DESIGN AND IMPLEMENTATION OF ARDUINO BASE 2D CNC PLOTTING MACHINE

A Project report is submitted in partial fulfillment of the requirements for the award of Degree of Bachelor of Science in Electrical and Electronic Engineering.

Submitted by

Abdus Sakur Khan
201-33-1212
Palash Chandra Roy
201-33-1205
Md. Rajaul Karim
201-33-1271
Md. Nazmul Hasan
201-33-1262

Supervised by

Ms.Tanjum Rahi Akanto
Lecturer
Department of Electrical and Electronic Engineering

Department of Electrical and Electronic Engineering

Faculty of Engineering

DAFFODIL INTERNATIONAL UNIVERSITY

JULY, 2024

DECLARATION

I hereby declare that this project, "**Design & Implemanation Arduino Base 2D CNC Plotting Machine**" represents my own work, which has been done in the laboratories of the Department of Electrical and Electronic Engineering under the Faculty of Engineering of Daffodil International University in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical and Electronic Engineering, and has not been previously included in a thesis or dissertation submitted to this or any other institution for a degree, diploma, or other qualifications. I have attempted to identify all the risks related to this research that may arise in conducting this research, obtained the relevant ethical and/or safety approval (where applicable), and acknowledged my obligations and the rights of the participants.

Signatures of the candidates

Name: Abdus Sakur khan
ID: 201-33-1212

Name: Palash Chandra Roy
ID: 201-33-1205

Name: Md. Rajaul Karim
ID: 201-33-1271

Name: Md. Nazmul Hasan
ID: 201-33-1262

APPROVAL

The project entitled **"Design & Implemanation Arduino Base 2D CNC Plotting Machine"** submitted by **Abdus Sakur Khan (201-33-1212), Palash Chandra Roy (201-33-1205), Md. Rajaul Karim (201-33-1271) & Md. Nazmul Hasan (201-33-1262),** has been done under my supervision and accepted as satisfactory in partial fulfillment of the requirements for the degree of **Bachelor of Science in Electrical and Electronic Engineering in July, 2024.**

Signed

Tanjum 07/09/24

____

Ms. Tanjum Rahi Akanto

Lecturer

Department of Electrical and Electronic Engineering

Faculty of Engineering

Daffodil International University

Dedicated

To

Our Parents

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

DIU	DAFFODIL INTERNATIONAL UNIVERSITY
EEE	ELECTRICAL AND ELECTRONIC ENGINEERING
CNC	COMPUTER NUMERICAL CONTROL
PCB	PRINTED CIRCUIT BOARD
ISO	INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
SVG	SCALABLE VECTOR GRAPHICS
OSHA	OCCUPATIONAL SAFETY AND HEALTH ADMINISTRATION
LCD	LIQUID CRYSTAL DISPLAY
OSHWA	OSHWA OPEN-SOURCE HARDWARE ASSOCIATION
CAM	COMPUTER-AIDED MANUFACTURING
UGS	UNIVERSAL G-CODE SENDER
ATC	AUTOMATIC TOOL CHANGING
DXF	DRAWING EXCHANGE FORMAT
PID	PROPORTIONAL INTEGRAL DERIVATIVE
BOM	BILL OF MATERIALS

ACKNOWLEDGEMENT

First of all, we want to give thanks to **Almighty Allah**. With his blessing we are able to complete our work with the best effort. We want to pay our utmost respect to our Supervisor **Ms. Tanjum Rahi Akanto** Lecturer of the **Department of EEE, Daffodil International University** for who has given us the chance to work on an impactful idea and taken care of every issue of the development of this concept. Then we would like to take this opportunity to express gratitude to our supervisor for being dedicated in supporting, motivating and guiding us throughout this project. This project can't be done without his useful advice and help. Also thank him very much for giving us the opportunity to work with this project.

We also want to convey our thankfulness to **Professor Dr. Md. Shahid Ullah, Professor & Head** of the **Department of EEE, Daffodil International University** for his support and encouragement. Apart from that, we would like to thank our entire class fellows for sharing knowledge; information and helping us in making this project a success. To our beloved family, we want to give them our deepest love and gratitude for being very supportive and also for their inspiration and encouragement during our studies in this Institution.

ABSTRACT

The Arduino-based 2D CNC plotting machine is a compact and versatile tool designed for enthusiasts, students, and makers interested in exploring digital fabrication. The X, Y, and Z axes of motion determine the limited region that the Arduino plotter machine operates on, based on the theory of computer numerical control. This project's goals are to design the plotter and create open-source software to operate the apparatus. Essentially, this plotter machine's system is modeled by Solid Work software to operate with three-axis stepper motors (represented as X, Y, and Z axes). The movement of these three axes stepper motors is shielded. The machine can produce complex designs with great accuracy thanks to its configuration, which enables accurate movement along three dimensions. A pen holder or similar tool attachment is mounted on the gantry, enabling it to draw or plot on a wide range of materials, including paper and thin wood. Technically speaking, a plottable machine mirrors the functionality of a CNC (Computer Numerical Control) machine. Specifically, it resembles a CNC router, albeit with a writing implement in place of a cutting tool. Interestingly, this concept is built around an open-source project that was started by BCN3D Technologies, so enthusiasts can easily replicate it. This project demonstrates the development of an affordable CNC plotter machine capable of writing or drawing art on paper or small wood through a simple algorithm and readily available components. The functionality of the CNC plotting machine is governed by an Arduino microcontroller, which serves as the brain of the system. The Arduino interprets G-code commands, a standard programming language for CNC machines, and translates them into motor control signals. This enables seamless coordination between the stepper motors and the movement of the gantry, ensuring smooth and precise plotting of designs. Software plays a critical role in the operation of the CNC plotting machine. Users have the flexibility to generate or import vector graphics files, which are then converted into G-code instructions using software tools. These G-code instructions dictate the movement of the gantry and control parameters such as speed and pen pressure, allowing users to achieve their desired drawing outcomes with ease. The Arduino-based 2D CNC plotting machine offers numerous advantages over traditional methods of drawing and plotting. Its compact size and affordability make it accessible to a wide range of users, from hobbyists to students to professionals. Additionally, its open-source nature allows for customization and expansion, enabling users to tailor the machine to their specific needs and preferences.

Keywords: Arduino, CNC, stepper motors, servo motor, Inkscape software, G-code file conversion.

CHAPTER 1

INTRODUCTION

1.1 Introduction

A 2D CNC (Computer Numerical Control) plotting machine is a device capable of moving a tool (usually a pen or a marker) across a flat surface to create drawings, designs, or other graphical representations. These machines are often used in various applications such as rapid prototyping, artwork creation, PCB (Printed Circuit Board) milling, and educational purposes. A sophisticated type of soft automation designed to regulate the movement and functionality of machine tools is computer numerical control. The manufacturing machine utilizes a coded format consisting of letters and digits when using a numerical control machine, which was developed in the 19th century to reduce workload. In the manufacturing industry, a procedure called CNC machining uses computers to operate machine equipment. In the ever-evolving landscape of digital fabrication and maker culture, the Arduino-based 2D CNC plotting machine emerges as a fascinating embodiment of creativity, technology, and innovation. This remarkable device, built upon the Arduino microcontroller platform, brings the precision and automation of computer numerical control (CNC) to the realm of two-dimensional graphical plotting. With its accessibility, versatility, and educational potential, the Arduino-based 2D CNC plotting machine has captured the imagination of hobbyists, educators, and professionals worldwide. At its essence, the CNC plotting machine represents a fusion of artistry and technology, transforming digital designs into tangible creations with unparalleled precision. By leveraging the capabilities of Arduino microcontrollers, enthusiasts can orchestrate the movement of a plotting tool—typically a pen or marker across a flat surface, translating intricate designs into physical reality. This process, known as Cartesian plotting, involves precise control of the plotting tool's position along three axes (X, Y and Z) to reproduce complex drawings, patterns, or text. The Arduino microcontroller serves as the cornerstone of the CNC plotting machine, providing the computational power and control necessary to execute precise movements and coordinate intricate patterns. Renowned for its affordability, ease of use, and robust community support, the Arduino platform empowers enthusiasts to explore the realms of CNC technology with confidence and creativity. Through a combination of hardware and software, Arduino-based CNC plotting machines offer a gateway to digital fabrication, enabling users to bring their ideas to life with unprecedented precision and efficiency. Building an Arduino-based 2D CNC plotting machine is a journey that begins with understanding the fundamental principles of CNC technology. Armed with this knowledge, they assemble the requisite components: an Arduino board, stepper motor drivers, linear motion systems, and a plotting tool holder into a cohesive structure that forms the backbone of the machine. Assembly of the CNC plotting machine involves careful consideration of mechanical and electrical components, ensuring compatibility and functionality. Linear motion systems, comprising precision rails, belts, or lead screws, provide the framework for controlled movement along the X and Y axes. Stepper motors, equipped with drivers to regulate their rotation, serve as the driving force behind the machine's movements, translating digital commands into physical motion. The plotting tool holder, whether a simple pen mount or a more sophisticated spindle mechanism, facilitates the transfer of designs onto the plotting surface. Enthusiasts harness the power of Arduino's versatile programming environment to develop custom firmware tailored to their specific needs. This firmware interprets G-code instructions, a standardized language used to control CNC machines and translates them into precise movements of the plotting tool. Calibration and testing represent crucial stages in the development of the CNC plotting machine, refining its performance and accuracy. Enthusiasts meticulously adjust motor steps, acceleration rates, and maximum speeds to achieve optimal results, fine-tuning the machine's responsiveness and precision. Through a series of test runs and iterative adjustments, they validate the machine's capabilities, ensuring that it can faithfully reproduce digital designs with fidelity and consistency. Enthusiasts load their designs into CNC control software, a bridge between the digital and physical realms that generates G-code instructions from their digital artwork. With a few clicks, intricate designs spring to life as the CNC plotting machine meticulously traces each stroke, bringing imagination to fruition with unparalleled precision and elegance. From educational workshops to artistic endeavors, its impact reverberates across disciplines, inspiring a new generation of makers to push the boundaries of what's possible. The Arduino-based 2D CNC plotting machine represents a convergence of technology, craftsmanship, and creativity. With its accessible design, modular construction, and endless potential, it empowers enthusiasts to turn dreams into reality, one plotted line at a time. As a symbol of ingenuity, it exemplifies the transformative power of DIY culture, inviting all who dare to explore the boundless realms of digital fabrication.

