IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⧓ Peer-reviewed / Refereed journal ⧓ Vol. 12, Issue 2, February 2025

DOI: 10.17148/IARJSET.2025.12215

LOW-COST STEPPER MOTOR CONTROL FOR ENHANCED PERFORMANCE IN CNC MACHINE

Anene chinelo Rita[1] and Ifionu Chimnonso Jesse.[2], Joseph Ngene Aniezi [3]

Department of Physics and Industrial Physics[1]

Department of Physics and Industrial Physics[2]

Department of Industrial and Medical Physics, David Umahi Federal University of Health Sciences, Uburu, Nigeria[3]

Abstract: This research explores the potential of stepper motors on Computer Numerical Control (CNC) machines focusing on simulation and control using Arduino. Stepper motors offer advantages like easier control and lower costs, but they face challenges like oscillations and limited torque. This study presents a novel, low-cost, closed-loop control system to address these issues. . The design and implementation of a stepper motor circuit, along with relevant Arduino code, enabled precise motor movement. The simulation results showed that optimizing delay and speed settings significantly improved the motor's performance, achieving faster and more reliable responses suitable for practical CNC applications.

Keynote: CNC, STEPPER MOTOR, CLOSED LOOP, LOW COST.

I. INTRODUCTION

Computer Numerical Control (CNC) machines have revolutionized modern manufacturing by enabling precise and efficient production processes. A crucial component of CNC machines is the motor system, which requires high precision, reliability, and cost-effectiveness. Stepper motors have emerged as a promising solution, offering advantages such as easier control, lower costs, and high positioning accuracy which is crucial in both professional manufacturing environments and hobbyist applications (Chinelo et al., 2019; Bhale et al., 2016).. However, stepper motors also face challenges, including oscillations, limited torque, and difficulty in achieving smooth motion.

To address these challenges, researchers and engineers have explored various control strategies and techniques. Closed-loop control systems have shown particular promise in enhancing stepper motor performance. However, these systems often require specialized hardware and software, increasing costs and complexity.

This study aims to develop a novel, low-cost, closed-loop control system for stepper motors in CNC machines. By leveraging the versatility and affordability of Arduino, this research seeks to optimize stepper motor performance, achieving faster and more reliable responses suitable for practical CNC applications. The proposed system has the potential to enhance the precision, efficiency, and cost-effectiveness of CNC machines, making them more accessible and viable for a wide range of manufacturing applications.

II. MATERIALS AND METHOD

The following components and tools were used for the simulation and control of the stepper motor in the CNC machine:

- Proteus Software: For designing and simulating the stepper motor control circuit.
- Arduino Uno: A microcontroller board used to control the stepper motor.
- L293D Motor Driver: An H-Bridge motor driver IC for bidirectional control of the stepper motor.
- 4-Phase Stepper Motor: For precise rotational movements.
- Connecting Wires: For linking the components and ensuring smooth data and current flow.
- Arduino IDE Software (Version 2.3.3): For writing, compiling, and uploading code to the Arduino microcontroller.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.066 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 2, February 2025

