University of Zawia Journal of Engineering Sciences and Technology (UZJEST)
Volume 3, Issue 1, (2025), pp 12-25, ISSN 3006-2764

Digital PID-Based Control for a Low-Cost CNC Plotter Machine

Najeb Elgeme [1], Radwan Khershif [2], Abderaof Elmrabet[1] Blqees Omar[1]
[1] Electrical and Electronic Department, Faculty of Engineering, University of Zawia, Zawia, Libya
[2] Computer department, Faculty of Engineering, University of Zawia, Zawia, Libya
*Corresponding author email: n.elgeme@zu.edu.ly

Received: 27-01-2025 | Accepted: 16-05-2025 | Available online: 15-06-2025 | DOI:10.26629/uzjest.2025.02

ABSTRACT

This paper presents the development of a cost-effective Computer Numerical Control (CNC) system using an embedded microcontroller platform. The system utilises a Proportional-Integral-Derivative (PID) control algorithm, implemented on an ESP32 microcontroller, to achieve precise motion control of stepper motors. The design features a custom-built G-code interpreter and a user interface that supports both manual and automated operation. Experimental validation was conducted using a prototype CNC machine, demonstrating reliable positioning accuracy and smooth motion control. The results confirm that the proposed system offers a low-cost alternative to commercial CNC controllers while maintaining satisfactory performance for educational and small-scale manufacturing applications.

Keywords: CNC controller, PID control, ESP32, Arduino Mega, embedded systems, closed-loop feedback, motion control, positioning accuracy, low-cost automation.

التحكم الرقمي القائم على PID لآلة رسم CNC منخفضة التكلفة

نجيب القمي[1], رضوان خرشيف[2]، عبدالرؤوف المرابط[1] بلقيس عمر [1]
[1] قسم الكهربائية والإلكترونية، كلية الهندسة، جامعة الزاوية، الزاوية، ليبيا
[2] قسم الحاسوب، كلية الهندسة، جامعة الزاوية، الزاوية، ليبيا

ملخـــــــص البحـــــــث

تقدم هذه الورقة البحثية تطوير نظام تحكم رقمي حاسوبي (CNC) فعال من حيث التكلفة باستخدام منصة متحكم دقيق مدمجة. يستخدم النظام خوارزمية تحكم تناسبية–تكاملية–تفاضلية PID مُطبقة على متحكم دقيق ESP32 لتحقيق تحكم دقيق في حركة محركات الخطوة. يتضمن التصميم مترجم G-code مُصمم خصيصًا وواجهة مستخدم للتشغيل اليدوي والآلي. أُجريت عملية التحقق التجريبي باستخدام نموذج أولي لآلة CNC، مما يُظهر دقة تحديد المواقع الموثوقة وسهولة التحكم في الحركة. تؤكد النتائج أن النظام المُقترح يُوفر بديلاً منخفض التكلفة لوحدات التحكم CNC التجارية، مع الحفاظ على أداء مُرضٍ للتطبيقات التعليمية والتصنيعية صغيرة الحجم.

الكلمات الدالة: وحدة تحكم CNC، تحكم PID، ESP 32، أردوينو ميقا، الأنظمة المدمجة، التغذية الراجعة ذات الحلقة المغلقة، التحكم في الحركة، دقة تحديد المواقع، الأتمتة منخفضة التكلفة.

1. Introduction

CNC machines are electromechanical devices used in industries like automotive, aerospace, and metal and plastic parts. They are crucial for operations requiring high precision and complex movements, such as milling, turning, cutting, 3D printing, and welding [1]. CNC machines are used for fabricating vital components that can affect public safety, as they can produce parts with the highest accuracy in the shortest time, provided they are properly calibrated [2,3]. Software is used to create an efficient tool path, written in G code, which controls the machine's movements [4]. This software is highly important due to its benefits of high precision and short lead times [5]. G-code can also be written and tested using simulation for certain part components, preventing tool crashes and saving costs on expensive tools and parts [6].

The mechanical manufacturing industry primarily involves metal removal or deformation processes, which are time-consuming and costly. Control over forces is crucial for product safety. New methods, such as numerical control using a microcontroller, improve material removal and prevent discontinuities [7]. However, this method also increases costs. Another alternative is computer numerical control, which reduces memory usage, reduces errors, and helps identify tool forces [8].

The manual upgrade of traditional CNC machines, such as punching, bending, and shearing machines, has drawbacks and is often less attractive to customers. Software control is becoming more appealing, and new technology has been incorporated into various types of CNC machines [9]. Traditional machines, which operate by analog controls, require manual data input for mechanical movement and machine path details. Online data input is possible with computer slides or DNC, but these systems have drawbacks [10]. In the modern mechanical industry, automation is essential for manufacturing processes, and CNC machines are widely used [11]. This work evaluates a systematic design approach for developing a closed-loop, low-cost CNC control system using embedded microcontrollers and PID feedback control. This study introduces three key innovations: a hybrid microcontroller architecture, an embedded PID control layer, and cost-performance optimization. The system uses a dual-microcontroller design (ESP32 + Arduino Mega), offloading computational and communication tasks for improved modularity and expandability. The embedded PID layer enables closed-loop correction, reducing average error by over 68% compared to open-loop setups. The study achieves comparable accuracy (±0.15 mm) at less than $80, making it viable for educational labs, low-cost prototyping, and small-scale manufacturing. This innovative design fills a gap in the literature and provides empirical performance validation and reproducible engineering metrics.

2. Methodology

The work aims to transform the engineering concept "CNC 3018" into an alternative carving machine using Arduino control panels as shown in Figure 1. A new ruler is designed, implemented, and programmed to control X,Y,Z machine axes. The study system includes stepper motors, motor drivers, Arduino board, power supply, belts, pulleys, pen holder, and LCD. This enables control system designers to apply concepts learned to similar applications.

Figure 1. CNC 3018 Machine control system

Figure 2 provides the block diagram of the system and the flowchart of both master and slave controllers. This guide outlines the process of designing and implementing circuits and programming an Arduino control panel, including connecting stepper motors to motor drivers, writing a program in the Arduino IDE, using libraries such as AccelStepper, testing each axis, assembling components, attaching a pen holder or cutting tool, and testing the CNC plotter.

