*3D printing converts digital designs into real objects, but loses steps in stepper motors. This research aims to improve the control of large-scale 3D printers through custom firmware for BLDC servo motors on the X and Y axes, in order to improve efficiency and quality. The increasingly affordable and high-performance BLDC motors drive the need for innovative solutions, as suitable BLDC servo firmware does not yet exist. By using Close-Loop Interrupt Speed, this firmware is designed to increase printing speed and precision. Experimental implementation and testing will ensure the compatibility and reliability of the system in the long term, especially on large-scale 3D printers. The results of the research are expected to improve the control of extruder movement and material extrusion, resulting in more precise and consistent printing on a large scale.*

### Keywords:

3D Printer  
G-Code  
Firmware  
BLDC

### Penulis Korespondensi:

Resti Dyah Ayu Retno Palupi,  
Jurusan Teknik Elektro,  
Politeknik Negeri Malang,  
Jl. Soekarno Hatta No.9, Malang, Indonesia, Kode Pos. 65141  
Email: [restipalupi83@gmail.com](mailto:restipalupi83@gmail.com)  
Nomor HP/WA aktif: +62 813-3325-3837

## 1. PENDAHULUAN

Teknologi 3D Printer merupakan sebuah inovasi baru yang menjadi bagian dari perkembangan dunia industri terutama pada bidang manufaktur [1]. Mesin ini dapat mencetak benda dalam bentuk tiga dimensi berdasarkan

p-ISSN: 2356-0533; e-ISSN: 2355-9195



9 772356 053009

desain CAD (*Computer Aid Design*) yang telah dibuat. Di Indonesia, 3D printer mulai digunakan pada industri dengan tujuan *rapid prototyping* yang memungkinkan proses pembuatan *prototype* dilakukan lebih cepat dan efisien [2]. Tahapan pembuatan *prototype* menggunakan 3D printer dimulai dengan mengonversi desain 3D (CAD) menjadi file digital, yang selanjutnya melalui serangkaian proses termasuk *slicing* menggunakan Simplify3D untuk menghasilkan G-Code. Setelah itu, file tersebut akan dibaca oleh Mikrokontroler dan setiap komponen seperti ekstruder dan motor stepper akan dimodifikasi menggunakan Marlin Firmware, sehingga dapat membentuk objek tiga dimensi yang nyata. [3].

Untuk pembuatan *prototype*, penggunaan teknologi 3D printer dianggap sangat ideal karena produk dibuat dari file digital yang berisi desain 3D yang dibuat melalui software CAD dan disimpan dalam format STL yang merupakan format yang paling sering digunakan dan dapat dibaca oleh 3D printer [5]. File dengan format STL tersebut kemudian diolah oleh mesin 3D printer untuk dicetak secara bertahap sehingga menjadi objek 3D yang sebenarnya [4]. Dalam pengembangan produk baru, teknologi 3D printing dapat meningkatkan efisiensi dan juga mempunyai keunggulan dalam mencetak komponen yang kompleks dengan cara produksi yang mudah dan singkat. Selain itu, 3D printing menawarkan banyak pilihan bahan baku dan kemampuan mencetak model yang rumit sehingga semakin populer dan menjadi tren saat ini. [6].

Dalam proses 3D printing, ketidakseimbangan dalam aliran bahan sering terjadi karena kecepatan gerakan dan ekstrusi yang tidak optimal. Untuk mengatasi masalah ini, diperlukan penentuan parameter kecepatan gerakan dan ekstrusi yang optimal, dimana dimensi cetakan akan meningkat seiring dengan peningkatan kecepatan ekstrusi. Setiap tingkat kecepatan gerakan memerlukan kecepatan ekstrusi yang optimal untuk menghasilkan hasil cetakan yang baik. Oleh karena itu, 3D printer membutuhkan ekstruder yang berkecepatan tinggi agar dapat mendorong bahan cair ke nozzle dengan cepat. Motor stepper yang berbasis Bigtreetech pada kontrol *high speed* ekstruder akan digunakan dalam untuk meningkatkan kecepatan cetakan dan efisiensi waktu pada mesin 3D printer berukuran 2x2x2 meter. Tujuannya adalah untuk mencapai kecepatan pencetakan hingga mendekati 25cm/s.

