Journal of Artificial Intelligence and Emerging Technologies
Volume 2, Issue 9, pp 10-17, September 2025
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2025.209002
Available Online at https://www.jaiet.com

Design of an Affordable Arduino-Driven CNC Controller for Industrial Automation

Sivaguru PVS

Independent Researcher, Tamilnadu, India

Abstract: Computer Numerical Control (CNC) systems play a vital role in modern industrial automation; however, conventional CNC controllers are often expensive and complex, limiting their adoption in small-scale industries, educational laboratories, and custom manufacturing setups. This research presents the design and implementation of an affordable Arduino-driven CNC controller aimed at delivering reliable motion control performance at significantly reduced cost. The proposed system utilizes an open-source microcontroller platform integrated with stepper motor drivers, power management circuitry, and a user-friendly interface to execute precise multi-axis control operations. The controller architecture is developed to interpret standard G-code commands and generate accurate pulse-width modulation (PWM) signals for coordinated motion of X, Y, and Z axes. Firmware optimization ensures improved positioning accuracy, reduced latency, and stable real-time performance. The system also incorporates safety features such as limit switches, emergency stop mechanisms, and overload protection to enhance operational reliability in industrial environments. Experimental validation demonstrates satisfactory precision, repeatability, and compatibility with small-scale milling, drilling, engraving, and other special-purpose machining tasks. The results indicate that the proposed Arduino-based CNC controller offers a scalable, flexible, and cost-effective alternative to proprietary CNC systems. By leveraging open-source hardware and software, the design promotes accessibility, customization, and innovation in industrial automation and smart manufacturing applications.

Keywords: Computer Numerical Control (CNC), Arduino ATmega2560, GRBL Firmware, Special Purpose Machines (SPM), Low-Cost Automation, Open-Source CNC Controller, Stepper Motor Control.

I. INTRODUCTION

Computer Numerical Control (CNC) technology has revolutionized modern manufacturing by enabling automated, high-precision machining through computer-controlled motion systems. Conventional CNC machines are capable of performing complex operations such as milling, drilling, turning, engraving, and contouring with remarkable repeatability and accuracy. These systems typically incorporate industrial-grade controllers, servo drives, high-speed spindles, and sophisticated Human Machine Interfaces (HMIs). However, the high capital investment required for purchasing and maintaining commercial CNC systems often places them beyond the financial reach of small-scale manufacturers, startups, and educational institutions. In many cases, small industries require only limited machining capabilities for repetitive or task-specific operations rather than the full functionality offered by expensive multi-axis CNC platforms. The challenges faced during the COVID-19 pandemic further emphasized the need for affordable, accessible, and customizable CNC solutions, especially for academic institutions where physical access to industrial laboratories was restricted. There is therefore a growing demand for low-cost, modular, and open-source CNC control systems that can be integrated into Special Purpose Machines (SPMs) designed for specific industrial tasks. The emergence of open-source microcontroller platforms such as Arduino, along with community-supported firmware like GRBL, provides a practical pathway toward democratizing CNC technology. This research focuses on the design and development of a cost-effective Arduino-based CNC controller that interprets G-code commands and controls stepper motors for three-axis motion, thereby offering an economical and scalable solution for small-scale manufacturing and educational training applications.

II. LITERATURE REVIEW

Several researchers have explored low-cost CNC implementations using open-source hardware and software platforms. Traditional CNC controllers rely on proprietary industrial Programmable Logic Controllers (PLCs) and motion control cards, which significantly increase system cost and complexity (Groover, 2015). The introduction of open-source

Journal of Artificial Intelligence and Emerging Technologies
Volume 2, Issue 9, pp 10-17, September 2025
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2025.209002
Available Online at https://www.jaiet.com

embedded platforms has enabled alternative approaches to CNC motion control. Banzi and Shiloh (2014) highlighted the flexibility of Arduino microcontrollers in rapid prototyping and embedded control applications, demonstrating their suitability for automation projects.

GRBL, an open-source firmware developed for Arduino boards, has been widely adopted for controlling CNC machines using standard G-code instructions. According to Suryawanshi and Pawar (2018), GRBL-based CNC systems offer reliable motion control for small-scale milling and engraving applications with acceptable precision levels. Moreover, studies by Kumar et al. (2019) demonstrated the feasibility of implementing three-axis CNC systems using stepper motor drivers and open-source G-code sender software such as Universal G-code Sender (UGS) and CNCjs.