DOI: 10.17148/IARJSET.2025.12215

Fig 1 : ARDRUNO UNO

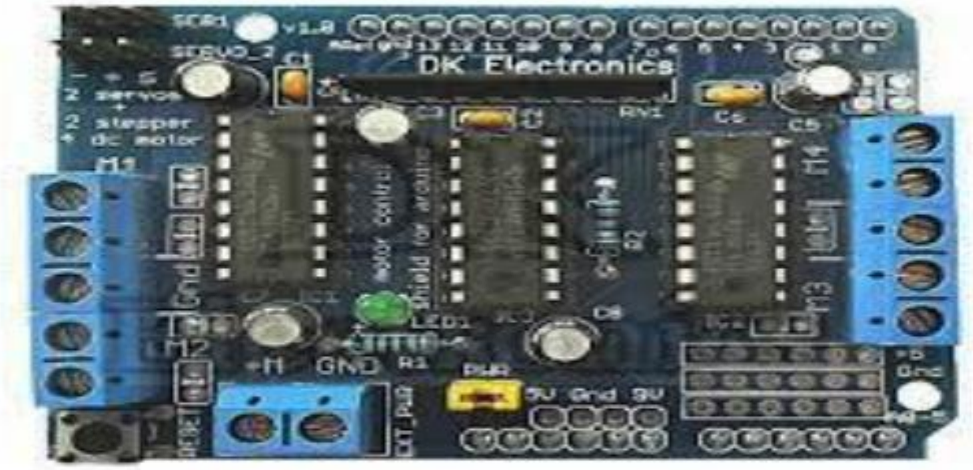

FIGURE 2: MULTI-MOTOR-DRIVER-SHIELD (L293D)

FIGURE 3: STEPPER MOTOR

DOI: 10.17148/IARJSET.2025.12215

Circuit Design
A circuit was designed using Proteus Software to simulate a stepper motor controlled by an Arduino Uno. The L293D motor driver was integrated to regulate current flow to the motor, enabling bidirectional rotation.

Code Development
The Arduino IDE was utilized to develop and upload code to the Arduino Uno, controlling the stepper motor's steps and speed. The motor was programmed to rotate 60 steps forward, followed by 40 steps backward, with a delay to facilitate observation of its performance.

```
#include <Stepper.h>
const int spr = 4;
Stepper stepper(spr, 8, 9, 10, 11);
void setup() {
  stepper.setSpeed(10);
  Serial.begin(9600);
}
void loop() {
  stepper.step(60);
  delay(200);
  stepper.step(-40);
  delay(200);
}
```

Step-by-Step Process
Proteus Software Simulation
1. The project began with simulating the circuit design in Proteus software to verify correct connections and functionality.
2. Critical components, including the Arduino Uno, L293D motor driver, and stepper motor, were tested to ensure proper control signals and motor response.
3. The simulation confirmed the stepper motor's rotational movement, providing confidence in the circuit design for real-world application.

Arduino Code Implementation
1. After validating the simulation, the code was implemented in Arduino IDE to control the stepper motor's movement and speed.
2. The code managed the motor's direction and step count, programming it to rotate 60 steps forward and 40 steps backward.
3. The motor's speed was set to 10 RPM, with delays introduced between movements for observation.

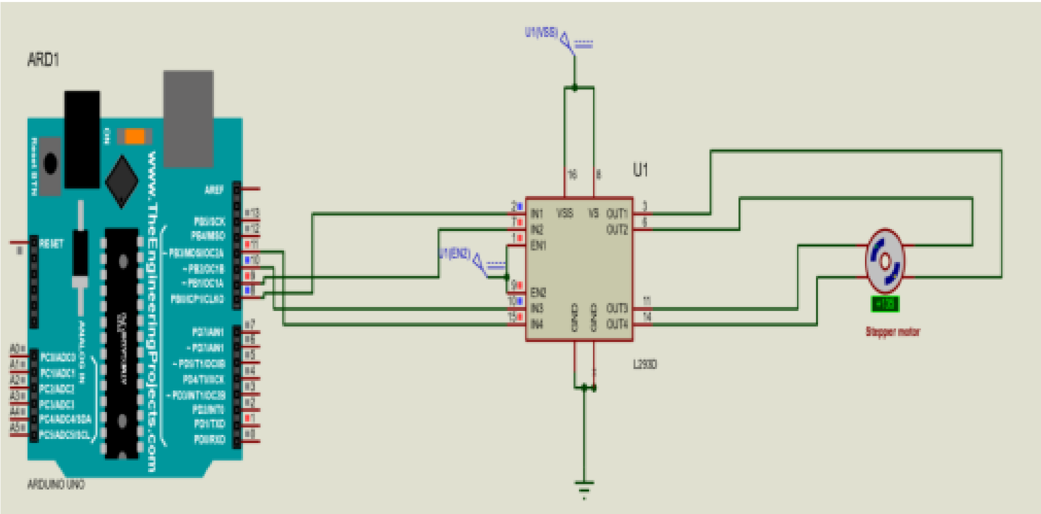

FIGURE 4: DIAGRAM OF THE SIMULATION OF STEPPER MOTOR

DOI: 10.17148/IARJSET.2025.12215

Simplification of Instruments

1. Initially, additional instruments like Logic Analyzers and Oscilloscopes were considered to capture detailed performance data.
2. However, the project focus was narrowed to demonstrate basic CNC motor control, omitting unnecessary complexity.
3. The stepper motor's performance was evaluated based on its movement and response to the code, ensuring the project met its primary goal moved by 40 steps backward, with a delay to facilitate observation of its performance