## 2. METODE PENELITIAN

### 2.1 Firmware dan G-Code

Penelitian ini bertujuan untuk mengembangkan firmware pada mesin 3D printer berukuran besar dengan konfigurasi simetris bilateral 2x2x2 meter, dilengkapi dengan motor servo BLDC (*Brushless Direct Current*) sebagai penggerak ekstruder. Firmware ini menggunakan metode *Close Loop Interrupt Speed* guna memastikan akurasi dan kecepatan gerakan ekstruder melalui pembacaan G-Code.

Tahapan penelitian meliputi:

#### a. Desain Mekanik dan Sistem Kontrol 3D Printer:

Pada tahap awal, desain mekanik dan sistem kontrol printer 3D dikembangkan dengan struktur simetris bilateral. Struktur ini menjaga stabilitas dan keseimbangan pergerakan pada sumbu X, Y, dan Z saat mencetak objek berukuran besar dan kompleks. Konfigurasi ini memastikan keandalan dalam pencetakan pada area sebesar 2x2x2 meter.

#### b. Pengembangan Firmware dan Integrasi Motor Servo BLDC:

Firmware dikembangkan untuk mengintegrasikan motor servo BLDC dengan sistem kontrol berbasis *close loop* yang menggunakan umpan balik dari sensor. Sistem ini membaca G-Code untuk mengatur gerakan ekstruder secara real-time, memastikan akurasi dan kecepatan pencetakan sesuai parameter. Pengendalian berbasis *close loop* memungkinkan mesin untuk mendeteksi perubahan kecil pada kecepatan atau posisi dan menyesuaikan secara instan melalui *interrupt*, yang meningkatkan presisi cetakan dan mengurangi kesalahan.

#### c. Pengujian Kinerja Firmware:

Pada tahap pengujian, kinerja firmware diuji dengan berbagai jenis material dan variasi kecepatan pencetakan untuk memastikan kompatibilitas dan stabilitas pada skala besar. Parameter seperti kecepatan, akurasi, konsumsi daya, dan daya tahan diuji untuk mengevaluasi efektivitas firmware dan metode *close loop* dalam membaca dan menerjemahkan G-Code bagi pergerakan ekstruder.

Resti Dyah Ayu Retno Palupi : Kontrol *High Speed Extruder*...

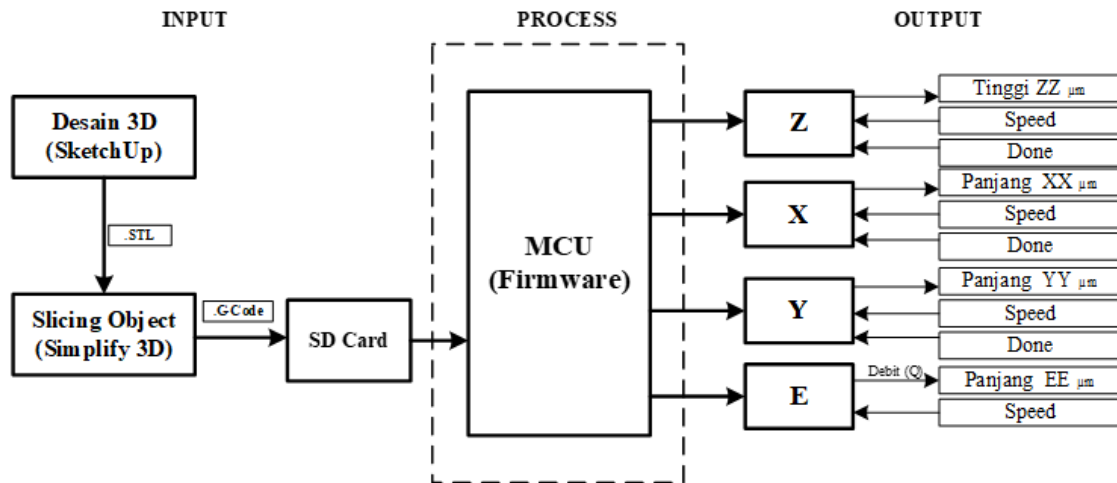
p-ISSN: 2356-0533; e-ISSN: 2355-9195



Hasil pengujian dianalisis untuk menilai efektivitas motor servo BLDC dalam mengontrol kecepatan dan akurasi gerakan ekstruder pada volume cetak besar, serta untuk mengidentifikasi peluang peningkatan agar firmware lebih responsif dan efisien dalam pencetakan skala besar.