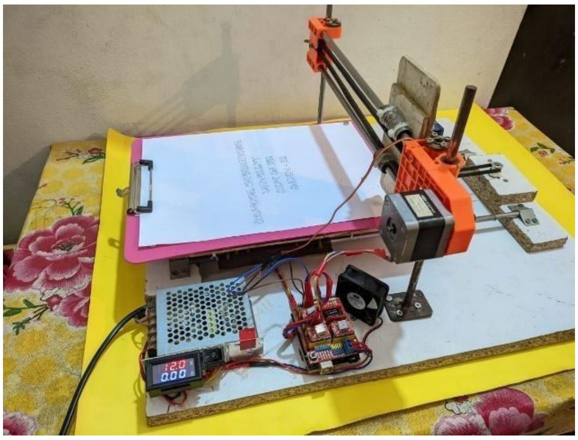
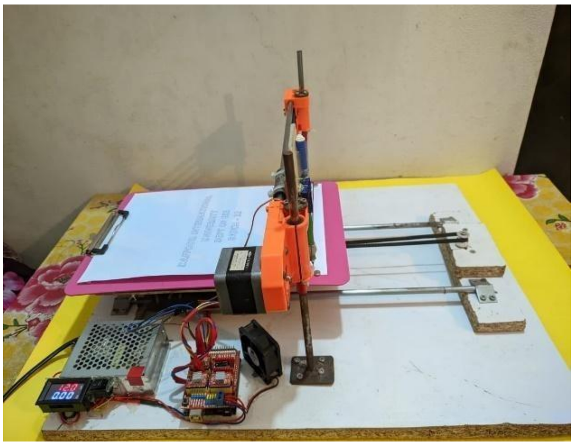



Fig1.1: Arduino Based 2D CNC Plotting Machine

1.2 Problem Statement:

The goal is to design and implement an Arduino-based 2D CNC plotting machine capable of accurately reproducing digital designs on physical surfaces. This CNC machine is intended for makers, students, and hobbyists who want to explore computer-controlled fabrication. It should be reasonably priced, simple to assemble, and straight for us.

Important prerequisites

Precision: The CNC plotting machine must achieve precise tool positioning to accurately replicate digital designs within prescribed tolerances.

Compatibility: The device should seamlessly integrate with popular drawing applications and support standard file formats like SVG, DXF, and G-code.

User Interface: The CNC machine should have an intuitive interface for easy operation, allowing users of all skill levels to set parameters, upload designs, and monitor progress.

Safety: Include safety features like limit switches, emergency stop buttons, and thermal protection for electronics and motors to prevent mishaps and protect users.

Cost-effectiveness: Design the CNC plotting machine with affordable, accessible components and open-source hardware and software to keep costs down and ensure broad accessibility.

Scalability: Offer expansion options like interchangeable tool heads and additional axes for 3D printing or milling to meet diverse user needs.

Documentation: Provide detailed documentation including schematics, a bill of materials, assembly instructions, and troubleshooting manuals to help users replicate or modify the CNC plotting machine.

Educational Value: Offer instructional materials and guides to help students and hobbyists learn programming, robotics, and mechatronics using the CNC plotting machine as a teaching tool.

Reliability: Ensure reliability with high-quality components and strict testing for continuous, trouble-free operation.

Community Engagement: Foster a community for collaboration and innovation, offering a platform to share designs, modifications, and experiences.

1.3 Objectives

Depending on the particular needs and aims of the project, the aim of an Arduino-based 2D CNC plotting machine project may differ, however, the following are some typical purposes:

- **Design and Build:** Use linear guides, stepper motors, and an Arduino microcontroller for a small, affordable CNC machine.
- **Precision Control**: Use stepper motors, calibration, and feedback systems for accurate X, Y, and Z axis movement.
- **User Interface:** Create an easy-to-use setup for drawing commands using custom software or G-code applications like Processing or GRBL Controller.
- **Safety and Reliability:** Add fail-safes like limit switches, emergency stops, and heat protection to prevent damage and injury.
- **Documentation and Sharing**: Provide detailed schematics, code, and assembly instructions for community replication and modification.
- **Outreach in Education**: Use the project to teach CNC fundamentals in programming, robotics, and mechatronics.
- **Personalization:** Provide options for tool head swaps and open-source designs to encourage experimentation and creativity.
- **Cost-Effectiveness:** Use affordable, off-the-shelf components and open-source solutions to keep costs low and enhance accessibility.

1.4 Brief Methodology

Using an Arduino, 2D CNC plotting machine requires a number of stages and parts.This is a succinct methodology:

- **Design and Planning:** Determine the size, quality, and content. Use CAD software such as AutoCAD or Fusion 360 to build the structure, including the motors, rails, and frame.
- **Compiling Elements:** Arduino board, Stepper motors: three are often used for the X, Y and Z axes. Stepper motor controllers, Motor power supply, Bearings, rods, and linear rails, Tool mount or pen holder, Pulleys and belts (used to drive the axes)
- **Mechanical Assembly:** Build the frame, install linear rails and rods, attach stepper motors with belts and pulleys, and mount the tool or pen holder on the carriage.
- **Electronics Configuration:** Connect stepper motor drivers to motors, link Arduino to drivers, power everything with the appropriate source, and optionally connect limit switches to the Arduino.
- **Programming:** Install or create Arduino firmware (e.g., GRBL) to interpret G-code, control motor movement, and manage homing, speed, and acceleration. Test with short commands to ensure proper operation.
- **Testing and Calibration**: Send test designs to the CNC, calibrate acceleration and motor steps/mm, and adjust pen/tool height and pressure for optimal results.
- **Integration and Expansion:** Add a computer interface, buttons, or an LCD panel. Consider features like a vacuum table or automatic tool changing for added functionality.
- **Safety features:** Install limit switches, an emergency stop button, and overcurrent protection. Provide safety instructions and enclose the machine in a transparent casing.
- **Systems of control:** Use Arduino to decode G-code and coordinate motor movements via stepper motor drivers. Implement a user-friendly interface (e.g., LCD or computer). Ensure accuracy and reliability through thorough testing.

1.5 Implementation Schedule

Creating an implementation schedule for the Arduino base 2D CNC plotting machine involves breaking down the project into specific tasks, estimating the time required for each task, and organizing them in a logical sequence. Here's a general outline for an implementation schedule:

Week 1-2: Planning and Design

- Define project goals, specifications, and budget constraints.
- Research components and materials needed for construction.
- Finalize bill of materials (BOM) and procurement plan.

Week 3-4: Component Acquisition

Purchase necessary components and materials based on (bill of materials) BOM. Verify compatibility and quality of components upon arrival. Address any discrepancies or issues with suppliers.

Week 5-6: Mechanical Assembly

- Construct the frame and assemble a linear motion system.
- Mount stepper motors and connect to linear axes with belts and pulleys.
- Install a pen holder or tool mount onto the carriage mechanism.
- Ensure precision and stability of mechanical assembly.

Week 7-8: Electronics Setup

- Wire stepper motors to their respective drivers.
- Connect drivers to Arduino board and power supply units.
- Integrate limit switches for end-stop detection, if desired.
- Test electrical connections and troubleshoot any issues.

Week 9-10: Programming and Firmware Development

- Develop or adapt firmware for Arduino control.
- Write code to interpret G-code commands and control motor movements.
- Implement functions for homing, coordinate mapping, and speed control.
- Debug and test firmware on the CNC plotting machine.

Week 11-12: Testing and Calibration

- Conduct comprehensive testing of the CNC plotting machine.
- Calibrate motor steps/mm, acceleration, and feed rates for accuracy.
- Fine-tune pen/tool pressure and height for optimal drawing quality.
- Verify functionality and address any remaining issues.

Week 13-14: Integration and Finalization

- Integrate additional features such as control interfaces or safety features. Document assembly process, wiring diagrams, and firmware instructions.
- Finalize user manuals and safety guidelines for operators.
- Conduct final tests and quality checks before deployment.

Week 15: Deployment and User Training

Deploy the Arduino-based 2D CNC plotting machine in its intended environment. Provide user training on machine operation, safety protocols, and maintenance procedures. Address any user questions or concerns and ensure proper utilization of the machine. This schedule is a guideline and may vary based on project complexity, resources, and individual circumstances. Adjust timelines and tasks as needed to accommodate specific requirements and challenges encountered during implementation.

1.6 Structure of the Report

This report will detail the design, construction, and testing of a 2D CNC plotting machine controlled by an Arduino microcontroller. Here's a breakdown of the chapters and their content:

Introduction:

- Briefly explain Computerized Numerical Control (CNC) machines and their applications.
- Introduce the concept of a 2D CNC plotter and its functionalities.
- State the goals and objectives of your project, highlighting the use of Arduino for control.

Background and Literature Review:

- Review of existing literature and projects related to CNC machines and Arduino-based automation.
- Analysis of different design approaches, components, and software used in similar projects.
- Identification of gaps or areas for improvement in existing solutions.

Bill of Materials (BOM) and Design Considerations:

- List all the components required to build the plotter, including Arduino board, stepper motors, motor drivers, frame materials, pen holder, etc.
- Discuss the design choices made for the frame, axis movement mechanisms (belts, lead screws), and selection of motors based on torque and speed requirements.

Mechanical Construction:

- Provide a step-by-step guide on building the frame of the plotter.
- Explain the assembly of the X, Y and Z axis drive systems, including motor mounting, belt/lead screw installation, and any end-stop switches used for homing.
- Detail the design and creation of the pen holder mechanism, focusing on smooth pen movement and lifting functionality.

Electronics and Control System:

- Discuss the wiring schematics for connecting the Arduino board to stepper motor drivers and any additional sensors.
- Explain the choice of motor driver and its configuration based on the specific stepper motors used.
- Briefly describe the power supply requirements for the entire system.

Software Development:

- Introduce the programming language used for Arduino (typically C++, java).
- Explain the functionalities of the code, including stepper motor control algorithms for movement in X, Y and Z directions.
- Discuss how the code handles pen lifting and lowering commands, and potentially integrates with software for sending drawing instructions (G-code)

Testing and Calibration:

- Describe the testing procedures for individual components (motors, drivers) and the overall system functionality.
- Explain the calibration process for setting the working area (X, Y, Z) travel limits) and ensuring accurate movement based on code commands.
- Discuss potential challenges encountered during testing and how they were addressed.