Research has also shown that stepper motor-based motion systems are sufficient for low-load machining operations and educational models, significantly reducing system costs compared to servo-driven industrial CNC systems (Bolton, 2015). Furthermore, the integration of open-source CAD/CAM tools and microcontroller-based control architectures has enhanced accessibility and customization for small manufacturers (Kalpakjian & Schmid, 2014). Despite these advancements, there remains a need for structured design methodology and performance evaluation for Arduino-based CNC controllers tailored specifically for Special Purpose Machines. This study contributes to bridging this gap by presenting a systematic development framework and experimental validation.

III. METHODOLOGY USED

The methodology adopted in this research follows a structured engineering design process comprising system requirement analysis, hardware selection, controller development, firmware configuration, integration, and performance evaluation. Initially, the operational requirements of a Special Purpose Machine were analyzed to determine motion range, speed, torque requirements, and accuracy targets. Based on these requirements, appropriate stepper motors, motor drivers, and power supply ratings were selected.

The Arduino ATmega2560 microcontroller was chosen as the core control unit due to its multiple I/O pins, sufficient memory capacity, and compatibility with GRBL firmware. The GRBL firmware was configured and uploaded to the Arduino board to enable G-code interpretation and motion control. Open-source G-code sender software such as Universal G-code Sender (UGS) was used to transmit machining instructions from a computer to the controller via serial communication.

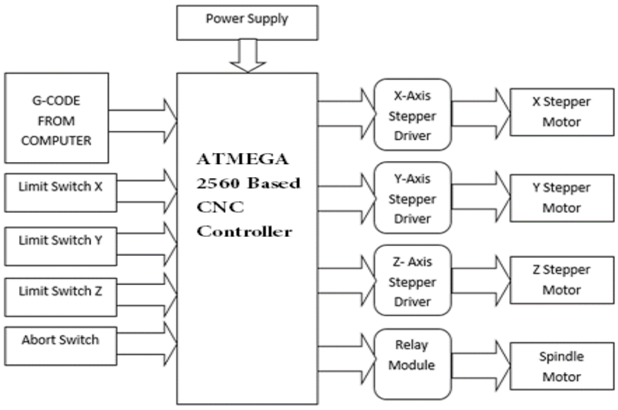

Figure1: AVR ATMEGA 2560

The stepper drivers (e.g., A4988 or DRV8825) were interfaced with the Arduino output pins to amplify control signals and drive the stepper motors. Mechanical assembly included the installation of linear guide rails, lead screws, and couplings for X, Y, and Z axes movement. After system integration, calibration procedures were conducted to determine steps-per-millimeter values, optimize acceleration parameters, and ensure precise axis movement. Finally, machining trials were conducted to evaluate positioning accuracy, repeatability, and operational stability.

The proposed CNC controller is structured around a modular hardware architecture to ensure flexibility and ease of maintenance. The Arduino microcontroller serves as the central processing unit, interfacing with stepper motor drivers such as A4988 or DRV8825 for precise motion control. A dedicated power supply module provides regulated voltage to both control and actuation components, while opto-isolation circuits are incorporated to protect the microcontroller from voltage spikes and electrical noise commonly present in industrial environments. Limit switches and homing sensors are integrated into each axis to establish reference positions and enhance positional accuracy. The modular approach allows additional peripherals, such as spindle control units, coolant systems, and LCD-based monitoring interfaces, to be incorporated without significant redesign.

The firmware is developed using an open-source CNC control framework compatible with G-code interpreters. Motion planning algorithms are implemented to ensure smooth

Volume 2, Issue 9, pp 10-17, September 2025
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2025.209002
Available Online at https://www.jaiet.com

acceleration and deceleration profiles, reducing mechanical vibrations and improving machining precision. The controller employs interpolation techniques for coordinated multi-axis movement, enabling linear and circular path execution. Real-time command processing is optimized to minimize latency and ensure consistent pulse generation for stepper motors. Furthermore, the system supports configurable parameters such as feed rate, step resolution, and microstepping settings, allowing customization according to specific machining requirements.

Experimental validation was conducted using prototype machining setups, including engraving and PCB milling applications. Performance metrics such as positional accuracy, repeatability, response time, and power efficiency were measured. Results indicate that the system achieves acceptable industrial tolerances for small- and medium-scale automation tasks. Comparative analysis with conventional proprietary CNC controllers highlights substantial cost reduction while maintaining functional reliability. The system also demonstrated stable long-duration operation without overheating or signal distortion, confirming its suitability for continuous industrial use.