### 2.2 Diagram Blok Sistem

Diagram blok system berisi masukan, proses, dan keluaran pada saat mengontrol high speed extruder pada 3D Printer 2x2x2 meter berbasis MCU Bigtree dapat dilihat pada pada Gambar 2.



Gambar 1. Blok Diagram

Input dalam printer 3D dikelola melalui G-Code, yang bertindak sebagai program firmware. Kode ini mengarahkan gerakan motor selama pencetakan, menentukan lokasi, jalur, dan kecepatan. G-Code diproses dalam MCU, papan mikrokontroler yang dirancang untuk pencetakan 3D, menggunakan firmware. Firmware dikonfigurasi dengan kode program untuk mengatur kecepatan motor dan diunggah ke MCU melalui komputer. MCU memproses input G-code, mengatur perintah gerakan dalam antrian untuk dieksekusi. Perintah ini mengendalikan motor stepper di sumbu Y (alas), sumbu X (gerakan horizontal ekstruder), dan sumbu Z (gerakan vertikal ekstruder). Outputnya adalah kecepatan gerakan di motor yang menggerakkan sumbu pencetakan 3D, yang memfasilitasi prosedur pencetakan.

### 2.3 Prinsip Kerja

Proses pencetakan objek 3D dimulai dengan mendesain objek menggunakan SketchUp dan menyimpannya dalam format STL. Desain tersebut kemudian di-slice menggunakan Simplify3D, di mana parameter kecepatan dapat disesuaikan. Setelah slicing, file STL diubah menjadi G-code yang berisi irisan berlapis. G-code diunggah ke MCU Bigtree dan disimpan di SD card yang terpasang pada LCD display. Firmware Marlin-bugfix-2.0.x dikonfigurasi sesuai spesifikasi printer dan di-upload ke MCU menggunakan Visual Studio Code. G-code yang diunggah mengarahkan pergerakan motor pada sumbu X, Y, Z, serta motor stepper E untuk mengekstrusi filament, memulai proses pencetakan.

### 2.4 Perancangan Firmware untuk Pembacaan G-Code

Perancangan firmware dimulai dengan integrasi G-Code sebagai input untuk mengendalikan gerakan motor pada sumbu X, Y, dan Z pada mesin 3D printer. Firmware ini dikembangkan dengan metode *close loop* untuk



menjaga akurasi melalui umpan balik sensor yang mengatur kecepatan dan posisi motor secara real-time. Setiap perintah dari G-Code diterjemahkan oleh MCU, mengatur antrian gerakan motor agar sesuai dengan urutan dan kecepatan yang dibutuhkan dalam proses pencetakan. Berikut adalah gambar potongan dari G-Code.

```
G90
M82
M106 S0
M140 S65
M190 S65
G28 ; home all axes
; process Process1
; layer 1, Z = 0.150
T0
G92 E0.0000
G1 E-1.0000 F1200
; feature skirt
; tool H0.360 W0.480
G1 Z0.150 F7500
G1 X722.751 Y16.476 F15000
G1 E0.0000 F1200
G92 E0.0000
G1 X727.515 Y11.711 E0.3389 F12000
G1 X737.515 Y11.711 E0.8418
G1 X739.515 Y9.711 E0.9840
G1 X749.515 Y9.711 E1.4869
G1 X751.515 Y7.711 E1.6291
G1 X773.866 Y7.711 E2.7531
G1 X775.866 Y9.711 E2.8954
G1 X789.866 Y9.711 E3.5994
G1 X792.631 Y12.476 E3.7961
G1 X792.631 Y22.826 E4.3166
G1 X789.866 Y25.591 E4.5132
G1 X787.866 Y25.591 E4.6138
```