Conclusion and Future Enhancements:

- Summarize the project's achievements, highlighting successful aspects of the design and construction.
- Discuss any limitations of the current design and potential areas for improvement.
- Briefly explore possibilities for future enhancements, such as adding a Z-axis for pen pressure control or incorporating different drawing tools.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

An overview of the subject and its importance is given in the introduction to the literature review for a 2D plotting machine based on Arduino. The notion of 2D plotting machines, their historical background, and the contribution of Arduino microcontrollers to their development are all introduced to the reader. Along with summarizing the goals and format of the literature study, the introduction emphasizes how crucial it is to understand the design and execution of the Arduino-based plotting machine project. The first section of the introduction sets the scene for the literature review by highlighting the significance and applicability of 2D plotting machines within the framework of maker culture and digital fabrication. In order to demonstrate how useful and relevant 2D plotting devices are in both creative and instructional contexts, it briefly tracks their growth from conventional pen plotters to contemporary Arduino-based systems. The importance of Arduino microcontrollers in the creation of 2D plotting machines is then covered in the introduction. It highlights how Arduino platforms' accessibility, cost, and versatility have made them a well-liked option for makers and hobbyists. People can experiment and play with digital design and fabrication by building and customizing their own drawing machines with Arduino-based plotting machines. The goal of this literature study is to examine and evaluate previous studies, initiatives, and materials pertaining to Arduino-based two-dimensional plotting devices. This is explained in the introduction. The review attempts to identify trends, best practices, obstacles, and possibilities in the creation and application of Arduino-based plotting machines by combining and condensing pertinent literature. Additionally, it aims to provide insights and recommendations based on past knowledge and experiences to guide the design and implementation of the Arduino-based 2D plotting machine project. Lastly, the introduction provides a quick overview of the main areas and themes that will be covered in the literature review as well as the organization of the study. An outline of Arduino-based plotting machines is provided, along with a review of pertinent studies and projects, an examination of design factors and implementation techniques, and a conversation about possible uses and future possibilities. In closing, the introduction highlights the value of the literature review in enhancing knowledge and comprehension of Arduino-based 2D plotting devices.

2.2 Related Works

1.“ DESIGN AND DEVELOPMENT OF CNC PLOTTER MACHINE ” . Patil, P., Lidhade , P., Khamkhar, S., Mane, Y., Dhole, A. and Patil, S.,

The authors of the paper "Design and Development of CNC Plotter Machine" (Patil, Lidhade, Khamkhar, Mane, Dhole, and Patil) probably want to draw attention to the steps involved in devising, planning, and putting into practice a CNC (Computer Numerical Control) plotter machine. Talking about the various stages of development, such as design considerations, hardware selection, software development, and control algorithm integration, may be the main topic. The study may also highlight the features and capabilities of the CNC plotter machine, including its potential applications in the sectors of prototype, design, and art, as well as its ease of use and capacity to reproduce digital designs on physical media with accuracy. In general, the study probably intends to provide new perspectives on the development and real-world application of CNC plotter machines, maybe addressing issues like accuracy, dependability, and affordability.

2. " Design and implementation of a microcontroller based low-cost computer numerical control (CNC) plotter using motor driver controller" Hasan, M.M., Khan, M.R., Noman, A.T., Rashid, H., Ahmed, N. and Reza, S.T., 2019, February.

As stated in Hasan, Khan, Noman, Rashid, Ahmed, and Reza's study "Design and implementation of a microcontroller-based low-cost computer numerical control (CNC) plotter using motor driver controller" (2019), the authors' main goal is probably to show off how a low-cost CNC plotter that makes use of microcontroller technology is developed and put into practice. In order to provide precise motion control for the plotter, the discussion may center on motor driver controller integration, hardware selection, and design issues. The study may also highlight the suggested CNC plotter solution's accessibility and cost, with a view to focusing on uses in small-scale production, education, and prototyping. In the context of electrical, computer, and communication engineering, the writers may also discuss the CNC plotter system's capabilities, performance, and possible drawbacks in addition to the advancement of CNC technology. The overall goal of the research is probably to highlight the affordability and usefulness of microcontroller-based CNC systems while offering insights into their design and implementation.

3. "Development of a low-cost Arduino-based CNC machine for educational purposes." Park, S., et al. (2017).

The authors of a study by Park et al. (2017) titled "Development of a low-cost Arduino-based CNC machine for educational purposes" highlight the design, build, and application of a CNC (Computer Numerical Control) machine based on Arduino microcontrollers that is especially suited for educational settings. Presumably, the main focus will be on solving the obstacles and educational demands related to teaching CNC ideas and practical abilities in engineering or technical courses. The study might emphasis features of the Arduino-based CNC machine that make it easy to use, inexpensive, and accessible for educational institutions with low funding. The writers may also go over the teaching methods, learning goals, and practical experiences made possible by the CNC machine in order to improve students' comprehension of automation, mechatronics, and CNC technology. In general, the study likely contributes to student experience and CNC machining skill development, which benefits engineering education and workforce development initiatives.

4. Sarguroh, S.S. and Rane, A.B., 2018, January. " Using GRBL-Arduino-based controller to run a two-axis computerized numerical control machine " . Sarguroh, S.S. and Rane, A.B., 2018, January.

The purpose of the study "Using GRBL-Arduino-based controller to run a two-axis computerized numerical control machine" by Sarguroh and Rane (2018) is probably to demonstrate how GRBL (G-code Real-time Boot Loader)-Arduino-based controllers can be used to run a two-axis computerized numerical control (CNC) machine. The topics of discussion might include how the GRBL firmware is implemented on Arduino microcontrollers, how sensors, motors, and other hardware components are integrated, and how the CNC machine is controlled overall. The benefits of utilizing GRBL-Arduino-based controllers, such as their affordability, ease of use, and interoperability with open-source CNC software, may also be included in the study. The performance, dependability, and possible uses of the CNC machine in a variety of fields, including manufacturing, fabrication, and prototyping, may also be covered by the writers. In the context of developing technologies and smart cities, the project likely intends to show the viability and practicality of employing GRBL-Arduino-based controllers for operating CNC machines, with a focus on two-axis systems in particular.

5 . "Low-cost Arduino-based CNC machine." Martínez, A., et al. (2017)

In the paper "Low-cost Arduino-based CNC machine," which was presented at the IEEE 15th International Conference on Industrial Informatics (INDIN) in 2017, Martínez and colleagues aim to demonstrate the feasibility and development of an inexpensive CNC (Computer Numerical Control) machine that makes use of Arduino microcontrollers. In contrast to conventional CNC systems, they want to highlight the

Arduino-based CNC machine's performance, accessibility, affordability, and openness. Additionally, by utilizing open-source hardware and software components—particularly Arduino—the researchers hope to further the democratization of CNC technology by making it more accessible to hobbyists, small businesses, and educational institutions.

6. " Design and implementation of an Arduino-based CNC system." Chen, Z., et al. (2019).

The authors of the paper "Design and implementation of an Arduino-based CNC system" (Chen et al., 2019) want to highlight the steps involved in developing, designing, and implementing a CNC (Computer Numerical Control) system that makes use of Arduino microcontrollers. The technical components of system design, such as software development, hardware configuration, and integration of control algorithms specifically designed for CNC operations, are probably the main focus. Furthermore, the research might emphasis the usefulness, affordability, and simplicity of customization of the Arduino-based CNC system by highlighting its practical applications, usability evaluation, and performance assessment across a range of industries. In general, the study is expected to provide valuable perspectives on the creation and application of Arduino-based CNC technology solutions, especially in the domains of automation, cybernetics, and systems engineering.

7. Kywe, T., Pwint, H.N.Y. and Aung, T.T.E., 2019." ARDUINO BASED AUTOMATIC MINI CNC MACHINE FOR DRAWING." Kywe, T., Pwint, H.N.Y. and Aung, T.T.E., 2019.

The authors of the study "Arduino-Based Automatic Mini CNC Machine for Drawing" (Kywe, Pwint, and Aung, 2019) most likely want to draw attention to the creation and use of an automatic mini-CNC (Computer Numerical Control) machine that uses Arduino technology and is specifically designed for drawing applications. The discussion may Centre on the small CNC machine's functioning, design, and build, with a particular emphasis on its capacity to generate drawings or sketches on its own using digital input. The technological aspects of the device, such as the integration of stepper motors, servo motors, Arduino microcontrollers, and control algorithms to produce precise and accurate drawing results, may also be covered in the study. The writers may also go over the automatic tiny CNC machine's possible uses in a variety of contexts, including art, education, and do-it-yourself projects, all while taking into account its accessibility and user-friendliness. In general, the study probably intends to demonstrate how Arduino-based solutions may automate drawing procedures and promote technological and engineering creativity.

8 . 2D Plotter Machine. Prajhwal, K.S., Sabarish, K.S., Karan, S., Subhashnan, A. and Ragunath, K., 2020.

The purpose of the study "2D Plotter Machine" by Prajhwal, Sabarish, Karan, Subhashnan, and Ragunath (2020) is probably to emphasis the features, development, and design of a 2D plotter machine. The discussion may center on the plotter machine's construction, functioning, and technical specifications. It will highlight the machine's capacity to generate two-dimensional drawings or plots from digital input. In order to get precise and accurate plotting findings, the study may also cover the integration of various components, including motors, controls, sensors, and actuators. In addition, the authors can go over the possible uses of the 2D plotter machine in disciplines like engineering, design, art, and education while taking affordability, usability, and scalability into account. Overall, the research likely aims to showcase the capabilities of the 2D plotter machine as a versatile tool for creating visual representations of digital designs or data.

2.3 Compare and Contrast

The compare and contrast section is of utmost importance in research and academic writing. This section enables researchers to clearly outline the differences and similarities between different concepts, ideas, or solutions. By presenting a comprehensive overview, it ensures that readers have a thorough understanding of the topic and can make informed decisions or draw accurate conclusions. Additionally, this section allows for critical evaluation and analysis, enabling researchers to assess the strengths, weaknesses, limitations, and advantages associated with each option. It showcases the researcher's expertise and enhances the credibility of the research. Moreover, the compare and contrast section serve as a guide for decision-making, enabling readers to select the most suitable solution or approach. It also generates further research questions, uncovering gaps in knowledge and suggesting future research directions.

Our Arduino base 2D CNC plotting machine vs Previous

When comparing our Arduino-based 2D CNC plotting machine to previous research, we aim to understand how our project differs or aligns with existing work in the field. Here's an explanation of the comparison:

Design:

- Our Arduino-based 2D CNC plotting machine may have a unique design based on our specific requirements and design considerations. This could include factors such as the size, shape, and materials used for construction.
- In contrast, previous research projects may have employed different design approaches depending on their objectives, constraints, and intended applications. They may have focused on factors such as portability, precision, or scalability.

Functionality:

- Both our machine and previous research projects aim to translate digital designs into physical drawings with precision and accuracy. They achieve this by controlling the movement of the drawing instrument along the X, Y and Z axes.
- Our machine may offer additional features or capabilities not present in previous research projects, such as automatic pen lifting, variable drawing speeds, or support for different drawing instruments.
- Conversely, previous research projects may have focused on specific functionalities or performance metrics, such as speed, accuracy, or ease of use, depending on their research goals and target audience.

Hardware Components:

- Our Arduino-based 2D CNC plotting machine likely shares common hardware components with previous research projects, such as Arduino boards, stepper motors, guide rods, and pen holders. These components form the basic infrastructure necessary for controlling the movement of the drawing instrument.
- However, our machine may incorporate specific hardware configurations or enhancements tailored to our design goals or performance requirements. These could include high-torque stepper motors for faster and more precise movement, or custom-designed pen holders for improved stability and compatibility with different drawing instruments.