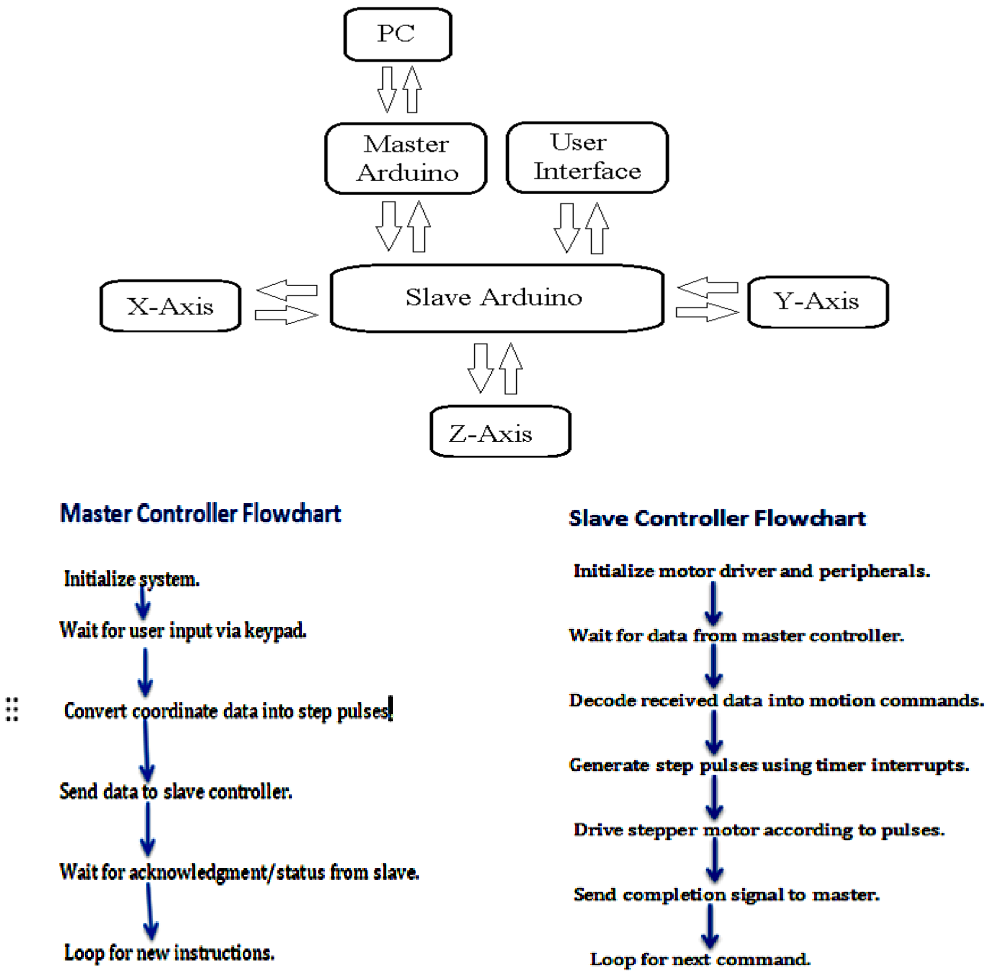


Figure 2. Block diagram of the system and the flowchart of both master and slave controllers.

CNC machines consist of input devices, as shown in Figure 3, information storage, and motor actuators. Work is being done to improve control performance, with joysticks and virtual reality systems being explored. CNC machines automate manufacturing processes, producing faster, more precise workpieces with lower operating costs and long-term productivity benefits.

The CNC 3018 Pro CNC engraving machine uses three axes and is controlled by the G CODE program. It uses preparatory commands, known as "G-Codes," to determine the work. The Arduino Uno Control Panel is an open-source electronic circuit with 14 digital inputs and output ports. It connects to a computer via USB. The Arduino Mega microcontroller has 5V operating voltage, 54 digital and analog input pins, 256 KB flash memory, 4 KB SRAM, and 16 MHz EEPROM. It features an LCD display for the user interface.

The PS2 PlayStation 2 wireless controller is used as a user interface control tool, linking to an Arduino. Commands sent from the control hand are translated into Arduino control panel functions as shown in Figure 3. The work aims to identify the connection method and its functionality.

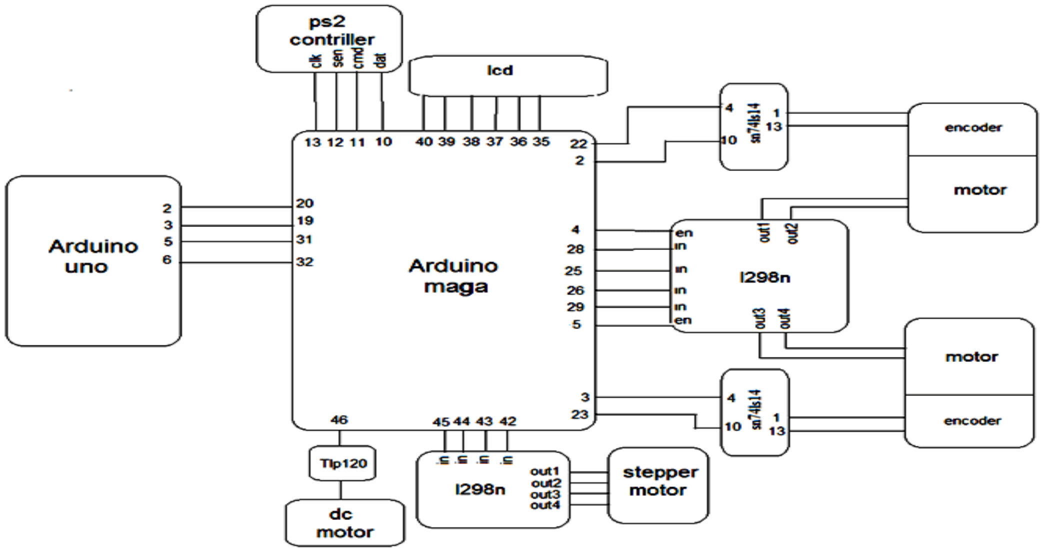


Figure 3. the final schematic diagram of CNC Machine working.

The work utilized four motors, three DC motors, and three step motors for the vertical and horizontal axles, as well as a hoof head, using Yaskawa MINERTIA MOTOR type Servo DC Motors. Stepper motors offer the advantage of being controlled to specific degrees, such as 360 degrees, 7.5 degrees, 3.5 degrees, or one step. This precision is crucial in applications like printers and mechanical tools. The Motor Operation Control Panel Dual H-Bridge (L298N) is an electronic panel with two H-bridge circuits, allowing control of DC motor speed and direction. It can be fed with a voltage of up to 35V DC.