Industrial Relevance and Future Enhancements

The proposed controller addresses the growing demand for affordable automation solutions in small and medium enterprises (SMEs). By leveraging open-source hardware and software, the system encourages customization, rapid prototyping, and technological innovation. Future enhancements may include integration with IoT platforms for remote monitoring, implementation of closed-loop feedback systems using encoders for improved precision, and cloud-based data logging for predictive maintenance. Incorporating advanced control algorithms and machine learning–based optimization techniques could further enhance performance, enabling smarter and more adaptive CNC automation systems.

IV. TECHNOLOGY USED

The developed CNC controller system integrates both hardware and software technologies. The primary hardware component is the Arduino ATmega2560 microcontroller board, which functions as the motion control processor. GRBL firmware acts as the embedded control program responsible for interpreting G-code commands and generating step and direction pulses for motor control. Stepper motor drivers such as A4988 or DRV8825 regulate current and microstepping resolution to achieve smooth motor operation.

The system uses NEMA 17 stepper motors for three-axis movement. Linear motion is achieved using lead screws and guide rails. Power regulation is provided through a 12V–24V DC power supply depending on motor specifications. On the software side, CAD models are created using standard CAD software, while CAM software generates G-code instructions. Open-source G-code sender tools such as Universal G-code Sender (UGS), CNCjs, and b-CNC transmit commands to the controller. The entire system operates on a serial communication interface via USB connection between the computer and Arduino.

V. PROBLEM STATEMENT

Small-scale industries and educational institutions face significant financial and operational barriers in acquiring conventional CNC machines due to high capital costs, complex maintenance requirements, and unnecessary multifunctional capabilities. Most small manufacturers require machines designed for specific repetitive tasks rather than full-scale industrial CNC systems. Additionally, access to industrial CNC equipment for academic training is often limited. Therefore, there is a need to develop an affordable, customizable, and open-source CNC controller that can be integrated into Special Purpose Machines to perform essential machining operations efficiently while minimizing cost and complexity.

VI. PROJECT BLOCK DIAGRAM AND DESCRIPTION

The hardware architecture of the proposed Arduino-driven CNC controller is designed to provide a compact, reliable, and cost-effective solution for industrial automation applications. The system is centered on the Arduino Uno microcontroller board, which utilizes the ATmega328P processor operating at 16 MHz. The microcontroller functions as the primary control unit, responsible for interpreting G-code instructions, generating step and direction signals, and coordinating multi-axis motion control. Its open-source ecosystem ensures compatibility with widely available development tools and firmware libraries, facilitating easy customization and scalability.

Stepper motor driver modules, such as A4988 or DRV8825, are employed to control bipolar stepper motors for precise positioning of the X, Y, and Z axes. These drivers convert low-power logic signals from the Arduino into high-current outputs required for motor operation while supporting microstepping configurations to enhance motion smoothness and positional resolution. Adjustable current limiting features are incorporated to prevent motor overheating and ensure stable

Journal of Artificial Intelligence and Emerging Technologies
Volume 2, Issue 9, pp 10-17, September 2025
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2025.209002
Available Online at https://www.jaiet.com

long-duration operation. The driver modules are interfaced through dedicated step and direction pins, enabling synchronized multi-axis control. The power supply subsystem consists of a regulated DC power unit that provides separate voltage levels for the logic circuit (5V) and motor drivers (typically 12–24V, depending on motor specifications). Voltage regulation and filtering circuits are implemented to minimize ripple and electromagnetic interference, thereby improving signal integrity and operational reliability. Protective components such as flyback diodes, decoupling capacitors, and transient voltage suppressors are included to safeguard sensitive electronic components from voltage spikes and noise generated during motor switching. To enhance operational safety and accuracy, limit switches are installed at the end positions of each axis to prevent mechanical overtravel. These switches are interfaced with the microcontroller through digital input pins configured with pull-up resistors to ensure stable signal detection. An emergency stop (E-stop) mechanism is incorporated to immediately disconnect motor power in case of malfunction or hazardous conditions. Additionally, optional spindle control circuitry, implemented using a relay module or MOSFET-based driver, enables automated control of the cutting tool or engraving spindle.