Software Tools:

- Both our machine and previous research projects are likely to rely on similar software tools for programming and controlling the machine. These could include the Arduino IDE for writing and uploading code to the Arduino board, as well as libraries like Accel Stepper for controlling stepper motors.
- However, our machine may utilize custom software solutions or additional features developed specifically for our project. These could include advanced drawing algorithms, user-friendly interfaces, or compatibility with different file formats and design software.

Applications:

- Our Arduino-based 2D CNC plotting machine, like previous research projects, can be used for a wide range of applications, including art, design, education, and prototyping. Both systems enable users to create intricate designs and artworks with ease.
- However, our machine may be optimized for specific applications or use cases based on our design objectives and target audience. This could involve customization of hardware, software, or functionality to better meet the needs of our intended users.

- In summary, comparing our Arduino-based 2D CNC plotting machine to previous research helps us understand how our project fits into the existing landscape of work in the field. It allows us to identify areas of innovation, differentiation, and potential collaboration, while also informing our design decisions and project direction.

2.4 Summary

The review summary for an Arduino-based 2D plotting machine distills the main findings and insights from the reviewed literature into a concise overview. It encompasses key research themes, methodologies, findings, and implications relevant to the development and application of Arduino-based plotting machines.

In the review summary, researchers synthesize the wealth of information gathered from various sources, highlighting recurring themes and trends observed across the reviewed literature. This includes discussions on the historical context of plotting technology, the role of Arduino microcontrollers in projects, and the diverse applications of 2D plotting machines in art, design, education, and prototyping. Researchers outline the methodologies and approaches employed in the literature review, shedding light on how sources were identified, selected, and evaluated. They may detail the criteria used to assess the relevance, reliability, and significance of the literature, as well as any qualitative or quantitative analysis techniques utilized to synthesize findings and draw conclusions. Key findings and insights derived from the literature are succinctly summarized, emphasizing main conclusions, notable observations, and emerging trends. Researchers may highlight gaps or areas for further exploration identified during the review process, offering suggestions for future research directions or areas requiring additional investigation. The literature review summary also addresses the implications of the findings for the design, development, and utilization of Arduino-based 2D plotting machines. This includes recommendations for design considerations, potential applications, and avenues for future research and development efforts. The summary aims to provide a concise yet comprehensive overview of the existing knowledge landscape, serving as a valuable resource for informing the design and implementation of the Arduino-based plotting machine project.

CHAPTER 3

PROJECT OR SYSTEM DESIGN

3.1 Introduction

The introduction to the project or system design of an Arduino-based 2D CNC plotting machine provides an overview of the objectives, scope, and significance of the project. It sets the stage for understanding the rationale behind designing such a machine and outlines the key considerations driving the design process.

In the introduction, the project's motivation and purpose are elucidated, highlighting the need for a versatile and accessible tool for creating 2D drawings and designs. This may include addressing the limitations of manual drawing methods or the desire to explore digital fabrication techniques in educational, artistic, or prototyping contexts. Moreover, the introduction outlines the scope of the project, defining the boundaries within which the design and development efforts will be focused. This encompasses aspects such as the intended functionalities of the plotting machine, the target user audience, and the specific applications or use cases that the machine aims to address. Additionally, the introduction emphasizes the significance of the project within the broader context of digital fabrication and maker culture. It may discuss the democratization of technology through open-source hardware and software platforms like Arduino, which empower individuals to create and innovate in diverse fields. Furthermore, the introduction may touch upon the challenges and opportunities inherent in designing an Arduino-based 2D CNC plotting machine. This includes considerations related to mechanical design, electronic layout, software development, integration processes, and usability factors. It acknowledges the interdisciplinary nature of the project, requiring expertise in mechanical engineering, electronics, programming, and user interface design. Overall, the introduction provides a comprehensive overview of the project or system design for an Arduino-based 2D CNC plotting machine, laying the groundwork for subsequent discussions on the design process, methodology, findings, and implications. It serves to orient the reader to the purpose, scope, and significance of the project, fostering a deeper understanding of the motivations and objectives driving the design efforts.

3.2 System Design and Components

Designing an Arduino-based 2D CNC (Computer Numerical Control) plotting machine requires careful consideration of various aspects, including mechanical structure, motion control, electrical components, and software implementation. In this detailed explanation, I'll cover the system design of a typical Arduino-based 2D CNC plotting machine, highlighting each component's function and its role in the overall system.

1. Mechanical Structure:

The mechanical structure of the CNC plotting machine provides the foundation for the entire system and ensures precise movement of the drawing implement across the drawing surface. Here are the key components and considerations:

Frame: The frame provides structural support and stability to the machine. It's typically constructed using a Partex board and rod stand to maintain rigidity while minimizing weight.

Linear Motion Systems: These systems facilitate movement in the X and Y axes. They consist of rails, bearings, and lead screws or belts. The rails guide the movement, while the lead screws or belts translate rotational motion into linear motion.

Drawing Platform: This is the surface where the drawing is created. It should be flat and stable to ensure accurate drawings.

Drawing Implement Holder: Holds the drawing tool (such as a pen or marker) securely in place. It may include mechanisms for adjusting the tool height and pressure to accommodate different drawing media.

2. Components:

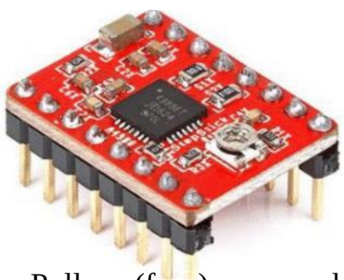
Arduino UNO: Micro controllers: ATmega328. 14 digital input/output pins (of which 6 can be used as PWM outputs) 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button.

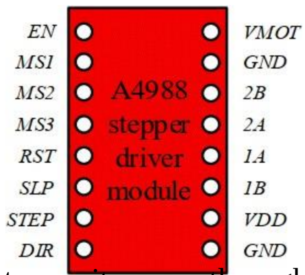
Servo motor: (SG90): The SG90 servo motor typically operates at voltages between 4.8V to 6V and offers a torque of approximately 1.8 kg/cm at 4.8V, or about 2.5 kg/cm at 6V. With a transit time of around 0.1 sec/60° at 4.8V or 0.08 sec/60° at 6V, it provides precise control over rotational movements up to 180 degrees, making it commonly used in small-scale robotics and hobbyist projects.

Stepper motor: (NAMA17): NEMA 17 Stepper motor has a 1.8° step angle. There are 200 steps or revolutions per angle. Each phase of the motor type can draw a voltage of 12V when it is in operation. As a result, the current can maintain a holding torque of 3.2 kg-cm.

Dual Digital (volt-ampere meter): A dual digital volt-ampere meter on an Arduino-based 2D CNC plotting machine monitors voltage and current using connected sensors. The Arduino reads these sensor values, converts them into readable voltage and current measurements, and displays them on the meter. This setup ensures real-time monitoring and helps detect power anomalies, enhancing the machine's safety and reliability.

Stepper Motor Driver (A4988): In an Arduino-based 2D CNC plotting machine, the A4988 stepper motor driver controls the stepper motors' movement. The Arduino sends step and direction signals to the A4988, which then regulates the current to the motor coils for precise movement. Adjustable potentiometers on the driver set the motor current limit, ensuring optimal performance. This setup enables accurate control of the CNC machine's motion.





Timing pulley: Pulleys (four)were used to transmit power through the belt via teeth and they were used to move the platform carrying the paper and the pen holder component by transmitting power from the stepper motors. Material: Aluminum (GT2 timing pulley), Number of teeth:20

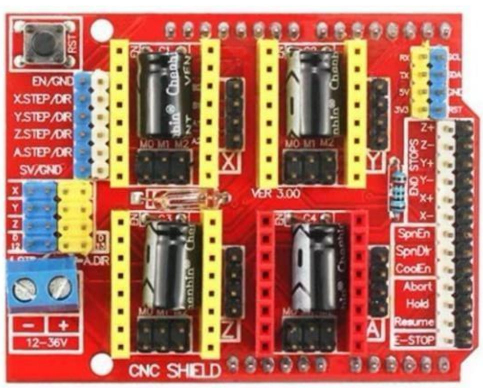
Timing belts: Timing belts typically feature a toothed profile, such as GT2 or MXL, to ensure precise motion control. They come in various widths and lengths to accommodate different applications and can transmit motion efficiently with minimal backlash. Commonly used in conjunction with timing pulleys, timing belts provide reliable power transmission in machinery and automation systems.

Steel Smooth Rods: In an Arduino-based 2D CNC plotting machine, steel smooth rods serve as linear guides for the moving components. The rods provide stability and precision for the carriage, which holds the pen or tool, ensuring accurate plotting. Bearings or linear bushings slide along these rods, allowing smooth, controlled motion. This setup enhances the machine's precision and reliability.

PVC board: It was used for our drawing platform. Thickness: 2mm , Area:8.5’’*12’’.

Power supply 12V,5A: A 12V, 5A power supply can deliver up to 60 watts of power, making it suitable for powering various electronics and small appliances. It typically features a standard barrel connector for easy connection to compatible devices and operates within a voltage range of approximately 11V to 13.8V. With a current rating of 5A, it can supply sufficient power for applications requiring moderate to high power consumption**.**

CNC shield: The Arduino CNC shield makes it simple to set up and operate that CNC project. Through the use of three pieces of stepper driver hardware—an Arduino, this shield, and a breakout board—it employs open-source firmware to operate three stepper motors. It is possible to construct a user-friendly, compact, and adaptable CNC plotter. The NC shield is powered by a 12-volt switched mode power supply, or SMPS. Every motor's current flow is regulated by the CNC shield. A stepper motor is going to move this 4x4 centimeter bed. Utilising this machine, I have produced substantial designs after elongating or fortifying the lead screw. All three-stepper motor power will be monitored by the CNC shield. Using the object's shortest path, the CNC machine.



Partex board: In an Arduino-based 2D CNC plotting machine, a Partex board serves as the sturdy base or frame. It provides a flat, stable surface for mounting all components, including motors, rods, and the plotting mechanism. The board ensures the structural integrity and alignment of the machine. This setup contributes to the precision and stability of the plotting process.

3. Motion Control:
The motion control system translates digital instructions into physical movement of the drawing implement. It involves the Arduino microcontroller, motor drivers, and motion planning algorithms:

Arduino Firmware: The code running on the Arduino microcontroller interprets G-code commands and controls the motion of the motors accordingly. Popular firmware options for Arduino-based CNC machines include GRBL (for stepper motors) and Marlin (for servo motors).

G-code Interpreter: Converts design files or instructions into a series of G-code commands that the Arduino can understand. G-code specifies the toolpath, feed rates, and other parameters for creating the drawing.

Motor Control: The Arduino sends signals to the motor drivers, which amplify these signals and deliver precise electrical pulses to the stepper motors or servo motors. This causes the motors to move the drawing implement along the specified toolpath.