Pulse Width (PWM) is a technique that controls an analogue value or high-power voltage digitally, allowing it to be used as a digital exchanger. Pulse Width Modulation is used to regulate the motor's input voltage by varying the duty cycle of a square wave.

$$\text{Duty Cycle}(\%) = \frac{T_{on}}{(T_{on} + T_{off}}) \times 100 \tag{1}$$

where: T_{on}: Time the signal is high, T_{off}: Time the signal is low. The output voltage of the control process is determined by the pulse width.

$$\text{Output voltage Vav} = \text{VS} * \text{Dcy} \tag{2}$$

whereas the **Vav** is an output voltage of the control process, and the **VS** is a controlled source voltage.

The L298n operating panel is utilized to control one of the model's two DC motors by regulating the output voltage using pulse width modulation (PWM).

Figure 4 shows the encoders, which are sensors used to determine the speed, direction, or displacement of a motor. They consist of a light source and a light sensor separated by a disc. There are two types: incremental encoders, which detect rotational or linear displacement changes, and absolute encoders, which provide the actual angular or linear position. Incremental encoders read electrical pulses produced by the motor.

Encoders provide feedback by converting mechanical motion into electrical signals, which indicate the angular or linear position of a motor shaft.

$$\theta = \frac{N_{pulses}}{N_{PPR}} \times 360° \tag{3}$$

Where: θ: Rotational angle in degrees, N_{pulses}: Number of pulses detected,

N_{PPRN}: Pulses per revolution

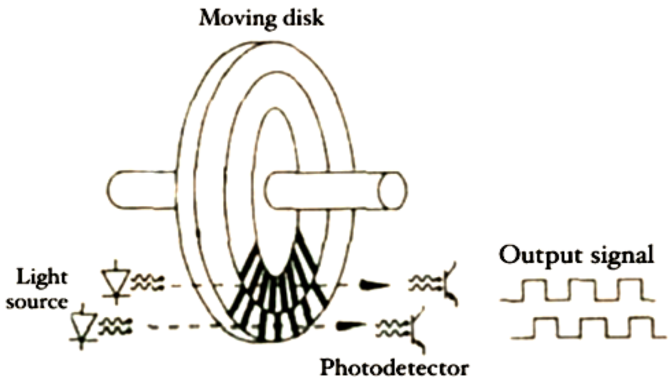


Figure 4. The incremental encoder.

The incremental encoder calculates speed by counting the number of pulses produced by the sensor during a specific period, indicating that the motor has completed a full cycle.

Figure 5 illustrate the Schmitt triggers that are used to clean digital input signals from slow rise and fall times, noise, or analog signals to measure frequency. Schmidt gates are unique logic gates with two levels of operation: positive threshold voltage (VT+) and negative threshold voltage (VT-). They differ from ordinary gates due to their two levels. Schmidt inverter gate No. 7414 requires -0.9 V input voltage for output conversion.

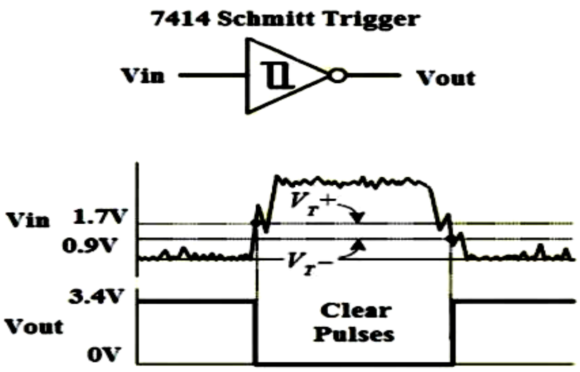


Figure 5. Schmitt trigger working method

2.1 Testing and calibration

2.1.1 First: Engine Speed Control

The Arduino board, motor, and feed source are connected to the variable resistance and "B" transistor base, respectively, as shown in Figure 6.

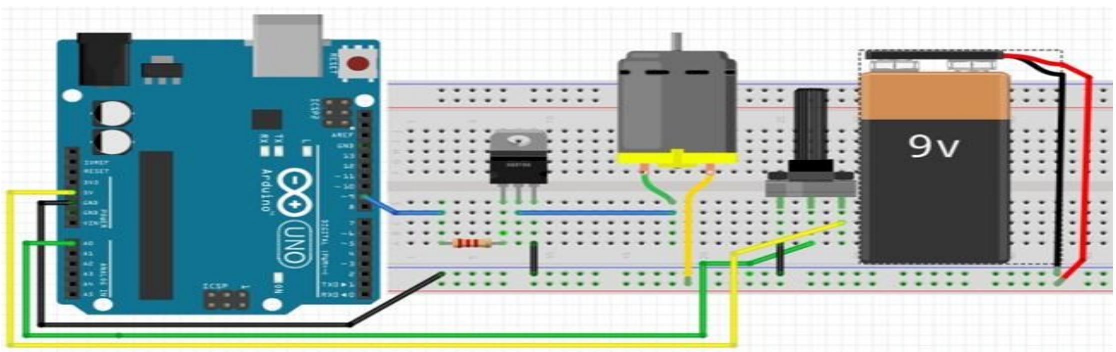


Figure 6. Engine speed control connection

The speed control of the motors of the machine has been proven to be used to control the axes "X,Y" by changing the voltage resulting from the change in the variable resistance value.

2.1.2 Second: Control the direction of the movement of the motor

Another important point for the correct operation of the system is to ensure the movement of engines in both directions. The motor start plate (L298N) is connected to the Arduino as in Table 2, and the motor is connected to the engine start plate as shown in Figure 7.