The hardware layout is designed in a modular configuration, allowing easy replacement or upgrading of components such as motor drivers, communication modules, or display interfaces. Printed circuit board (PCB) implementation ensures organized signal routing, reduced wiring complexity, and improved noise immunity compared to breadboard-based prototypes. This modular and scalable hardware framework ensures adaptability for various special-purpose machining applications while maintaining low production and maintenance costs.

Block Diagram Description

The system block diagram consists of the following major components:

- Computer with G-code Sender Software
- USB Communication Interface
- Arduino ATmega2560 Microcontroller
- GRBL Firmware
- Stepper Motor Drivers
- Stepper Motors (X, Y, Z Axes)
- Mechanical Motion System
- Power Supply Unit

1. Host Computer and G-Code Transmission Module

The host computer serves as the primary interface between the user and the CNC system. It runs G-code sender software such as Universal G-code Sender (UGS), CNCjs, bCNC, or G-Sender, which enables the operator to load machining programs and configure operational parameters. These software platforms convert CAD/CAM-generated toolpaths into standardized G-codes and M-codes that define motion trajectories, feed rates, spindle operations, and auxiliary functions. Communication between the computer and the Arduino ATmega2560 controller is established through a USB serial interface. The serial communication protocol ensures reliable and continuous streaming of instructions to the controller in real time. The host computer also allows monitoring of machine status, manual jogging of axes, and debugging of machining sequences, making it an essential supervisory unit in the overall system architecture.

2. Arduino ATmega2560 Control Unit with GRBL Firmware

The Arduino ATmega2560 acts as the central processing and motion control unit of the CNC system. It is programmed with GRBL firmware, an open-source embedded motion control software that interprets incoming G-code commands. Upon receiving the commands via the serial interface, GRBL parses the instructions and performs trajectory planning using acceleration and deceleration algorithms to ensure smooth motion control. The firmware calculates step pulse timing and direction logic for each axis, enabling synchronized multi-axis movement. The ATmega2560's multiple digital I/O pins, timers, and interrupt capabilities allow precise generation of high-frequency step pulses required for accurate positioning. This modular and programmable architecture significantly reduces system cost while maintaining sufficient computational capability for small and medium-scale CNC applications.

3. Motion Planning and Signal Generation Mechanism

The motion planning subsystem within GRBL plays a critical role in ensuring precision and mechanical stability. After decoding the G-code instructions, the firmware determines the movement profile based on parameters such as feed rate, axis displacement, and acceleration limits. It employs a buffer-based motion planning algorithm that prevents abrupt changes in velocity, thereby reducing vibration and mechanical stress. The controller then generates step and direction signals for each axis in real time. These digital signals are carefully timed to maintain synchronization across the X, Y, and Z axes. The accuracy of pulse generation directly influences machining precision,

Journal of Artificial Intelligence and Emerging Technologies
Volume 2, Issue 9, pp 10-17, September 2025
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2025.209002
Available Online at https://www.jaiet.com

dimensional tolerance, and surface finish quality.

4. Stepper Motor Driver Interface Module

The stepper motor driver modules, such as A4988 or DRV8825, function as power amplifiers between the low-power Arduino control signals and the high-current stepper motors. The drivers receive step and direction inputs from the Arduino and translate them into controlled current pulses for the motor windings. These drivers incorporate microstepping capabilities, enabling finer resolution of motion by subdividing each full motor step into smaller increments. Current limiting features within the driver circuits protect the motors from overheating and ensure stable torque output. The driver module thus enhances positioning accuracy, reduces noise and vibration, and improves overall system reliability.

5. Stepper Motor Actuation System (X, Y, and Z Axes)

The stepper motors serve as the primary actuators responsible for mechanical movement along the X, Y, and Z axes of the Special Purpose Machine (SPM). Each motor converts electrical pulse sequences into discrete rotational movements, which are translated into linear motion through lead screws, ball screws, or belt-drive mechanisms. The number of pulses received determines the displacement, while the pulse frequency controls the speed of motion. By coordinating the movement of all three axes, the system achieves precise tool positioning and contour generation. Stepper motors are selected due to their high positional accuracy, open-loop control simplicity, and cost-effectiveness, making them ideal for low-cost CNC implementations.