Gambar 2 G-Code

Semua proses dalam mesin cetak 3D akan dikontrol melalui firmware dengan menggunakan metode kecepatan interupsi loop tertutup. Secara umum, dalam pengembangan firmware untuk printer 3D, penggunaan sistem kontrol umpan balik dapat membantu meningkatkan akurasi dan kinerja cetak. Interupsi, dalam konteks mikrokontroler, dapat digunakan untuk menangani kejadian eksternal yang memerlukan respons cepat, seperti input sensor atau kondisi kesalahan, di mana proses dimulai dengan membaca G-Code dari hasil pemotongan untuk menentukan proses yang dieksekusi.

```
void GCodeReader::readGCode(const char* filePath) {
    File file = SD.open(filePath);
    if (!file) {
        Serial.println("Error: Could not open file.");
        return;
    }

    while (file.available() && commandCount < 100) { // Limit to 100 commands for simplicity
        String line = file.readStringUntil('\n');
        line.trim(); // Remove whitespace

        // Ignore comments and empty lines
        if (line.length() == 0 || line.startsWith(";")) {
            continue;
        }

        GCodeCommand command;
        command.command = line.substring(0, line.indexOf(' '));

        // Parse the line for X, Y, Z values
        int xIndex = line.indexOf('X');
        if (xIndex != -1) {
            command.x = line.substring(xIndex + 1).toDouble();
        }

        int yIndex = line.indexOf('Y');
```

Gambar 3 Pemrograman pada Firmware

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Hasil Pengujian G-Code untuk Nilai RPM

Resti Dyah Ayu Retno Palupi : Kontrol *High Speed Extruder*...

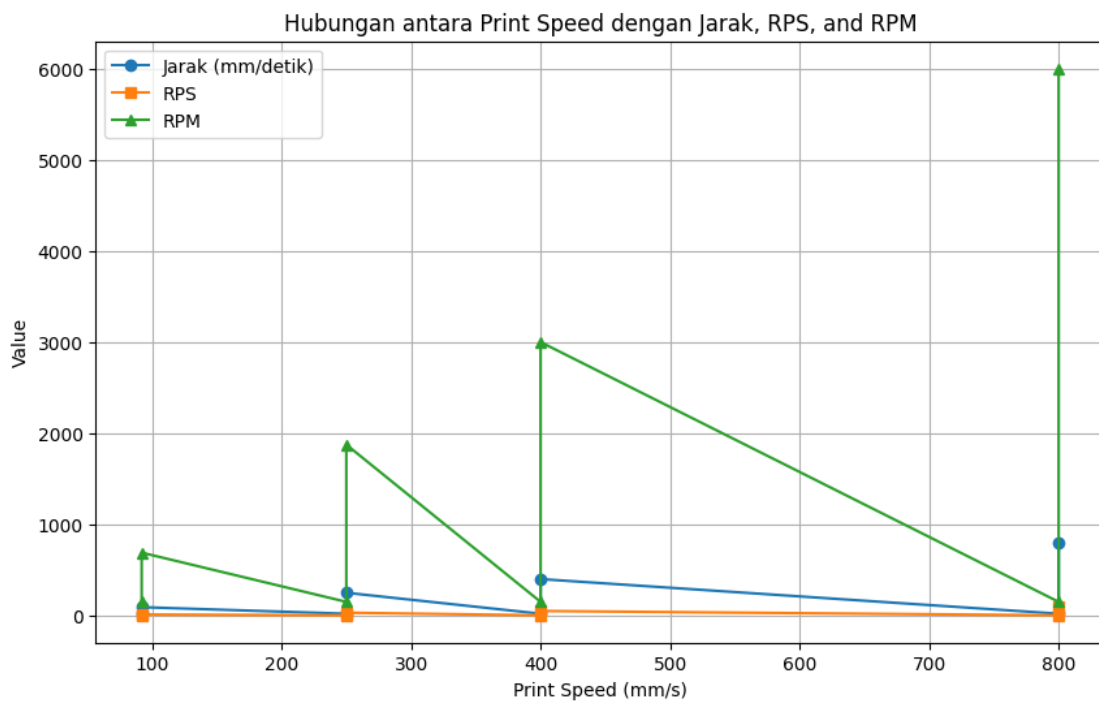
p-ISSN: 2356-0533; e-ISSN: 2355-9195



Pengujian ini dilakukan untuk mengetahui pengaruh penggunaan G-Code terhadap nilai RPM (Rotasi per Menit) dalam proses cetak 3D. Melalui berbagai skenario kecepatan cetak (Print Speed) yang diterapkan, diukur bagaimana variasi G-Code berkontribusi terhadap perubahan nilai RPM. Data yang diperoleh menunjukkan hubungan yang erat antara nilai G-Code dengan peningkatan kecepatan rotasi mesin, yang pada akhirnya mempengaruhi performa cetakan. Hasil pengujian ini dirangkum sebagai berikut:

Tabel 1. Hubungan Print Speed dengan arak, RPS, dan RPM

Print Speed (mm/s)	G-Code	Jarak (mm/menit)	Jarak (mm/detik)	Lead (mm)	RPS	RPM
92	F1200	1200	20	8	2.5	150
	F5520	5520	92	8	11.5	690
250	F1200	1200	20	8	2.5	150
	F15000	15000	250	8	31.25	1875
400	F1200	1200	20	8	2.5	150
	F24000	24000	400	8	50	3000
800	F1200	1200	20	8	2.5	150
	F48000	48000	800	8	100	6000



Gambar 4 Grafik Hubungan antara Print Speed dengan Jarak, RPS, dan RPM



Data tersebut menunjukkan hubungan antara kecepatan cetak (*Print Speed*), G-code, jarak per menit, jarak per detik, *lead* (langkah ulir), rotasi per detik (RPS), dan rotasi per menit (RPM). Berikut adalah analisis dari data tersebut:

1. Hubungan antara Print Speed, Jarak, dan RPM:
  - 1) Print Speed (mm/s) dan Jarak (mm/detik) berbanding lurus. Ketika Print Speed meningkat, Jarak per detik (mm/detik) juga meningkat. Contohnya, pada kecepatan cetak 250 mm/s, Jarak per detik adalah 250 mm/detik.
  - 2) Print Speed juga mempengaruhi nilai RPS (Rotasi Per Detik) dan RPM (Rotasi Per Menit). Semakin tinggi Print Speed, semakin tinggi pula RPS dan RPM. Misalnya, untuk kecepatan cetak 400 mm/s, RPM mencapai 3000, sedangkan pada 800 mm/s, RPM menjadi 6000.
2. Penggunaan G-Code dan Perubahan Kecepatan:
  - 1) G-Code F1200 konstan di setiap kasus pada kecepatan yang rendah, yaitu 20 mm/detik dengan Jarak per menit sebesar 1200 mm/menit.
  - 2) Pada nilai G-code yang lebih tinggi, seperti F5520, F15000, F24000, dan F48000, kecepatannya meningkat drastis, sesuai dengan Print Speed yang tertera.
    - G-Code F5520 menghasilkan kecepatan 92 mm/detik.
    - G-Code F15000 menghasilkan kecepatan 250 mm/detik.
    - G-Code F24000 menghasilkan kecepatan 400 mm/detik.
    - G-Code F48000 menghasilkan kecepatan 800 mm/detik.
3. Konsistensi Lead (Langkah Ulir) dan Pengaruh pada RPS:
  - 1) Lead atau langkah ulir tetap konstan pada nilai 8 mm untuk semua data. Meskipun lead tetap, RPS bervariasi sesuai dengan kecepatan cetak. Misalnya:
  - 2) Pada kecepatan cetak 92 mm/s, RPS adalah 11.5. Pada kecepatan cetak 800 mm/s, RPS naik menjadi 100.
4. RPM (Rotasi per Menit):
  - 1) RPM sangat bervariasi, mulai dari 150 RPM pada kecepatan cetak 92 mm/s hingga mencapai 6000 RPM pada kecepatan cetak 800 mm/s. Hal ini menunjukkan bahwa mesin harus berputar lebih cepat untuk menghasilkan kecepatan cetak yang lebih tinggi.

### 3.2 Hasil Pengujian Software

Pengujian ini bertujuan untuk mengetahui pengaruh perubahan parameter di simplify 3d yang berpengaruh pada G-Code.

Tabel 1. Data Simplify 3D dan G-Code

Pengaturan Posisi Objek (mm)			G-Code (mm)		
X	Y	Z	X	Y	Z
300	50	2	336	43	2
Pengaturan Speed (mm/s)			G-Code (mm/menit)		
X	Y	Z	X	Y	Z
50	50	50	3000	3000	3000

Dari tabel yang disajikan, terdapat dua pengaturan penting terkait posisi objek dan kecepatan. Berikut adalah analisis dari masing-masing bagian:

1. Pengaturan Posisi Objek (mm):
  - 1) Tabel ini menunjukkan perbandingan antara **posisi objek** (X, Y, Z) yang diatur secara manual dengan nilai posisi yang dihasilkan oleh **G-Code**.
    - a. Untuk sumbu **X**, posisi yang diatur adalah **300 mm**, sementara hasil G-Code menunjukkan posisi sebesar



**336 mm**. Ini berarti terdapat perbedaan **36 mm**, yang kemungkinan disebabkan oleh faktor-faktor seperti offset atau pengaturan tambahan dalam sistem G-Code.