4. Software Implementation:

The software components enable the CNC plotting machine to interpret digital designs or instructions and generate corresponding motion commands. Here's an overview of the key software components:

G-code: To draw a text file or design a circuit layout by the CNC plotter firstly the files need to be converted into G-code. G-Code is a set of instructions that contains a number of X, Y, and Z, coordinates depending on the file. G-Code instructs the X-axis of the machine to travel fromX1 to X2 points with a specific speed and the same is true for Y axis, but for Z axis the coordinates are fixed because only vertical up & down movements are involved. G-code is a type of function used in Numerical Control

programming language that contains the information to position a tool to do actual work. G-code is separate from M-code and T-code, codes that control the machine and tooling. Although G-code as after m is often used interchangeably with Numerical Control (NC), it is actually only one component and works in conjunction with other codes to perform the appropriate tasks. The standard version of G-code is known as RS-274D, although other versions do exist for CNC machines with compatibility issues.

Arduino IDE: The Arduino IDE software facilitates programming and controlling an Arduino-based 2D CNC plotting machine. Users write the control code in the IDE, compile it, and upload it to the Arduino board via a USB connection. The uploaded code directs the CNC's stepper motors and sensors to control movement and plotting. The IDE supports various libraries to simplify hardware control. The serial monitor feature allows for real-time data monitoring and debugging. This process enables precise and customizable operations of the CNC machine, ensuring accurate plotting and efficient performance.

Inkscape: Inkscape is used to design the plotted diagram or text. In this project by using this software G-code file of a selected image or text is created. G-code is a commonly used numerical control programming language which includes X, Y, Z coordinates.

GRBL Controller: GRBL Controller software manages the operation of an Arduino-based 2D CNC plotting machine. Here's how it works:

Installation: Load GRBL firmware onto the Arduino to interpret G-code commands.

G-code Upload: Use the GRBL Controller to send G-code files, which define the plotting paths, to the Arduino.

Communication: The software communicates with the Arduino via a USB connection.

Command Execution: GRBL firmware on the Arduino translates G-code into precise motor movements.

Motor Control: The Arduino sends signals to stepper motor drivers, moving the CNC's axes accordingly.

Real-time Monitoring: The software provides real-time feedback on machine status and position.

Adjustments: Users can adjust settings like speed and spindle control through the software. This setup enables accurate, automated plotting based on the G-code instructions.

3.3 Design Specifications. Standards and Constraints

Design specifications, standards, and constraints are critical aspects of engineering design, guiding the development process and ensuring the performance, safety, and compliance of the final product. In the context of an Arduino-based 2D CNC (Computer Numerical Control) plotting machine, these elements play a crucial role in defining the machine's capabilities, limitations, and adherence to industry standards.

1. Design Specifications:

Design specifications outline the specific requirements and performance criteria that the CNC plotting machine must meet. These specifications provide a clear framework for the design process and serve as benchmarks for evaluating the machine's performance. The following are essential design specifications for an Arduino-based 2D CNC plotting machine:

Accuracy: The CNC plotting machine must be capable of achieving precise and repeatable movements to ensure accurate drawings. Specifications for accuracy typically include tolerances for positioning and repeatability, often measured in micrometers or millimeters.

Speed: The machine should operate at a reasonable speed to minimize drawing time while maintaining accuracy. Speed specifications may vary depending on the application and complexity of the drawings.

Resolution: The resolution of the machine refers to the smallest increment of movement achievable. Higher resolution allows for finer details in drawings but may come at the expense of speed.

Drawing Size: The maximum drawing area defines the size of drawings that the machine can produce. Specifications for drawing size depend on the intended application and may vary from small-scale artwork to large-format engineering drawings.

Compatibility: The machine should be compatible with various drawing media, such as paper, cardboard, or thin metal sheets. Specifications may include maximum and minimum thicknesses for supported media.

User Interface: Design specifications for the user interface include features such as touchscreen controls, buttons, knobs, or a computer interface. The interface should be intuitive and user-friendly to facilitate ease of operation.

Power Requirements: Specifications for power requirements include voltage, current, and power consumption. The machine should be compatible with standard electrical outlets or other power sources.

Portability: Depending on the intended use, portability may be a specification. Portable machines should be lightweight and compact for easy transport and setup.

Maintenance: Specifications for maintenance include accessibility to components for cleaning, lubrication, and replacement. The machine should be designed for easy maintenance to minimize downtime.

2. Standards:

Standards are established guidelines and criteria that ensure consistency, interoperability, and compliance within a particular industry or application. Adhering to standards is essential for ensuring the quality, safety, and regulatory compliance of the CNC plotting machine. The following are relevant standards for an Arduino-based 2D CNC plotting machine:

ISO 6983 (G-code Standard): This standard specifies the G-code format used to communicate instructions to CNC machines, including CNC plotting machines. Adhering to this standard ensures compatibility with various design software and G-code generators.

ISO 230-1 (Testing Methods for CNC Machine Tools): This standard specifies testing methods for determining the accuracy and repeatability of CNC machine tools, including CNC plotting machines. Compliance with this standard ensures that the machine meets industry-accepted accuracy standards.

IEC 60601 (Medical Electrical Equipment Safety Standard): If the CNC plotting machine is intended for use in medical applications, compliance with this standard ensures electrical safety and compatibility with medical device regulations.

CE Marking (European Conformity Marking): If the CNC plotting machine is intended for sale in the European Union, it must comply with relevant EU directives and bear the CE marking. This marking indicates conformity with essential health, safety, and environmental protection requirements.

UL 61010 (Safety Requirements for Electrical Equipment): This standard specifies safety requirements for electrical equipment, including CNC machines. Compliance with UL 61010 ensures electrical safety and reduces the risk of fire, electric shock, and other hazards.

3. Constraints:

Constraints are factors that limit the design options and influence decision-making throughout the design process. These constraints may arise from various sources, including budget limitations, physical space constraints, regulatory requirements, and technological limitations. The following are common constraints that designers may encounter when developing an Arduino-based 2D CNC plotting machine:

Cost: Cost constraints influence the selection of components, materials, and manufacturing processes. Design decisions must balance performance requirements with budget limitations to ensure affordability.

Size: Physical space constraints may limit the overall size of the CNC plotting machine. Design considerations such as frame size, drawing area, and component dimensions must align with available space.

Weight: Weight constraints impact portability and structural requirements. Heavy machines may require sturdier frames and support structures, while lightweight machines may sacrifice rigidity for easier transport.

Power Supply: Power constraints dictate the maximum power consumption and voltage requirements of the machine. Design choices must ensure compatibility with available power sources and electrical infrastructure.

Material Compatibility: Constraints on material compatibility define the types of drawing media that the machine can accommodate. Design considerations such as tooling and cutting parameters must align with the properties of the selected materials.

Regulatory Compliance: Regulatory constraints encompass safety, environmental, and industry-specific regulations that govern the design, manufacturing, and use of CNC plotting machines. Compliance with relevant regulations is essential for market acceptance and legal compliance.

3.4 System Analysis or Design Analysis

System analysis involves a comprehensive examination of the components, functions, and interactions within a system to understand its requirements, constraints, and performance metrics. In the context of an Arduino-based 2D CNC plotting machine, system analysis helps identify the necessary components, their roles, and the overall system's capabilities and limitations.

1. Requirement Analysis: (Functional Requirements:)

Interpret digital designs: The CNC machine must be able to interpret digital designs or instructions, typically in the form of G-code commands, to create the desired drawings.

Control motion: It should accurately control the movement of the drawing implemented in the X, and Z axes to follow the specified toolpath.

Interface with user: The machine should have a user interface that allows users to upload design files, configure machine settings, and monitor the plotting process.

Non-functional Requirements:

Accuracy: The machine must produce drawings with high precision and repeatability to ensure the fidelity of the final output.

Speed: It should operate at a reasonable speed to minimize drawing time while maintaining accuracy.

Reliability: The machine should be reliable, with minimal downtime and consistent performance over time.

Ease of use: The user interface should be intuitive and user-friendly to facilitate ease of operation for users of varying technical backgrounds.

2. Component Analysis: (Mechanical Components:)

Frame: Provides structural support and stability to the machine. It must be rigid enough to minimize vibrations during operation.

Linear Motion Systems: Enable movement of the drawing implemented in the X and Y axes. Components include rails, bearings, lead screws, or belts.

Drawing Platform: Provides a stable surface for creating drawings. It must be flat and level to ensure accurate drawings.

Drawing Implement Holder: Secures the drawing tool, such as a pen or marker, in place during operation. It may include mechanisms for adjusting the tool height and pressure.

Electrical Components:

Arduino Microcontroller: Acts as the brain of the machine, interpreting G-code commands and generating motor control signals.

Motors: Stepper motors or servo motors drive the linear motion systems to move the drawing implement.

Motor Drivers: Convert digital signals from the Arduino into precise motor movements. Popular options include A4988 or DRV8825 for stepper motors and PWM controllers for servo motors.

Power Supply: Provides electrical power to the entire system, delivering the necessary voltage and current to operate the motors and other components.

Software Components:

G-code Interpreter: Converts digital designs or instructions into a series of G-code commands that the Arduino can understand. It specifies the toolpath, feed rates, and other parameters for creating the drawing.

Arduino Firmware: The code running on the Arduino microcontroller that interprets G-code commands and controls the motion of the motors accordingly. Popular firmware options include GRBL for stepper motors and Marlin for servo motors.

Inkscape: Inkscape is used to design the plotted diagram or text.

Processing: Processing is open-source programming language software which is used for electronic drawings.

User Interface: Allows users to interact with the machine, upload design files, configure machine settings, and monitor the plotting process. It could be implemented through a computer interface or a dedicated control panel.

3. Interactions and Interfaces:

- The Arduino microcontroller interfaces with both the mechanical and electrical components of the machine, translating digital instructions into physical movement.
- The user interface provides a means for users to interact with the machine, allowing them to upload designs, configure settings, and monitor the plotting process.
- Mechanical components interact to ensure smooth and accurate movement of the drawing implement, with linear motion systems guiding the motion and the drawing platform providing a stable surface.
- Electrical components interface with the Arduino to receive motor control signals and power the system, with motor drivers amplifying these signals and delivering precise electrical pulses to the motors.

4. Performance Analysis:

Accuracy: The machine's accuracy is influenced by factors such as motor resolution, mechanical precision, and calibration procedures. It should be capable of achieving

sub-millimeter accuracy for intricate drawings.

Speed: The machine's speed depends on factors such as motor characteristics, acceleration/deceleration settings, and drawing complexity. It should be capable of producing drawings within a reasonable timeframe while maintaining accuracy.

Reliability: The machine's reliability is essential for minimizing downtime and ensuring consistent performance over time. Factors affecting reliability include component quality, maintenance procedures, and error handling mechanisms.

Ease of Use: The user interface should be intuitive and user-friendly to facilitate ease of operation for users of varying technical backgrounds. It should provide clear instructions and feedback to users throughout the plotting process.