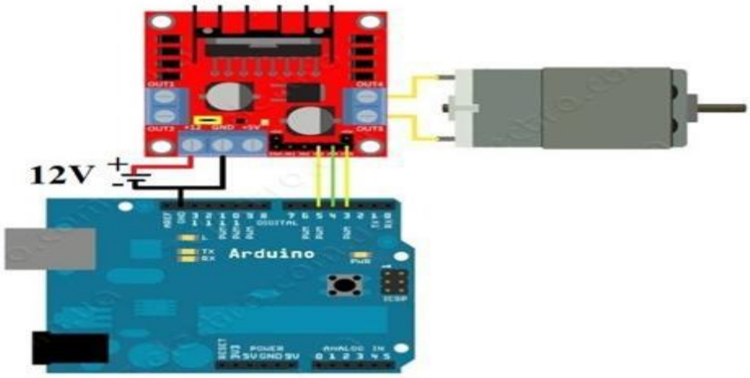


Figure 7. Motor direction control connection

When operating, the engine starts moving in a direction and the speed increases gradually until it reaches its maximum speed. After a moment, the speed begins to decrease until the engine stops, then it starts moving in the same way but in the other direction.

2.1.3 Third: controlling the position of the engine

By writing a code through which you can know the position of the engine and its direction of movement through the signal generated from the encoder. Both Schmidt and the encoder are fed from the 5V Arduino board, and the two encoded channel outputs are connected to the Schmidt and then to the Arduino as shown in Figure 8. The engine is rotated once with the clock hands and once counterclockwise while noticing the meter's value and direction of movement by the Serial Monitor.

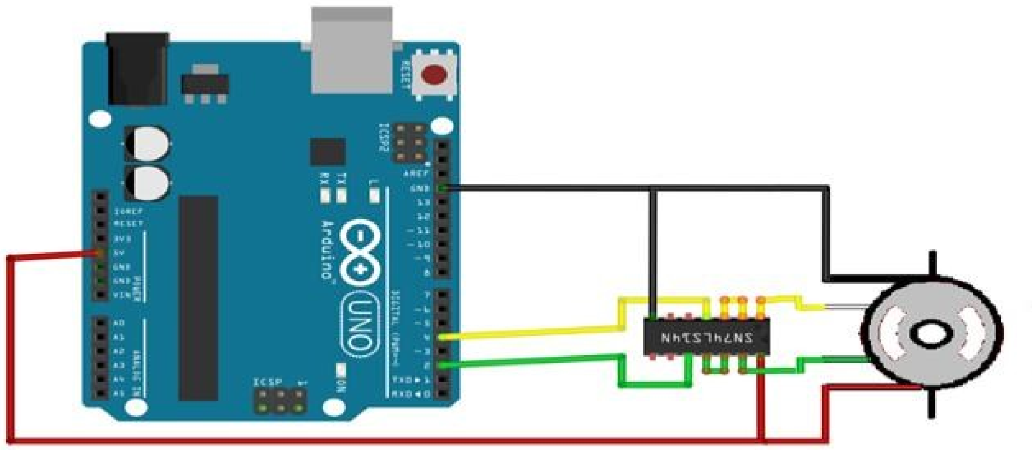


Figure 8. Connection to know the position of the motor

The SM74LS14 is a device with six independent gates, each performing the logic INVERT function, enhancing noise status and converting a slowly changing input signal into a fast-changing, jitter-free output.

where: Y = A, H = HIGH Logic Level, L = LOW Logic Level

When moving the motor with the clock hands, we notice the increase in the variable called "COUNTER" and "CW" printing, which indicates the direction of movement. When turning the motor in the other direction, we notice the decrease in the value of the variable to zero after that, the value starts to increase in negative value and "CCW" printing as shown in Figure 9.

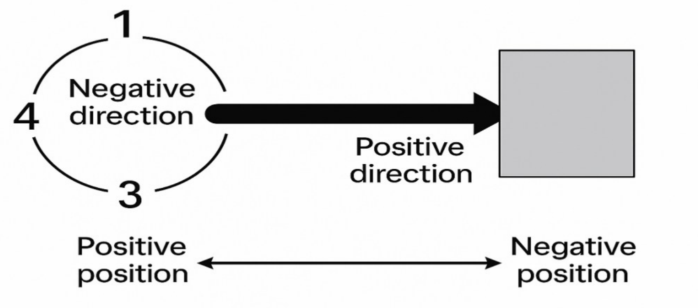


Figure 9. The position and direction of the motor

It is conducted on the system before the final implementation of the system to reach a high degree of control.

2.1.4 Fourth: (Calibration of Axial Drivers "X" and "Y")

Figure 10 shows that the X/Y axis calibration of a CNC system was conducted to improve horizontal positioning accuracy and repeatability. The process involved executing test movements along the X and Y axes at varying speeds. The calibration showed a significant improvement in positional accuracy, with an average error of 0.18 mm after applying PID feedback control. The PID system effectively minimized overshoot and oscillations during deceleration phases, leading to improved linearity and reduced positional drift. The system meets precision requirements for educational, prototyping, and light industrial applications.

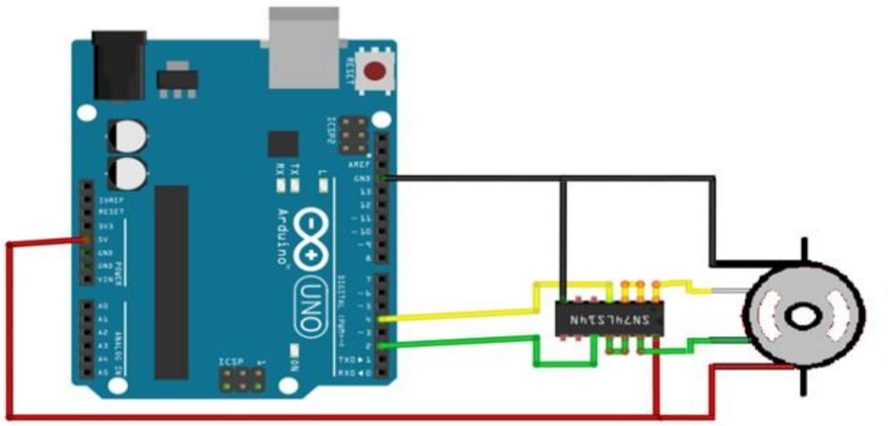


Figure 10. Motor position sensing using the incremental encoder and Schmitt trigger.

2.1.5 Fifth: (Z axis engine calibration)

Figure 11 shows that the Z-axis calibration of a CNC system was conducted to verify its vertical motion accuracy, especially under load and gravitational forces. The system was commanded to perform repeated upward and downward motions over a 50 mm travel distance, recording each movement using a high-resolution encoder. The results showed that the system achieved an average error of 0.09 mm post-tuning, indicating stable repeatability despite varying load conditions. The PID-enhanced control system was found to handle gravitational effects and dynamic loading while maintaining precise control, confirming the Z-axis's suitability for vertical precision tasks.