- b. Untuk sumbu **Y**, posisi yang diatur adalah **50 mm**, sedangkan G-Code menunjukkan nilai **43 mm**. Terdapat perbedaan **7 mm**, yang bisa disebabkan oleh koreksi atau pergerakan dari alat cetak.
- c. Untuk sumbu **Z**, tidak ada perbedaan. Baik nilai yang diatur maupun nilai G-Code tetap **2 mm**, menunjukkan bahwa pergerakan pada sumbu ini konsisten dan sesuai dengan pengaturan.

2. Pengaturan Speed (mm/s):

Bagian ini menunjukkan perbandingan antara kecepatan cetak yang diatur secara manual dengan kecepatan yang dihasilkan oleh G-Code.

- a. Untuk semua sumbu (X, Y, Z), kecepatan yang diatur adalah **50 mm/s**.
- b. Namun, nilai G-Code mengkonversi kecepatan tersebut ke dalam **mm/menit**, yang menjadi **3000 mm/menit** ( $50 \text{ mm/s} \times 60 \text{ detik} = 3000 \text{ mm/menit}$ ). Hal ini menunjukkan konversi standar yang dilakukan oleh sistem G-Code, karena kecepatan sering dinyatakan dalam mm/menit dalam proses pengkodean CNC dan printer 3D.

### 3.3 Hasil Pengujian Jarak pada Sumbu X (Extruder)

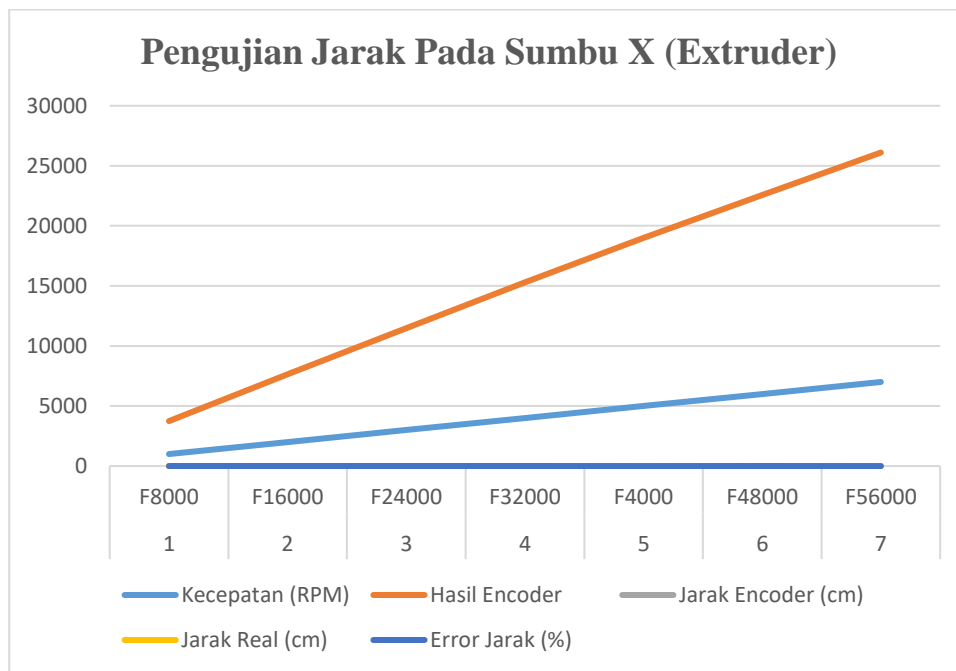
Hasil pengujian jarak linier pada sumbu X terhadap jarak perhitungan encoder dapat dilihat pada table 1.3 berikut.