5. Constraints and Limitations:

Cost: Budget constraints may limit the selection of components and materials used in the machine's construction. Cost-effective solutions must be considered without compromising performance or reliability.

Size: Physical space limitations may restrict the overall size of the machine, influencing the design of the frame and other mechanical components.

Weight: Weight constraints may impact the choice of materials for the frame and other structural components, affecting portability and stability.

Power Requirements: The availability of electrical power may impose constraints on the machine's power consumption and voltage requirements.

6. Improvement Opportunities:

Enhance Accuracy: Improve mechanical precision and calibration procedures.

Increase Speed: Optimize motor control algorithms and reduce mechanical inertia.

Improve Reliability: Implement robust error handling mechanisms and preventive maintenance procedures.

3.5 Experimental Setup

Block Diagram:

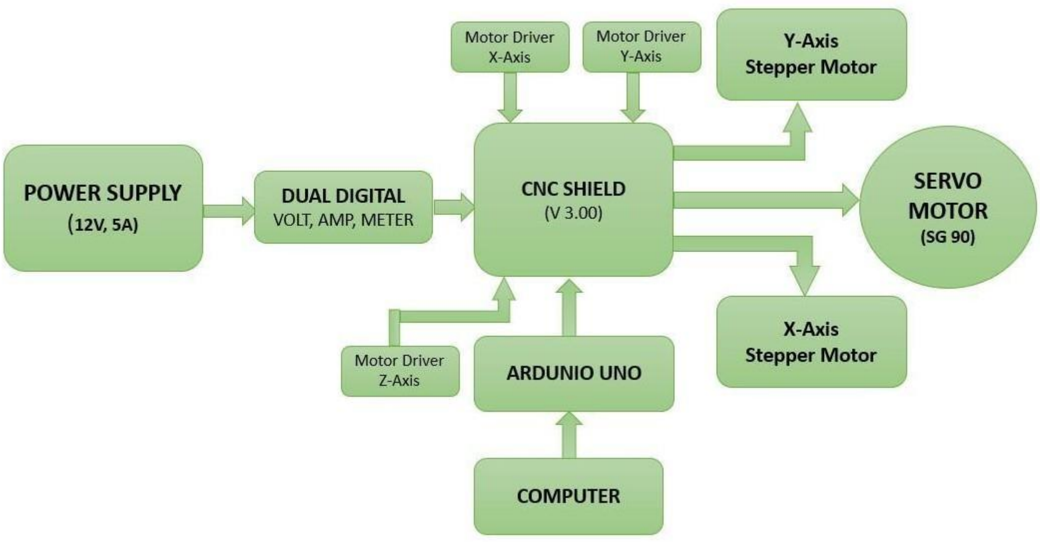


Fig:3.1: Block Diagram of Arduino Base 2D CNC Plotting Machine

Circuit Diagram:

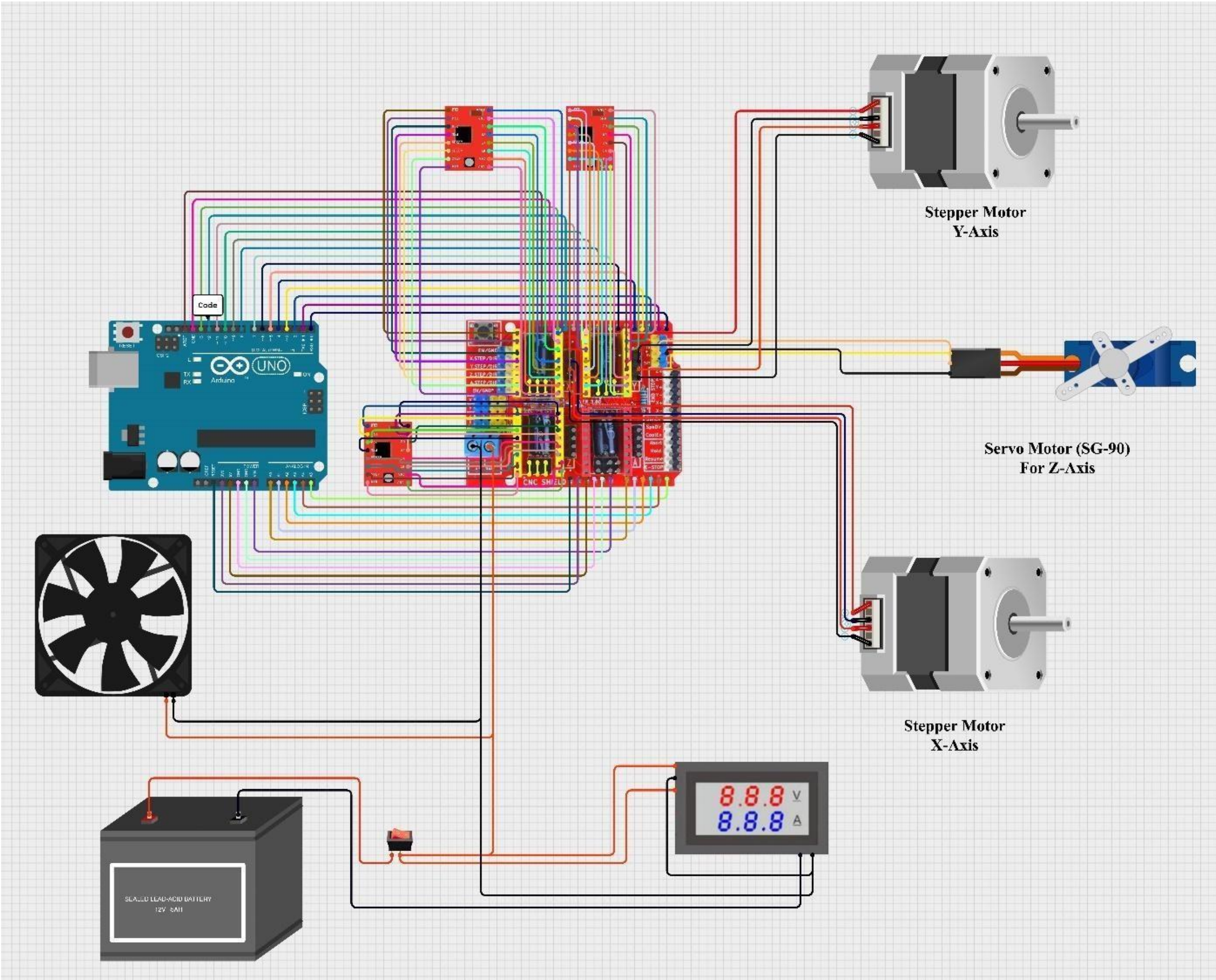


Fig:3.2: Circuit Diagram of Arduino Base 2D CNC Plotting Machine

3.6 Summary

The Arduino-based 2D CNC plotting machine is a precision drawing system integrating mechanical, electrical, and software components. Its mechanical design features a sturdy frame, precise linear motion systems, a drawing platform, and an implement holder. Electrically, it employs an Arduino microcontroller to interpret G-code commands and drive motors via motor drivers, powered by a dedicated supply. Software-wise, it utilizes a G-code interpreter and Arduino firmware for accurate toolpath execution, complemented by a user interface for seamless interaction. Performance considerations like accuracy, speed, and reliability are balanced with constraints such as cost and space limitations throughout the design process. Ultimately, this CNC plotting machine represents an efficient and versatile solution for translating digital designs into precise physical drawings.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Results

The Arduino-based 2D plotting machine represents a remarkable fusion of creativity, engineering, and technology, offering enthusiasts a platform to transform digital designs into tangible artworks. This project encompasses a comprehensive process involving mechanical assembly, electronic integration, precise programming, testing, and calibration, culminating in a versatile tool capable of executing intricate drawings with precision and finesse. At the core of the plotting machine lies its mechanical structure, meticulously crafted to provide stability and accuracy in motion. Components such as stepper motors, guide rods, pulleys, and a pen holder are strategically arranged on a sturdy frame to facilitate controlled movement along the X,Y and Z axes. The design ensures smooth and precise motion, allowing the machine to replicate intricate designs with fidelity. Electrically, the system is powered and controlled by an Arduino microcontroller, which serves as the brain of the machine. The Arduino board communicates with motor drivers, which regulate the speed and direction of the stepper motors responsible for driving the machine's movements. By sending and receiving signals, the Arduino orchestrates the intricate dance of the stepper motors, translating digital commands into physical motion with remarkable accuracy. Programming is a critical aspect of the plotting machine, as it dictates how the machine interprets and executes drawing commands. Through meticulous coding, drawing instructions are translated into precise movements of the stepper motors. Libraries such as Accel Stepper are leveraged to ensure smooth and fluid motion, enhancing the quality of the resulting drawings. The programming process involves iterative refinement, testing, and optimization to achieve the desired level of precision and performance. Testing and calibration are essential steps in fine-tuning the machine's performance and ensuring consistent results. Basic shapes and patterns are sent as commands to the machine to evaluate its accuracy and reliability. Calibration involves adjusting parameters such as step size, acceleration, and maximum speed to optimize the machine's precision and responsiveness. Calibration tools such as rulers or grid patterns are used to assess the accuracy of drawn shapes and make necessary adjustments to achieve desired results. Integration of additional features further enhances the functionality and versatility of the plotting machine. Limit switches and end stops may be incorporated to define boundaries and prevent overtravel, safeguarding the machine from potential damage. User interaction elements such as buttons or touchscreen interfaces enable intuitive control and customization options, allowing users to tailor the machine to their specific needs and preferences. Furthermore, the plotting machine offers ample opportunities for expansion and customization. Accessories such as laser engraver attachments or color capabilities can be integrated to broaden its functionality, unlocking new creative possibilities. Documentation of modifications and experiences serves as a valuable resource for the broader Arduino community, fostering collaboration and knowledge sharing among enthusiasts and makers.

In conclusion, the Arduino-based 2D plotting machine represents a captivating blend of creativity, engineering, and technology. Through meticulous craftsmanship, precise programming, and iterative refinement, enthusiasts can unleash their creativity and bring their digital designs to life in tangible form. As a testament to the boundless potential of Arduino technology, the plotting machine embodies the spirit of exploration and innovation, inspiring makers and creators worldwide to push the boundaries of what's possible.