III. RESULTS AND ANALYSIS

PROTEUS SOFTWARE SIMULATION:

The Proteus software simulation was conducted to verify the correct operation of the components, particularly the stepper motor and motor driver. The simulation setup consisted of the stepper motor connected to the Arduino Uno board via the L293D motor driver. The circuit was powered by the microcontroller.

Stepper Motor Terminal Voltages

The stepper motor's terminal voltages were recorded using Proteus simulation tools. The readings are presented below

A1	11.90V
A2	98.37mV
B1	98.37mV
B2	11.90V

The simulation results confirmed that the stepper motor and motor driver operated correctly. The recorded terminal voltages indicated that the motor driver successfully controlled the current flow to the stepper motor. This verification step ensured that the physical implementation would function as intended.

Arduino Code Implementation

The Arduino IDE was utilized to develop and upload the code controlling the stepper motor. Initially, the motor was programmed to rotate 60 steps forward and 40 steps backward in an alternating pattern, with a speed setting of 10 RPM and a 200-millisecond delay between each step.

Initial Code Limitations

Although the initial code effectively drove the stepper motor, a noticeable delay between movements resulted in slower rotation speeds.

Adjustments and Optimizations

To enhance performance, the following adjustments were made:

1. Speed Adjustment: The stepper motor's speed was increased from 10 RPM to 30 RPM, improving rotational movement and reducing delay.
2. Delay Reduction: The delay between steps was decreased from 200 milliseconds to 100 milliseconds, resulting in smoother and quicker rotation.

Updated Code

The updated code reflecting these optimizations is:

```
#include <Stepper.h>
const int spr = 4;
Stepper stepper(spr, 8, 9, 10, 11);
void setup() {
  stepper.setSpeed(30); // Increased speed
  Serial.begin(9600);
}
void loop() {
  stepper.step(60);
```

DOI: 10.17148/IARJSET.2025.12215

```
  delay(100); // Reduced delay
  stepper.step(-40);
  delay(100);
}
```

Performance Improvement

The optimized code significantly enhanced the motor's performance, reducing the delay between rotations to 1.5 seconds. This improvement makes the system more suitable for CNC applications requiring faster response times.

IV. RESULTS

Stepper Motor Behavior

The stepper motor consistently rotated according to the provided commands, with improved efficiency and minimal lag between movements after optimization. The motor executed 60 steps forward and 40 steps backward in a controlled manner, exhibiting minimal oscillations and instability. The delay between movements was reduced to 1.5 seconds, considered optimal for this setup.

Oscilloscope Readings

Oscilloscope measurements were taken to verify the voltage levels at different points on the Arduino Uno pins controlling the stepper motor. The readings confirmed that the motor received correct signals from the Arduino board, facilitating proper movement.

A (Pin 11)	5V
A (Pin 11)	5V
B (Pin 10)	0V
D (Pin 9)	0V

Oscillations and Stability

Initial minor oscillations were within acceptable limits. After adjusting the speed and delay, the stepper motor displayed stable and consistent movement. The updated settings resulted in smoother transitions between steps, reducing noticeable instability.

Terminal Voltages

A1	11.90V
A2	98.37mV
B1	98.37mV
B2	11.90V

These consistent voltage readings confirmed the stepper motor's proper operation and receipt of correct control signals.

V. CONCLUSION

This study successfully demonstrated the application of a stepper motor in a computer numerical control (CNC) machine setup, focusing on simulation and control using Arduino. The design and implementation of a stepper motor circuit, along with relevant Arduino code, enabled precise motor movement. The simulation results showed that optimizing delay and speed settings significantly improved the motor's performance, achieving faster and more reliable responses suitable for practical CNC applications.