Tabel 3. Pengujian Jarak pada Sumbu X (Extruder)

No.	G-Code	Kecepatan (RPM)	Hasil Encoder	Jarak Encoder (cm)	Jarak Real (cm)	Error Jarak (%)
1	F8000	1000	3744	5,84	5,7	2,45
2	F16000	2000	7634	11,93	11,8	1,10
3	F24000	3000	11494	17,94	17,6	1,93
4	F32000	4000	15304	23,90	23,5	1,70
5	F4000	5000	19010	29,71	29,6	0,37
6	F48000	6000	22584	35,29	35,1	0,54
7	F56000	7000	26104	40,79	40,7	0,22
Rata – Rata Error (%)						1,13

Tabel di atas menunjukkan hasil pengujian dengan menggunakan G-code pada berbagai kecepatan (RPM). Hasil encoder, jarak encoder yang dihitung, dan jarak yang sebenarnya (*real distance*) dicatat pada setiap pengujian dan seperti grafik berikut.





Gambar 5 Pengujian Jarak Pada Sumbu X (Extruder)

Pada pengujian ini menggunakan beban dengan berat 2667 gram yang mana beban tersebut diletakkan pada bagian extruder. Kemudian ulir yang dipasang pada extruder memiliki lead 8mm. Encoder yang terpasang pada poros motor telah di atur dengan resolution 512 yang artinya bahwa 1 putaran bernilai 512 tik atau pulse. Sehingga rumus untuk menghitung nilai encoder menjadi jarak adalah sebagai berikut.

$$\text{Jarak encoder (Len CM)} = \left( \left( \frac{\text{hasil nilai encoder}}{512} \right) \times 8 \right) / 10 \quad (1)$$

Kemudian untuk mengetahui presentase error dapat menggunakan rumus.

$$\text{Presentase Error} = \left| \frac{\text{Jarak real} - \text{Jarak encoder}}{\text{Jarak real}} \right| \times 100\% \quad (2)$$

$$\text{Rata - rata Error} = \frac{\sum \text{Presentase Error (\%)}}{\text{Jumlah Data}} \times 100\% \quad (3)$$

Berdasarkan data yang disajikan dalam Tabel 1.2, pengujian jarak pada sumbu X menunjukkan hubungan antara kecepatan (RPM) dan error jarak antara hasil pembacaan encoder dan jarak yang sebenarnya (real distance). Hasil pengujian menunjukkan bahwa:

- a. **Error Jarak Menurun dengan Peningkatan Kecepatan:** Secara umum, error jarak cenderung menurun seiring dengan peningkatan kecepatan (RPM). Pada kecepatan rendah (1000 RPM), error mencapai 2,45%, sedangkan pada kecepatan tinggi (7000 RPM), error hanya 0,22%. Ini menunjukkan bahwa dengan peningkatan kecepatan, akurasi pembacaan encoder semakin baik.
- b. **Fluktuasi Error pada Kecepatan Menengah:** Meskipun ada penurunan umum dalam error, pada kecepatan tertentu, seperti pada 3000 RPM (error 1,93%) dan 4000 RPM (error 1,70%), terdapat fluktuasi kecil dalam nilai error. Hal ini mungkin disebabkan oleh faktor-faktor seperti pengaturan parameter yang tidak optimal pada proses slicing atau perbedaan karakteristik motor dan sistem encoder pada kecepatan menengah.





- c. **Error Jarak Rata-Rata:** Rata-rata error pada semua pengujian adalah **1,13%**, yang menunjukkan bahwa secara keseluruhan, akurasi sistem encoder cukup baik, dengan error yang relatif rendah meskipun pada variasi kecepatan yang berbeda.

#### 4. KESIMPULAN

Firmware khusus untuk kontrol motor servo BLDC pada 3D printer berukuran besar dengan struktur simetris bilateral 2x2x2 meter, yang mampu meningkatkan efisiensi dan kualitas cetakan melalui metode Close-Loop Interrupt Speed. Metode ini memungkinkan penyesuaian kecepatan dan posisi ekstruder secara real-time menggunakan umpan balik sensor, menghasilkan cetakan yang lebih presisi dan konsisten. Hasil pengujian menunjukkan bahwa peningkatan kecepatan cetak (print speed) berbanding lurus dengan peningkatan RPM, yang mengurangi error jarak hingga rata-rata 1,13%. Pengaturan G-Code dengan variasi kecepatan serta parameter lain, seperti jerk dan slicing, juga berpengaruh pada hasil cetakan dengan mengurangi kesalahan dimensi dan stringing. Kesimpulannya, firmware ini berhasil meningkatkan akurasi dan stabilitas dalam pencetakan 3D skala besar, dengan keandalan sistem encoder yang memungkinkan pengendalian ekstruder yang akurat pada berbagai kecepatan cetak.