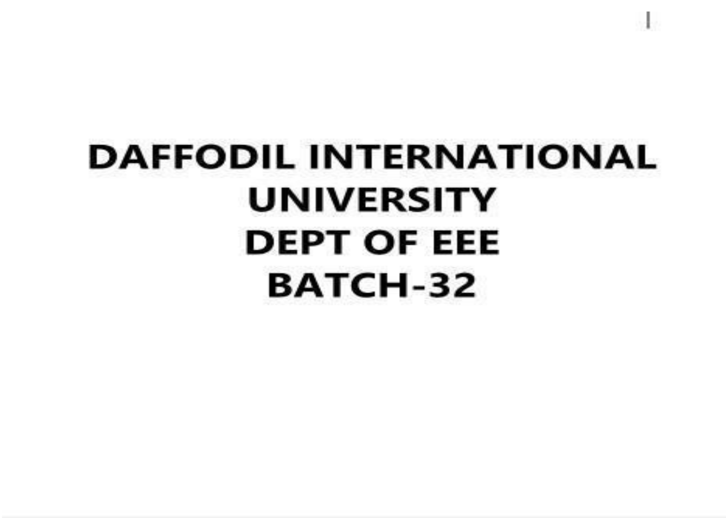


Fig: 4:1: Input Writing

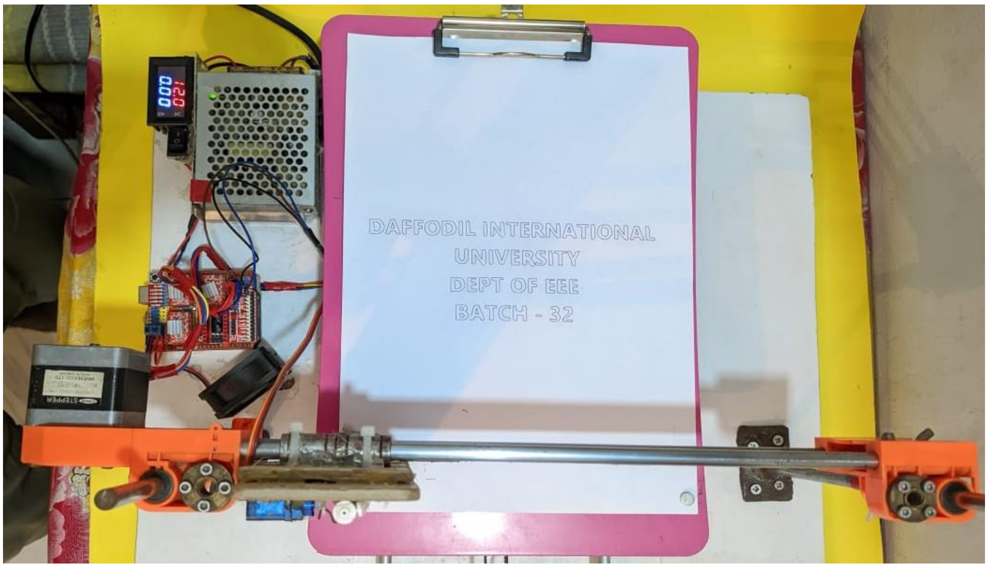


Fig: 4:2: Output Writing

Fig: 4:3: Input Writing

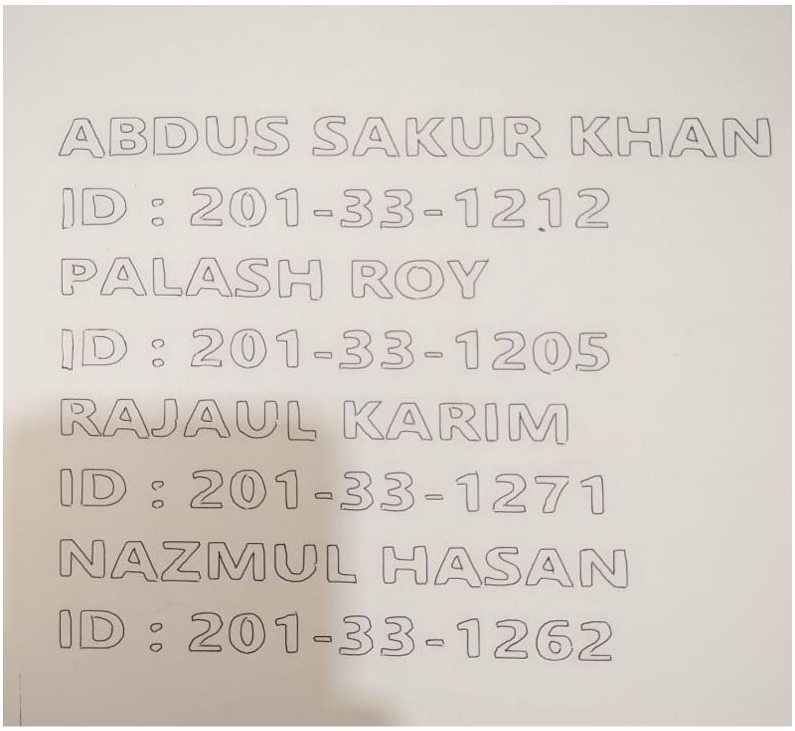


Fig: 4:4: Output Writing

Fig: 4:3: Input Writing

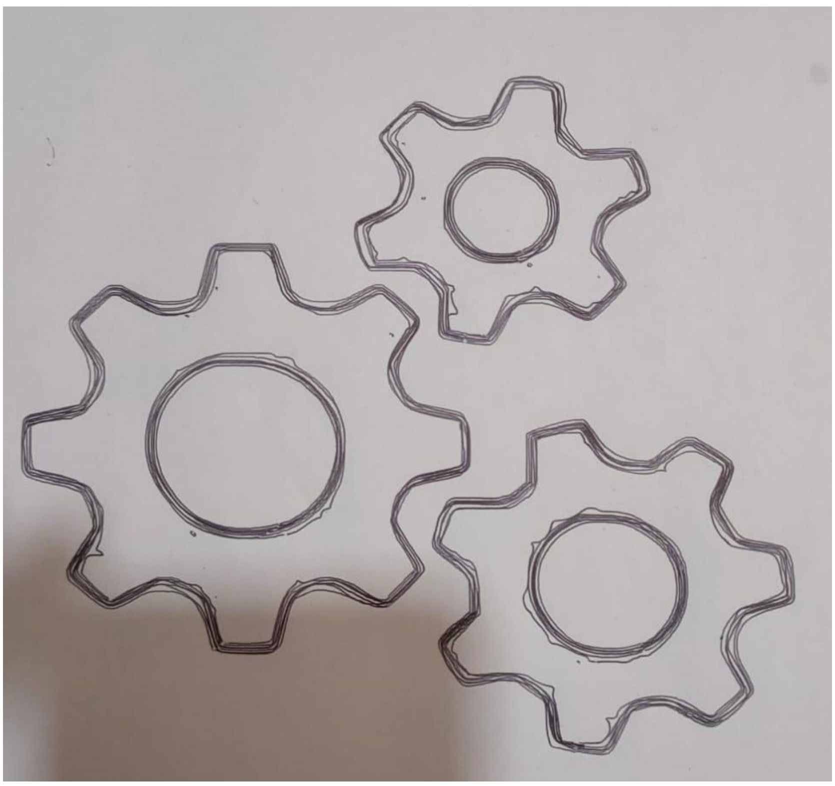


Fig: 4:4: Output Writing

4.2 Discussions

The Arduino-based 2D plotting machine stands as a testament to the intersection of creativity and technology. This project represents an opportunity to combine technical expertise with artistic vision, enabling individuals to transform digital designs into tangible works of art. Throughout this discussion, we will explore the various facets of building and utilizing an Arduino-based 2D plotting machine, from conceptualization to execution.

The concept of a 2D plotting machine harkens back to the early days of computer graphics, where pen plotters were utilized to produce intricate drawings and diagrams. These machines relied on mechanical mechanisms to control the movement of a drawing instrument, translating digital commands into physical motion on paper. While traditional pen plotters have largely been supplanted by digital printers and other imaging technologies, the concept of a 2D plotting machine persists within the DIY electronics community, serving as a testament to the enduring allure of mechanical precision and craftsmanship.

The Arduino platform has emerged as a preferred choice for constructing 2D plotting machines due to its versatility, affordability, and accessibility. Arduino microcontrollers offer a flexible platform for interfacing with stepper motors, controlling their movements, and executing drawing commands. With the right combination of hardware components and programming expertise, enthusiasts can create their own plotting machines capable of producing intricate drawings and designs with remarkable precision.

At the heart of the plotting machine lies the stepper motor, a precision actuator that enables controlled movement along the X, Y and Z axes. Stepper motors offer unparalleled accuracy and repeatability, making them ideal for applications that require precise positioning and motion control. By driving the stepper motors with carefully crafted commands, users can create intricate patterns, shapes, and designs with remarkable fidelity and detail.

The mechanical assembly of the plotting machine is a critical aspect of the build process, as it determines the machine's stability, accuracy, and reliability. The frame of the machine must be sturdy and well-constructed to withstand the rigors of operation and ensure consistent performance over time. Components such as guide rods, pulleys, and a pen holder are carefully arranged to facilitate smooth movement of the drawing instrument, ensuring that each stroke is executed with precision and finesse.

Electrically, the plotting machine is powered and controlled by an Arduino microcontroller, which serves as the brain of the machine. The Arduino board communicates with motor drivers, which regulate the speed and direction of the stepper motors responsible for driving the machine's movements. Power is supplied to the system to enable its operation, with careful attention paid to voltage and current requirements to ensure safe and reliable performance.

Programming plays a crucial role in the functionality of the plotting machine, as it dictates how the machine interprets and executes drawing commands. Through meticulous coding, drawing instructions are translated into precise movements of the stepper motors, allowing users to create intricate designs with remarkable accuracy and detail. Libraries such as

Accel Stepper are leveraged to ensure smooth and fluid motion, enhancing the quality of the resulting drawings and designs.

Testing and calibration are essential steps in fine-tuning the performance of the plotting machine, ensuring that it operates with precision and consistency. Basic shapes and patterns are sent as commands to the machine to evaluate its accuracy and reliability, with adjustments made as necessary to optimize its performance. Calibration involves adjusting parameters such as step size, acceleration, and maximum speed to ensure that the machine produces drawings that faithfully reproduce the intended designs.

Integration of additional features further enhances the functionality and versatility of the plotting machine, allowing users to tailor the machine to their specific needs and preferences. Limit switches and end stops may be incorporated to define boundaries and prevent overtravel, safeguarding the machine from potential damage. User interaction elements such as buttons or touchscreen interfaces enable intuitive control and customization options, allowing users to experiment with different drawing techniques and styles.

Furthermore, the plotting machine offers ample opportunities for expansion and customization, allowing users to unlock new creative possibilities and explore new avenues of expression. Accessories such as laser engraver attachments or color capabilities can be integrated to broaden the machine's functionality, enabling users to create a diverse range of designs and artworks. Documentation of modifications and experiences serves as a valuable resource for the broader Arduino community, fostering collaboration and knowledge sharing among enthusiasts and makers.