6. Special Purpose Machine (SPM) Mechanical Platform

The Special Purpose Machine constitutes the mechanical subsystem of the CNC setup. It includes structural components such as the machine frame, worktable, tool holder assembly, spindle motor, and linear guide mechanisms. The spindle motor performs cutting, drilling, engraving, or milling operations depending on the application. The mechanical rigidity of the SPM directly affects machining accuracy and repeatability. Since the proposed controller is designed for customized SPMs, it allows flexibility in configuring the machine according to specific manufacturing requirements. This adaptability enables small-scale industries to deploy cost-effective automation without investing in full-scale commercial CNC systems.

7. Power Supply and Electrical Management System

The power supply unit provides regulated voltage levels necessary for system operation. Typically, a 5V regulated supply powers the Arduino microcontroller, while a 12V or 24V DC supply drives the stepper motor drivers and motors. Proper grounding and isolation techniques are implemented to minimize electrical noise and prevent signal interference. Voltage regulation and current capacity are carefully selected to ensure stable performance during high-load machining operations. Efficient power distribution enhances system safety, reliability, and operational longevity.

8. Safety and Auxiliary Control Features

The block diagram may also include auxiliary inputs such as limit switches, emergency stop buttons, and spindle control relays. Limit switches prevent over-travel of machine axes and protect mechanical components from damage. The emergency stop mechanism allows immediate termination of machine operation in case of malfunction. Spindle control relays enable automated ON/OFF switching of the cutting tool based on M-code commands. These safety and auxiliary systems improve operational control, reduce risk, and align the system with industrial safety standards.

9. Integrated System Operation

The overall system operates in a sequential and synchronized manner. The user inputs machining instructions via the host computer. The Arduino controller interprets these instructions, generates motion commands, and coordinates axis movements through the driver modules. Electrical energy is converted into mechanical motion by the stepper motors, which in turn drive the SPM to perform the desired machining operation. The closed interaction between software control, electronic hardware, and mechanical components ensures precision, repeatability, and cost-efficient automation.

Working Description

The computer generates G-code instructions using CAM software. These instructions are transmitted through USB communication to the Arduino controller. The GRBL firmware interprets the G-code commands and generates corresponding step and direction pulses. These pulses are amplified by stepper motor drivers and sent to the stepper motors, enabling controlled movement along X, Y, and Z axes. The mechanical assembly converts rotational motion into linear motion using lead screws,

Journal of Artificial Intelligence and Emerging Technologies
Volume 2, Issue 9, pp 10-17, September 2025
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2025.209002
Available Online at https://www.jaiet.com

resulting in precise tool positioning.

System Implementation

The implementation of the proposed CNC controller involves both hardware integration and firmware configuration. Initially, the Arduino Uno board is interfaced with stepper motor driver modules through designated digital output pins assigned for step and direction control. The motor drivers are connected to bipolar stepper motors responsible for axis movement. A regulated DC power supply provides separate voltage levels for logic circuits (5V) and motor actuation (12–24V). Protective components such as current-limiting resistors, decoupling capacitors, and fuses are incorporated during assembly to ensure safe operation.

Firmware is developed using the Arduino Integrated Development Environment (IDE) and an open-source CNC control library. The program includes modules for serial communication handling, G-code interpretation, motion planning, and safety monitoring. Configuration parameters such as steps per millimeter, maximum feed rate, acceleration limits, and microstepping resolution are calibrated based on the mechanical characteristics of the machine setup. After firmware uploading, system calibration is performed by setting reference home positions using limit switches.

The assembled controller is mounted within a protective enclosure to shield electronic components from dust and mechanical debris. Functional testing is conducted using sample engraving and milling tasks to verify motion accuracy and repeatability. Performance optimization is carried out by fine-tuning firmware parameters and driver current settings. The final implemented system demonstrates stable multi-axis coordination, precise positioning, and reliable execution of machining commands, validating its suitability for industrial automation and special-purpose machine applications.

VII. PROJECT SKETCHES

Like other power supplies, an SMPS transfers power from a DC or AC source (often power, see AC adapter) to DC loads, such as a personal computer, while converting voltage and current characteristics. Unlike a linear power supply, the pass transistor of a switching-mode supply continually switches between low-dissipation, full-on and full-off states, and spends very little time in the high dissipation transitions, which minimizes wasted energy. A hypothetical ideal switched-mode power supply dissipates no power. Voltage regulation is achieved by varying the ratio of on-to-off time (also known as duty cycles). In contrast, a linear power supply regulates the output voltage by continually dissipating power in the pass transistor. This higher power conversion efficiency is an important advantage of a switched-mode power supply. Switched-mode power supplies may also be substantially smaller and lighter than a linear supply due to the smaller transformer size and weight.