#### 5. UCAPAN TERIMAKASIH

Ucapan terima kasih yang mendalam disampaikan kepada **BIMA | Kemendikbudristek 2024** atas dukungan finansial yang diberikan pada **Penelitian Dasar – Penelitian Tesis Magister**. Keberhasilan setiap tahap penelitian tidak terlepas dari dukungan keuangan yang memadai. Bantuan yang diberikan tidak hanya memfasilitasi proses eksperimen dan pengumpulan data, tetapi juga menciptakan kondisi yang memungkinkan fokus penuh pada aspek intelektual dan metodologis penelitian.

#### DAFTAR PUSTAKA

- [1] Jinyang Huang, Fang Yamin. Design of CNC Feed Servo System Based on Fuzzy PID. 2021.
- [2] Awalia Anief, Sumbodo Wirawan. Perancangan 3D Printer Tipe Core XY Berbasis Fused Deposition Modeling (FDM) Menggunakan Software Autodesk Inventor 2015. Fakultas Teknik Universitas Negeri Semarang. Semarang. 2018.
- [3] Web Dinas Kementrian Perindustrian [Internet]. [Diakses 15 September 2023]. Available from: [Kemenperin.go.id](http://Kemenperin.go.id)
- [4] Bukhari SBH, Tanveer T, Abid A, Anwar S. Design and Fabrication of Inexpensive Portable Polar 3D Printer. 2023.
- [5] Maravi DA, Iparraguirre GM, Prado SR. Implementation of a Digital PID Control for the Compensation of Loss Steps from CORE XY 3D Printer Motors Working at High Speeds. IEEE. 2020.
- [6] Konstantyan VN, Sannikov AS, Atabaeva DM, Morozov AI. Highly Integrated Compact Motion Control System for Brushless DC Machine. In: 2020 IEEE International Conference on Industrial Technology (ICIT). IEEE; 2020. p. 327-329.
- [7] C. Qing, L. Zhongjin, and W. Mengtao, VNDS: An algorithm for cross platform vulnerability searching in binary firmware, J. Comput. Res. Develop., vol. 53, no. 10, pp. 22882298, 2016.
- [8] D.D.Chen,M.Woo,andD.Brumley, Towardsautomateddynamicanalysis for linux-based embedded firmware, in Proc. NDSS. 2016, pp. 116.
- [9] J. Zaddach, L. Bruno, and A. Francillon, AVATAR: A framework to support dynamic security analysis of embedded systems Firmwares, in Proc. NDSS, 2014, pp. 116.
- [10] M. Muench, D. Nisi, and A. Francillon, Avatar 2: A multi-target orchestration platform, in Proc. Workshop Binary Anal. Res. (Colocated NDSS Symp.), Feb. 2018, Art. no. 18.
- [11] A. Costin, A. Zarras, A. Francillon, Automated dynamic firmware analysis at scale: A case study on embedded Web interfaces, in Proc. 11th ACMAsia Conf. Comput. Commun. Secur., 2016, pp. 437448.
- [12] S.H.H.Ding,B.C.M.Fung,andP.Charland, Asm2Vec:Boosting static representation robustness for binary clone search against code obfuscation and compiler optimization, in Proc. IEEE Symp. Secur. Privacy (SP), May 2019, pp. 472489.
- [13] A. Costin, J. Zaddach, and A. Francillon, A large-scale analysis of the security of embedded firmwares, in Proc. Usenix Secur. Symp., 2014, pp. 95110.
- [14] J. Pewny, B. Garmany, and R. Gawlik, Cross-architecture bug search in binary executables, in Proc. IEEE Symp. Secur. Privacy, May 2015, pp. 709724.
- [15] Y. David, N. Partush, and E. Yahav, Similarity of binaries through re-optimization, in Proc. 38th ACM SIGPLAN Conf. Program. Lang. Design Implement., 2017, pp. 7994.



- [16] Q. Feng, R. Zhou, and C. Xu, Scalable graph-based bug search for rmware images, in Proc. ACM SIGSAC Conf. Comput. Commun. Secur., 2016, pp. 480491.
- [17] D. Zhao, H. Lin, and L. Ran, CVSkSA: Cross-architecture vulnerability search in rmwarebasedonkNN-SVMandattributedcontrol owgraph, Softw. Qual. J., vol. 27, pp. 124, Feb. 2019.