DOI: 10.17148/IARJSET.2025.12215

Recommendations

Future research directions and practical applications could consider the following:

1. Incorporating Additional Instruments: Utilizing logic analyzers or oscilloscopes could provide more precise voltage and current measurements in real-time, offering valuable insights into the motor's behavior.
2. Physical Implementation: Validating the simulation results through physical implementation in hardware would provide hands-on experience and further confirm the findings.
3. Stepper Motor Optimization: Fine-tuning motor speed and step delays for specific CNC machine applications could ensure maximum efficiency and accuracy, potentially leading to improved manufacturing outcomes.

By exploring these avenues, researchers and practitioners can further advance the application of stepper motors in CNC machines, ultimately enhancing the precision and efficiency of manufacturing processes.

REFERENCES

[1]. Chinelo R. Anene, Onyeka A. Ikenga(2019) " A Survey on Low Cost Function Generator for Stepper Motor" International Journal of Innovative Research in Science,Engineering and Technology Vol. 8, Issue 10, October 2019

[2]. Bhale, P., et al. (2016). Stepper motor driven CNC machine: A review. International Journal of Advanced Research in Computer Science, 7(3), 1-5.

[3]. Lee, J., et al. (2020). Optimization of stepper motor control for CNC machines. Journal of Intelligent Information Systems, 56(2), 257-271.

[4]. Xiao W., Zheng L., Huan J., Lei P., "Integration of Stepper Motors in CNC Machines," Robotics and Computer-Integrated Manufacturing, 2015.

[5]. M. L. Duc et al.: Design and Implement Low-Cost Industry 4.0 System Using Hybrid SS Methodology FOR CNC2023 IEEE ACCESS ONLINE ACCESSED ON 27TH JAN 2025 4.13 AM

[6]. Sameera and Mohd Asif Hasan(2017) A Study of Stepper Motors for Performance Improvement of the CNC MachinesAdvanced Research in Electrical and Electronic Engineeringp-ISSN: 2349-5804; e-ISSN: 2349-5812 Volume 4, Issue 4, ONLINE ACCESSED 26TH JAN 2025 5.08 AM

[7]. Hughes, A. (2006), Electric Motors and Drives – Fundamentals,Types and Applications", "Newnes (An imprint of Elsevier Ltd.),UK

[8]. Yongmao Xiao ET AL, (2021) A novel approach to CNC machining center processing parameters optimization considering energy-saving and low-cost" Journal of Manufacturing Systems"Volume 59, April 2021, Pages 535-548 https://doi.org/10.1016/j.jmsy.2021.03.023 ACCESSED ON 26TH JAN 2025 5.30AM

[9]. Pradeep Kumar G et al, (2020) A NOVEL TECHNIQUE FOR CONTROLLING CNC SYSTEMS BY USING ARDUINO, IJARIIE-ISSN(O)-2395-4396 Vol-6 Issue-1 2020

[10]. Ognyan Manolov et al, (2018) "Low-cost Control of Stepper Motors with Multi-purpose for Robotics" • BULGARIAN ACADEMY OF SCIENCES PROBLEMS OF ENGINEERING CYBERNETICS AND ROBOTICS, 69 София • 2018 • Sofia . o https://www.academia.edu/73725260/Low_cost_Control_of_Stepper_Motors_with_Multi_purpose_for_Robotics accessed online 26th jan 2025 9.01am.

[11]. Ms Veenita Swarnkar et al (2022)," Stepper Motor Controller Using Arduino (ATMEGA 328)" International Journal of Research in Engineering and Science (IJRES) ISSN (Online): 2320-9364, ISSN (Print): 2320-9356 www.ijres.org Volume 10 Issue 5 ǁ 2022 ǁ PP. 32-34

[12]. Stepper Motor and Its Wide Application in CNC Machining https://www.casunsteppermotor.com › news › stepper-mo...casunsteppermotor.com ONLINE ACCESSED JAN 25TH 8.59 PM

[13]. Arduino Software (IDE), https://www.arduino.cc/en/Guide/Environment , from ©Arduino