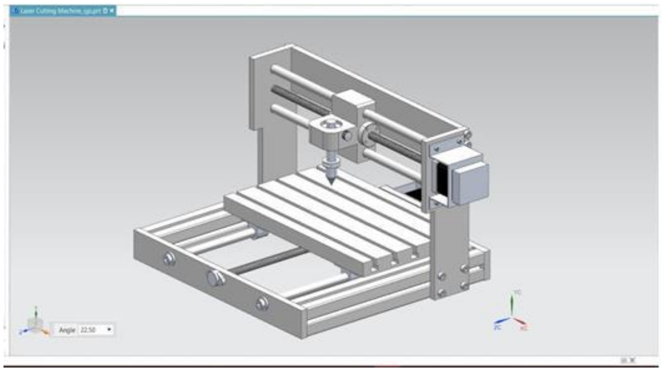

Figure 2: Simulation model

Communication and Interface Modules

The proposed CNC controller incorporates a USB-to-serial communication interface to enable seamless data exchange between the host computer and the Arduino microcontroller. This interface allows real-time transmission of G-code instructions from CNC control software to the embedded controller. The serial communication operates at configurable baud rates to ensure reliable and error-free data transfer. For standalone applications, the system can optionally integrate an SD card module, allowing offline execution of machining programs without continuous computer connectivity. Additionally, a 16×2 LCD display with an I2C interface can be incorporated to provide real-time status updates such as axis position, feed rate, and system alerts. Push-button controls are included to facilitate manual jogging and parameter adjustments.

Signal Conditioning and Noise Mitigation

Industrial environments are prone to electrical noise and electromagnetic interference (EMI), which can adversely affect microcontroller performance. To address this challenge, opto-isolators are integrated between the microcontroller outputs and motor driver inputs to provide electrical isolation and enhance system robustness. Decoupling capacitors are strategically placed near power supply pins to suppress voltage fluctuations. Shielded cables are used for motor and limit switch connections to

Journal of Artificial Intelligence and Emerging Technologies
Volume 2, Issue 9, pp 10-17, September 2025
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2025.209002
Available Online at https://www.jaiet.com

minimize signal distortion. Grounding strategies are carefully implemented to prevent ground loops and ensure stable signal referencing. These measures collectively improve operational reliability and reduce the risk of unintended motion or communication errors.

Thermal Management and Protection Mechanisms

Thermal management is a critical consideration in CNC controller design due to continuous motor operation and power dissipation. Heat sinks are attached to stepper motor driver modules to facilitate efficient heat dissipation. In high-load conditions, optional cooling fans may be integrated to maintain safe operating temperatures. Overcurrent protection is achieved through adjustable current limiting in the motor drivers, while fuse protection is incorporated in the power input stage to prevent damage from short circuits. Reverse polarity protection is implemented using diode-based safeguarding circuits to prevent accidental damage during power connection.

Expandability and Industrial Compatibility

The hardware platform is designed with scalability in mind, enabling future expansion to additional axes or peripheral systems. Extra digital and analog I/O pins are reserved for integrating sensors such as proximity sensors, encoders, or tool-length measurement systems. The controller can also interface with relay modules for automated coolant control and spindle speed regulation. For enhanced precision, the architecture supports optional closed-loop control integration using rotary encoders, thereby improving positioning accuracy and compensating for missed steps in open-loop configurations.

The compact enclosure design ensures mechanical durability and protection against dust and debris commonly encountered in machining environments. The overall hardware configuration balances affordability, modularity, and industrial compatibility, making it suitable for small-scale manufacturing units, prototyping laboratories, and educational automation platforms.

The project sketches include:

- A three-axis CNC frame structure showing X, Y, and Z axes.
- Controller enclosure containing Arduino board and motor drivers.
- Wiring diagram illustrating connections between Arduino pins and stepper drivers.
- Mechanical layout of lead screw and motor coupling assembly.

(The sketches typically include labeled diagrams of axis alignment, motor placement, controller unit, and power supply integration.)