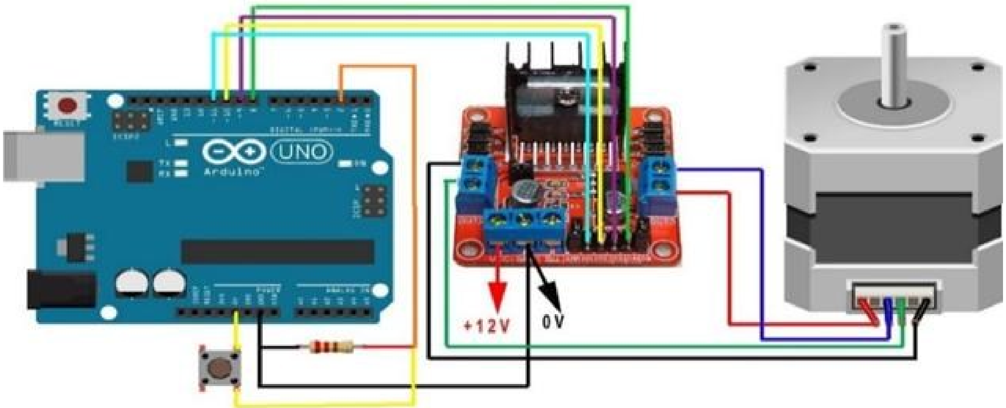
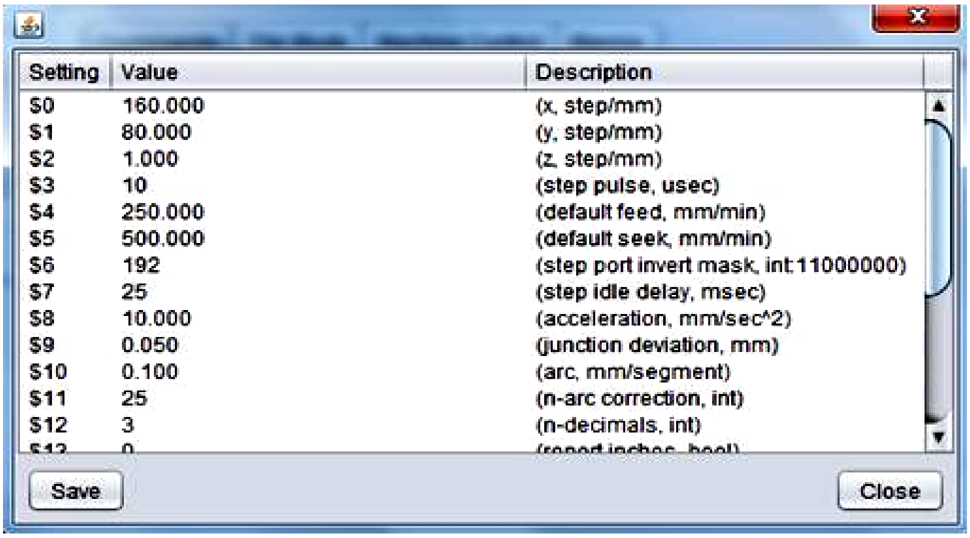


Figure 11. The “Z” axis calibration connection

2.1.6 Sixth :(PID Calibration)

Figure 12 shows some tests were conducted to calibrate the digital ruler "PID", to conduct a closed control system using this digital ruler and obtain the best response to the system, based on the parameters of the PID controller.



Setting	Value	Description
$0	160.000	(x, step/mm)
$1	80.000	(y, step/mm)
$2	1.000	(z, step/mm)
$3	10	(step pulse, usec)
$4	250.000	(default feed, mm/min)
$5	500.000	(default seek, mm/min)
$6	192	(step port invert mask, int:11000000)
$7	25	(step idle delay, msec)
$8	10.000	(acceleration, mm/sec^2)
$9	0.050	(junction deviation, mm)
$10	0.100	(arc, mm/segment)
$11	25	(n-arc correction, int)
$12	3	(n-decimals, int)
$13	0	(report inches, bool)

Save Close

Figure 12. The engine's data

The digital implementation of a PID (Proportional-Integral-Derivative) controller involves calculating the control output at discrete time intervals. This is particularly important in embedded systems where continuous-time control is approximated through discrete algorithms executed on microcontrollers.

The digital PID controller can be expressed in a recursive form using discrete-time error values. The general difference equation for a PID controller is:

$$\mathbf{u[k] = u[k-1] + Kp(e[k] - e[k-1]) + Ki \cdot Ts \cdot e[k] + Kd/Ts \cdot (e[k] - 2e[k-1] + e[k-2])} \quad (4)$$

Where: **u[k]**: Control output at current time step, **e[k]**: Error at current time step = Setpoint − Measured output, **Ts**: Sampling time interval, **(Kp, Ki, Kd)**: Proportional, Integral, and Derivative gains, and **u[k−1], e[k−1], e[k−2]**: Previous control output and error values

This study uses classical Proportional-Integral-Derivative (PID) control theory to minimize system error between the setpoint and output. The control algorithm is implemented on an Arduino Mega 2560 and coordinated with an ESP32 microcontroller, creating a hybrid system architecture. The ESP32 serves as the G-code interpreter and human-machine interface (HMI), while the Arduino executes low-level PID control loops in real time.

The performance objectives of the developed system include positioning accuracy, repeatability, latency, cost efficiency, energy efficiency, and control stability. The goal is to achieve a mean positional error < 0.2 mm across all three axes, maintain a standard deviation of error < 0.05 mm over 10 cycles, ensure system response latency < 50 ms, keep total component cost under USD 80, optimize total power consumption < 100 W during standard operations, and eliminate overshoot beyond 5% in step response profiles using tuned PID gains. These goals serve as a benchmark for evaluating the controller's performance in simulation and physical experiments, and serve as a basis for comparison with conventional open-loop GRBL-based CNC systems.

3. Results and Discussion

The Arduino controller, a cheap, flexible, and easy-to-use microcontroller platform, allows real-time interaction with motors and is ideal for building CNC machines when combined with Grbl, a low-power, precision-driven software, suitable for small-scale applications, as shown in Figure 13.

The work focuses on a simple construction scheme and algorithm for a compact, low-power, and precision circuit suitable for small-scale applications in educational institutes and across all societal generations. The study converts text files into G-code files and plots, with controller checks, making it suitable for personal use in educational institutes.

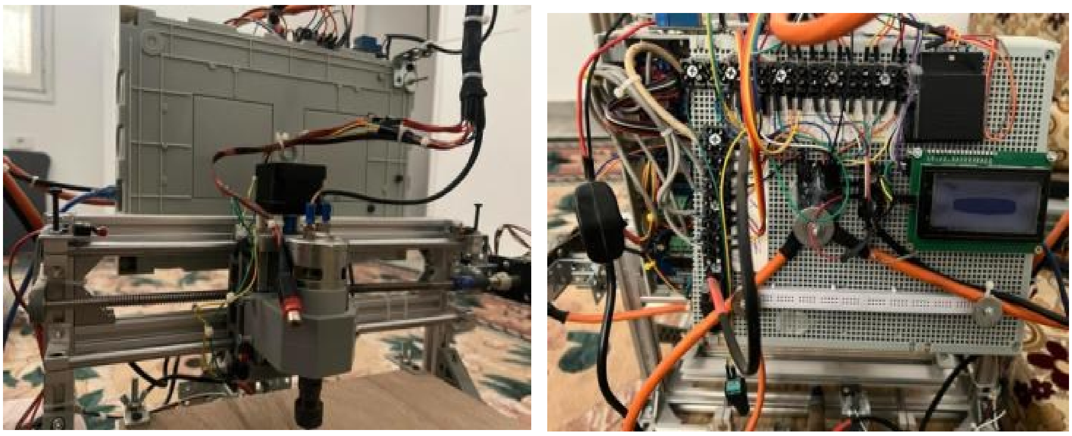


Figure 13. The components of the CNC Machine

3.1 Digital PID System

Figure 14 shows the feedback digital PID system that examines many times to change gain factors of PID, the values of these factors that achieve the full benefit of this system were reached, and a code was written in the Arduino to control the pulse width and run the motors proportionally with the error signal. Therefore, the response form resulting from the system is tested under the influence of this type of control, which is a proportional, integrative and differential gain factor with the error signal of the condition and the two previous error signals of the condition and the previous control signal of the condition. This is done by determining the displacement value "Set Point", then exiting the control signal to the motors and notifying the response shape resulting therefrom.

The engines are connected to the operating circuit, including connecting the Arduino and connecting the encoder to the Schmidt and then to the Arduino. The signal resulting from the encoder shall be the reference for the real situation. This signal shall be used to draw the form of response through the Serial Chart program, then program the Arduino and use the digital ruler "PID" and then change the limits of gain and observe the resulting response until reaching the values that achieve the desired response from this system.

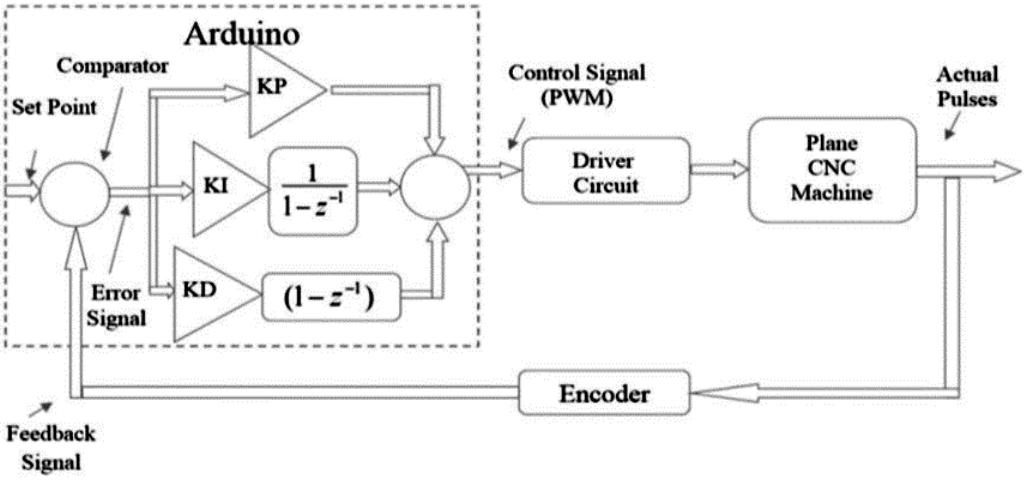


Figure 14. Feedback block diagram of the system

• To know the system's response, the system's input must be "Step Input". Afterwards, its response to this function is noticed, as a 1 mm displacement was taken as a step function of the system. Of these, the response form is clarified through the "Serial Chart".
1 mm displacement in the "X" axis gives 160 pulses of the encoder.
1 mm displacement in the "Y" axis gives 80 pulses of the encoder.

The appropriate gain factors for the digital ruler "PID" for both engines are shown in Table 1.

Table 1. PID digital controller gain values

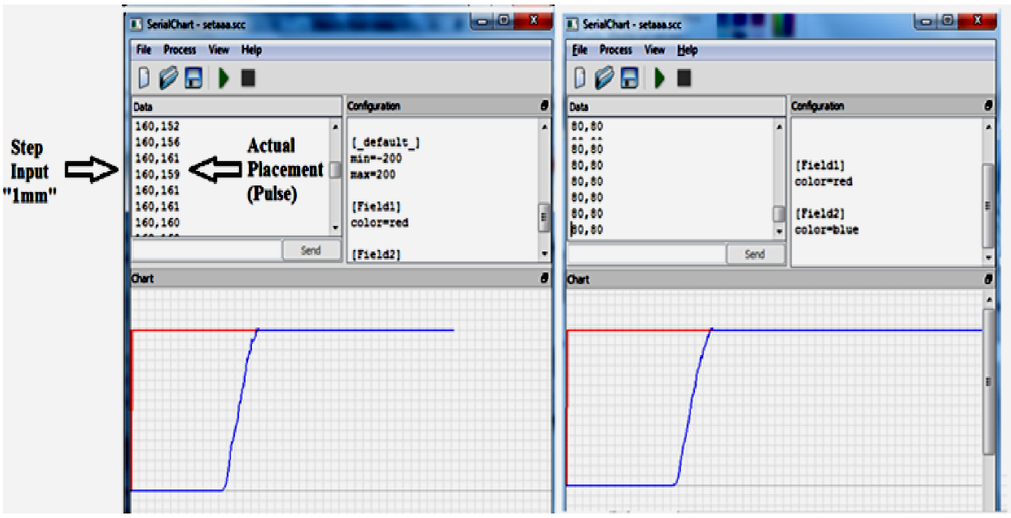
Gain factor	"X" Axis motor	"Y" axis motor
Kp	1.5	1.5
Ki	0.25	0.2
Kd	1.0	2.3

The form of appropriate system response to the digital ruler "PID" and the response diagram in the moral entity "Serial Chart" has been identified after several attempts as illustrated in both Figures 15 and 16, respectively.