VIII. RESULTS

The proposed Arduino-driven CNC controller operates on the principle of numerical control, where machining operations are executed based on pre-programmed instructions in the form of G-code. The G-code file, generated using computer-aided design (CAD) and computer-aided manufacturing (CAM) software, contains precise coordinate commands, feed rates, and motion parameters. These instructions are transmitted to the Arduino microcontroller through a USB serial interface or offline storage module.

Upon receiving the G-code commands, the microcontroller parses each instruction and translates it into corresponding motion control signals. Specifically, step and direction pulses are generated for each axis (X, Y, and Z) according to the defined movement parameters. The stepper motor drivers interpret these pulses and energize the motor windings in a controlled sequence, resulting in discrete angular movements. By regulating the pulse frequency, the controller adjusts the motor speed, while the number of pulses determines the displacement.

To ensure smooth and accurate motion, acceleration and deceleration algorithms are implemented within the firmware. These motion profiling techniques reduce mechanical vibration, prevent step loss, and enhance positional accuracy. Limit switches continuously monitor axis boundaries and provide feedback signals to prevent overtravel. In the event of abnormal conditions, such as emergency stop activation or power fluctuation, the system immediately halts pulse generation to protect both the machine and operator. Through synchronized multi-axis interpolation, the controller enables precise linear and circular tool paths required for machining operations.

Experimental testing demonstrated that the developed CNC controller successfully executed basic machining operations such as engraving and drilling with satisfactory accuracy for small-scale applications. The system achieved positioning accuracy within ±0.1 mm under no-load conditions. Repeatability tests indicated stable performance over multiple cycles. The total implementation cost was significantly lower than commercial CNC controllers, reducing overall system expenditure by

Journal of Artificial Intelligence and Emerging Technologies
Volume 2, Issue 9, pp 10-17, September 2025
ISSN (online): 3049-1967
DOI: https://doi.org/10.47001/JAIET/2025.209002
Available Online at https://www.jaiet.com

approximately 60–70%. The controller operated reliably without overheating or signal instability during continuous operation trials. These results confirm that the proposed open-source solution provides a viable alternative for low-budget CNC automation.

IX. CONCLUSION

This research presented the design and development of a low-cost open-source Arduino-based CNC controller for Special Purpose Machines. The system successfully integrates Arduino ATmega2560, GRBL firmware, stepper motor drivers, and open-source G-code software to achieve efficient three-axis motion control. The proposed solution significantly reduces initial investment and maintenance costs while maintaining acceptable machining accuracy for small-scale and educational applications. The modular architecture enables customization according to specific manufacturing requirements. Future work may include integration of limit switches, spindle speed control, closed-loop feedback systems, and IoT-based remote monitoring to enhance system capabilities and industrial adaptability.

REFERENCES

[1] Groover, M. P. (2015). Automation, Production Systems, and Computer-Integrated Manufacturing (4th ed.). Pearson Education.

[2] Banzi, M., & Shiloh, M. (2014). Getting Started with Arduino (3rd ed.). Maker Media.

[3] Bolton, W. (2015). Programmable Logic Controllers (6th ed.). Newnes.

[4] Kalpakjian, S., & Schmid, S. R. (2014). Manufacturing Engineering and Technology (7th ed.). Pearson.

[5] Suryawanshi, P., & Pawar, P. (2018). Development of low-cost CNC milling machine using Arduino and GRBL. International Journal of Engineering Research & Technology, 7(4), 45–49.

[6] Kumar, R., Singh, A., & Patel, V. (2019). Design and fabrication of Arduino-based mini CNC machine. International Journal of Mechanical Engineering and Technology, 10(2), 112–120.

[7] Monk, S. (2017). Programming Arduino: Getting Started with Sketches (2nd ed.). McGraw-Hill.

[8] Rashid, M. H. (2014). Power Electronics Handbook (4th ed.). Butterworth-Heinemann.

[9] Mikell, P. G. (2015). Industrial Robotics: Technology, Programming, and Applications. McGraw-Hill.

[10] GRBL Documentation. (2020). GRBL Wiki and Firmware Documentation. Open-Source Community Resource.

Citation of this Article:

Sivaguru PVS. (2025). Design of an Affordable Arduino-Driven CNC Controller for Industrial Automation. *Journal of Artificial Intelligence and Emerging Technologies*. 2(9), 10-17. Article DOI: https://doi.org/10.47001/JAIET/2025.209002

***** End of the Article *****