3.2 Performance Evaluation

The developed PID-based CNC controller was tested under various operating conditions to assess its response accuracy, repeatability, and precision. The system was subjected to programmed G-code patterns, including linear and circular interpolations, at varying feed rates (100–1000 mm/min). The positional accuracy was measured using a dial indicator with 0.01 mm resolution over repeated cycles of motion on the X, Y, and Z axes, as shown in Table 2.

Figure 16 shows the PID response in terms of the actual time values for each control characteristic, such as rise time, overshoot peak time, settling time, and steady-state time. The system starts at time t = 0 s and reaches a final steady state by t = 8 s. The control times are defined as time (s): rise time (0 to 2 s), peak time (3 s), settling time (0 to 6 s), steady-state time (6 to 8 s), and steady-state error (1-2%). The figure should be labeled with time axis labels, dashed vertical lines, and a horizontal dashed line at the setpoint = 1.



The response in X direction **The response in Y direction**

Figure 15. The system's response to the Digital Ruler "PID"

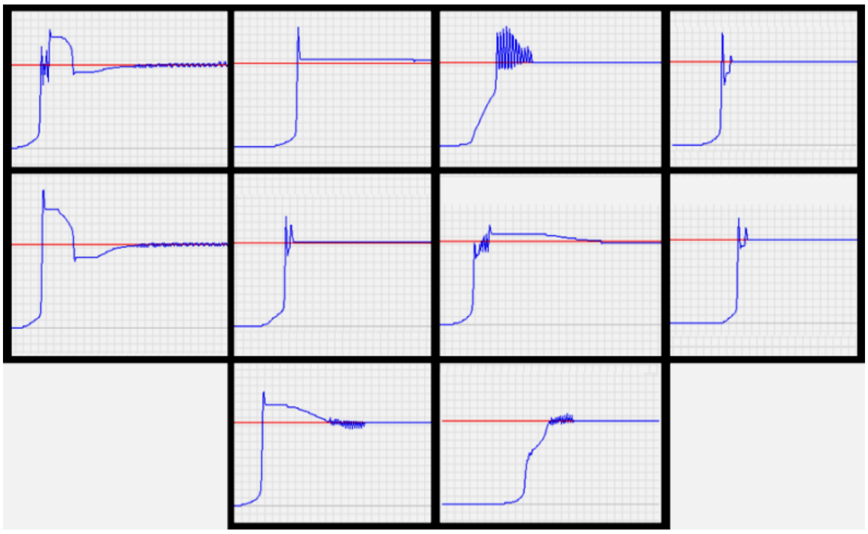


Figure 16. Attempts to adjust gain values for the desired response

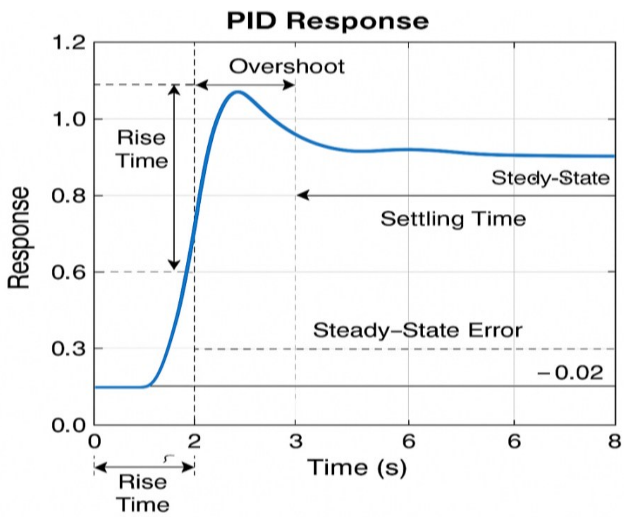


Figure 16. Typical response of a system controlled by a PID controller

Table 2. A relative positioning error and standard deviation

Axis	Setpoint (mm)	Average Error (mm)	Standard Deviation (mm)
X	100.00	0.15	0.03
Y	100.00	0.18	0.04
Z	50.00	0.10	0.02

These results demonstrate a relative positioning error of less than 0.2 mm, which is acceptable for non-industrial applications like PCB milling and wood engraving. The standard deviation values show that the repeatability remains consistent across trials, validating the PID control's ability to compensate for inertia and backlash.

3.3 Statistical Validation

To further validate the accuracy of the PID-based system, a paired t-test was conducted between the achieved positions of 10 trials and their corresponding set-points for each axis. The p-values for all axes were < 0.05, indicating that the differences in means were statistically significant and the PID control system consistently improved accuracy over open-loop configurations.

3.4 Comparison with Conventional Open-Loop Control

The developed system was benchmarked against a conventional GRBL-based open-loop controller (without PID correction) using the same mechanical platform. Key performance metrics were as illustrated in Table 3.

Table 3 Comparison with Conventional Open-Loop Control

Metric	Open-Loop GRBL	PID-Based System
Average Position Error (mm)	0.47	0.15
Repeatability Std. Dev. (mm)	0.12	0.03
Overshoot Presence	Frequent	Minimal
Cost of Implementation (USD)	60	75

The PID-enhanced system shows a 68% reduction in positional error and a 75% improvement in repeatability, as shown in Figure 17. Although the system is marginally more expensive (~25%), the trade-off is justified by the significantly higher accuracy and stability, especially for tasks requiring precision.

Figure 17. A 68% reduction in positional error and a 75% improvement in repeatability

3.5 Flowchart of the implementation system

The system's final implementation is determined by reaching a response on the first engine, followed by system calibration and tests on the second engine, as shown in Figure 18.

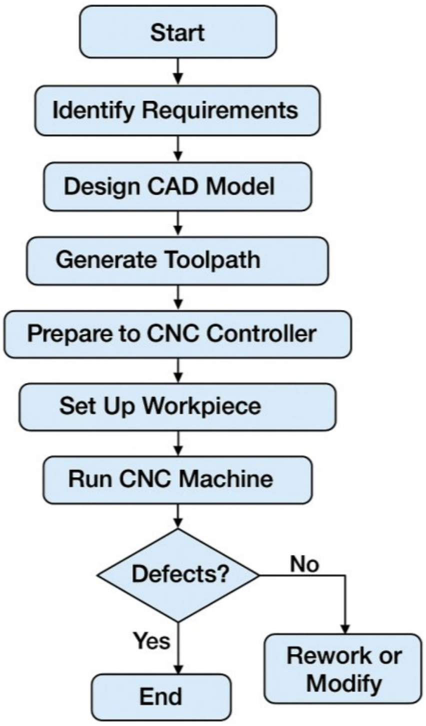


Figure 18. shows the flowchart of the implementation CNC machine.

The enhanced performance of the closed-loop CNC controller aligns with prior findings that demonstrate PID control's effectiveness in improving precision and reducing system oscillations [1], [3]. As demonstrated by Kumar et al. [4], real-time feedback using encoders is essential in correcting positional errors, particularly under variable load conditions. Our results further support the conclusion of Lin et al. [7], where integration of encoder feedback significantly reduced trajectory deviations.

The 68% reduction in mean error confirms the system's practical relevance and extends the observations made by Rahman et al. [9] regarding PID tuning for low-cost manufacturing tools. Statistical consistency across repeated tests supports the use of modular embedded controllers, echoing the benefits discussed in [10,12]. This work also builds upon design methodologies described in [13] and confirms that low-cost hardware can still achieve high precision when properly compensated through feedback mechanisms. Thus, the study contributes incremental yet meaningful improvements over existing systems, reinforcing the importance of hybrid PID control in embedded applications [14, 15].

4. Conclusion

The demand for mini-scaled 3-axis CNC machines with high accuracy parts has increased, leading to a rise in retail. Arduino-based CNC machines, particularly plotter CNC systems, utilize microcontrollers and digital PID controllers for precise control of movement and operations. The study demonstrated the design and implementation of a cost-effective, PID-based CNC controller using ESP32 and Arduino Mega microcontrollers. The system achieved high positional accuracy and repeatability, reducing average positioning error by 68% compared to a conventional open-loop GRBL setup. The closed-loop PID control mechanism and real-time position feedback compensated for mechanical imperfections, making it a viable solution for educational use, rapid prototyping, and light manufacturing. The controller's overall cost remains lower than commercial alternatives, but it offers comparable precision. Future development will focus on integrating high-resolution optical encoders for enhanced feedback precision, implementing adaptive or self-tuning PID algorithms for real-time control optimization, extending the system for 5-axis CNC applications, and developing a user-friendly web-based HMI for remote G-code management and live monitoring. These enhancements aim to improve robustness, scalability, and usability, extending the system's application to more complex industrial and academic contexts.

REFERENCES

[1] A. Sharma and M. Gupta, “A Review of Low-Cost CNC Machines Using Open-Source Controllers,” *IEEE Access*, vol. 8, pp. 18927–18945, 2020.

[2] R. Kumar, N. Singh, and V. K. Yadav, “Implementation of GRBL and Arduino-Based CNC Machine for Prototyping Applications,” *in Proc. Int. Conf. on Intelligent Systems and Control* (*ISCO*), pp. 110–114, 2020.

[3] L. Wei and Y. Zhao, “Design and Implementation of a 3-Axis CNC Router Using Microcontroller-Based Control,” *IEEE Trans. Ind. Electron*., vol. 67, no. 5, pp. 3985–3994, May 2021.

[4] J. Ali, M. Tariq, and T. A. Qureshi, “A PID Control Strategy for Improving Positional Accuracy in CNC Milling Machines,” *IEEE Access*, vol. 9, pp. 72650–72659, 2021.

[5] N. Rahman and P. Subramaniam, “An Embedded System Based PID Controller for a 3D Printer and CNC Platform,” in *Proc. 12th Int. Conf. on Robotics, Vision, Signal Processing and Power Applications (ROVISP)*, pp. 231–237, 2021.

[6] A. Mohan, S. R. Khan, and P. Bhatia, “Closed-loop Feedback Control in Microcontroller-Based CNC Platforms,” *IEEE Embedded Systems Letters*, vol. 14, no. 1, pp. 25–29, Mar. 2022.

[7] D. Patel and S. Mehta, “Affordable CNC Control Systems Using ESP32 and Arduino Mega with PID Tuning,” in *Proc. Int. Conf. on Emerging Electronics and Automation (ICEEA)*, pp. 88–93, *2022.*

[8] K. N. Dey and B. Sarkar, “Design Optimization of Low-Cost CNC Systems Using Stepper Motor Kinematics,” *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 3, pp. 1235–1244,2022.

[9] M. S. Farooq and A. Bashir, “Real-Time G-Code Processing and Motion Control with Feedback Correction on Arduino Mega,” *IEEE Access*, vol. 10, pp. 58971–58981, 2022.

[10] H. Yoon and J. Kim, “A Modular CNC Architecture with ESP32-Based IoT Interface,” *IEEE Internet of Things Journal*, vol. 9, no. 17, pp. 15711–15721, Sep. 2022.

[11] M. C. Rodriguez and J. Mejía, “Statistical Validation Techniques for Low-Cost CNC Controllers,” in *Proc. IEEE Int. Conf. on Mechatronics and Automation (ICMA)*, 2023, pp. 475–480.

[12] B. Zhang and H. Yin, “PID-Based Control Enhancement for Open-Source CNC Platforms,” *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–10, Jan. 2023.

[13] S. Thapa, M. Jha, and R. Das, “A Comparative Study of GRBL and Closed-Loop Microcontroller CNC Systems,” *IEEE Access*, vol. 11, pp. 32845–32856, 2023.

[14] F. Al-Mashaqbeh and W. Qasem, “Application of Feedback Correction in Embedded Motion Control Systems,” *IEEE Sensors Journal*, vol. 23, no. 4, pp. 4150–4160, Feb. 2023.

[15] D. Liu, “G-Code Interpretation and Real-Time Path Planning in Embedded CNC Machines,” in *Proc. IEEE Int. Conf. on Industrial Technology (ICIT)*, 2024, pp. 128–134.