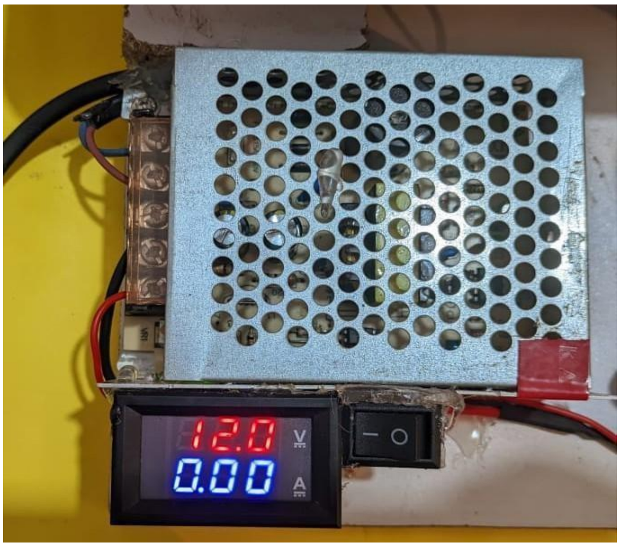
In conclusion, the Arduino-based 2D plotting machine represents a captivating blend of creativity, engineering, and technology, offering enthusiasts a unique opportunity to unleash their creativity and bring their digital designs to life in tangible form. Through meticulous craftsmanship, precise programming, and iterative refinement, users can create intricate drawings and designs with remarkable precision and detail, pushing the boundaries of what's possible with DIY electronics and maker culture. As a testament to the enduring allure of mechanical precision and craftsmanship, the plotting machine serves as a reminder of the power of human ingenuity and creativity, inspiring makers and creators worldwide to explore new frontiers of artistic expression and technological innovation.

CHAPTER 5

PROJECT MANAGEMENT

5.1 Task, Schedule and Milestones

As project members, our inspiration for this project came from a friend during a visit to a company where we encountered a CNC machine. Seeing it sparked the idea that we could create one ourselves, especially since we already had a foundational understanding of Arduino. We dedicated ourselves to making the project a success, recognizing the exorbitant cost associated with 3D CNC models and the potential effectiveness of crafting one at a lower cost in the future. Thus, we initiated our endeavor with a 2D CNC model, conducting thorough research on the topic and drawing references from open-source programs. Development time: 3.5 months, time: Weekend – 8-9 hours. The versatility of the machine can be enhanced through various adaptations. By substituting the pen with a laser, the machine can perform laser engraving and cutting tasks. Alternatively, replacing the pen with a robust drill enables milling and drilling functionalities. Introducing a stepper motor and a 3-D pen in place of the servo and pen respectively can transform it into a 3-D printer capable of producing dimensional objects. The machine's working area can be expanded by extending its axes while maintaining the algorithm intact. Integration of a multi-station tool and automatic clamping mechanism, such as the Draw bar Mechanism, can further enhance its capabilities. Additionally, Automatic Tool Changing (ATC) devices utilizing Robotic Arm Technology can streamline operations. In industrial CNC milling, plans involve upscaling the prototype by employing larger motors, reinforcing the frame and worktable with materials like



aluminum or cast iron, and enhancing the CNC control software with simulation capabilities for pre-run analysis. Furthermore, incorporating 3D printing technology into the existing hardware framework enables the fabrication of three-dimensional models.

Fig: 5.1: Connection of SMPS, dual digital volt-ampere meter and Power Switch.

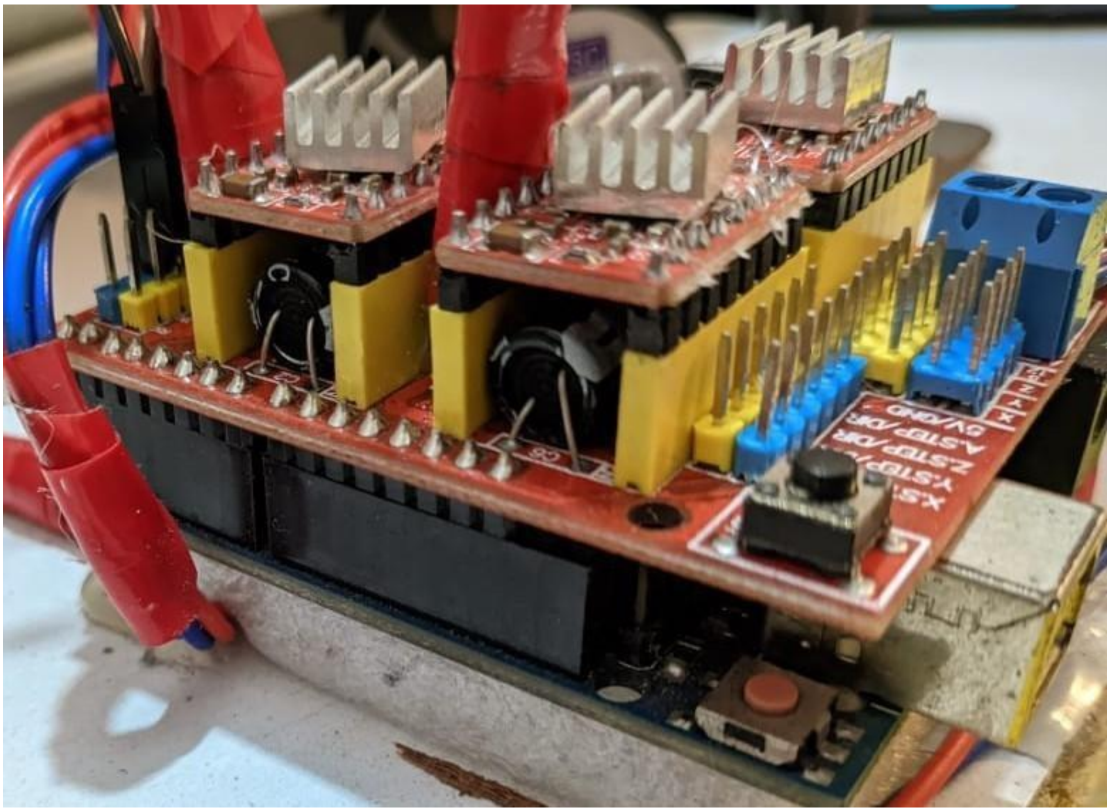


Fig: 5.2: Connection of Arduino, CNC Shield and Motor Driver Module

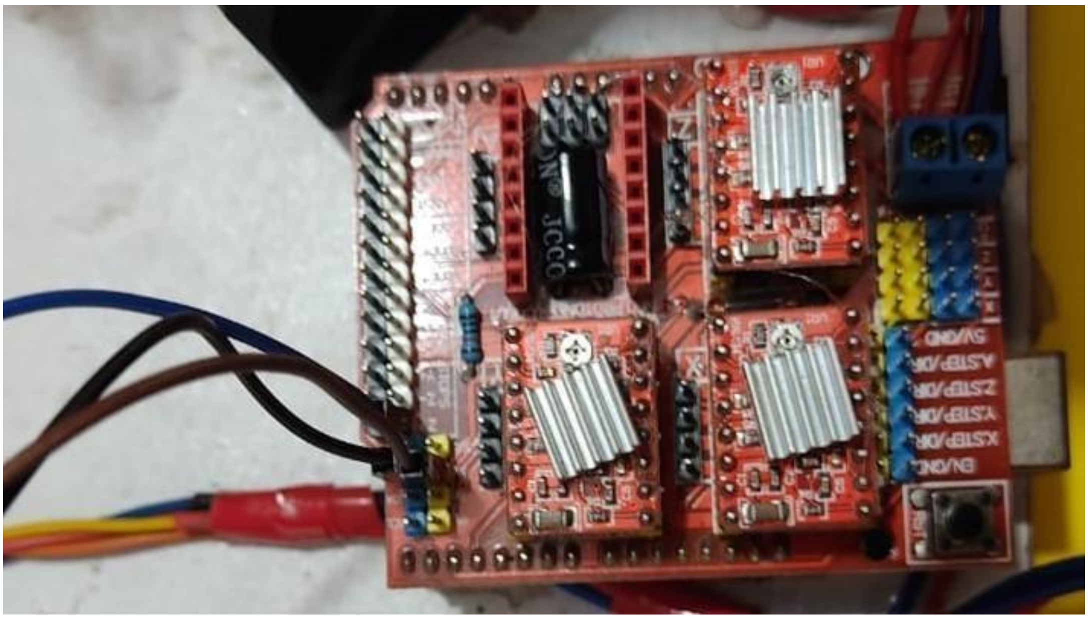


Fig: 5.3: Connection of Servo motor with CNC Shield.

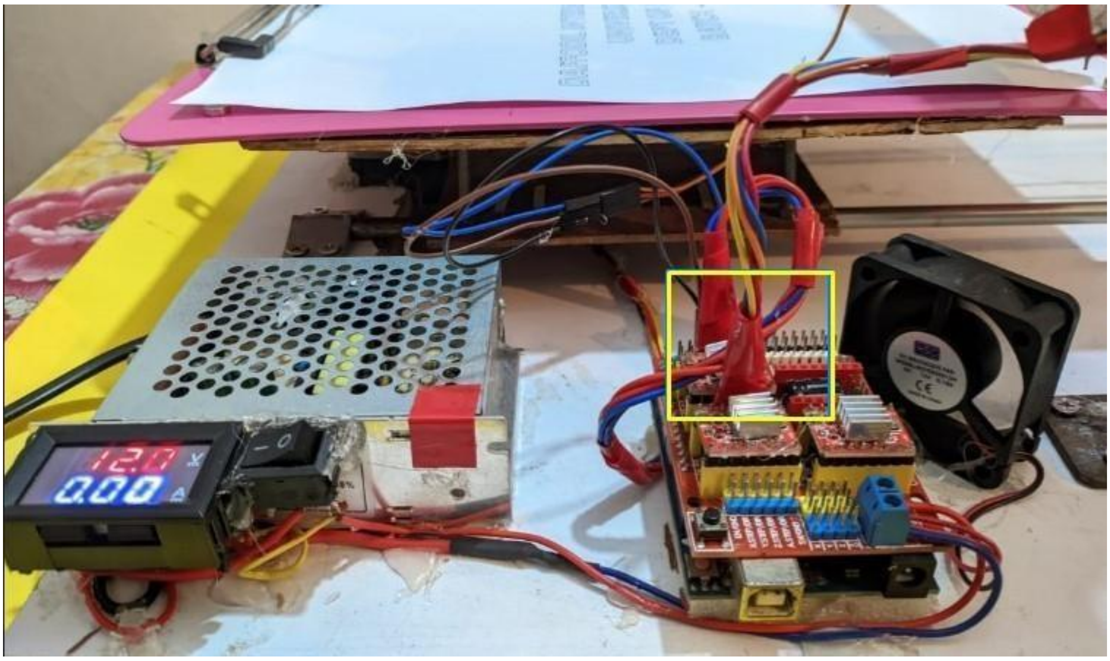


Fig: 5.4: Connection of CNC Shield X and Y Axis with Stepper Motor.

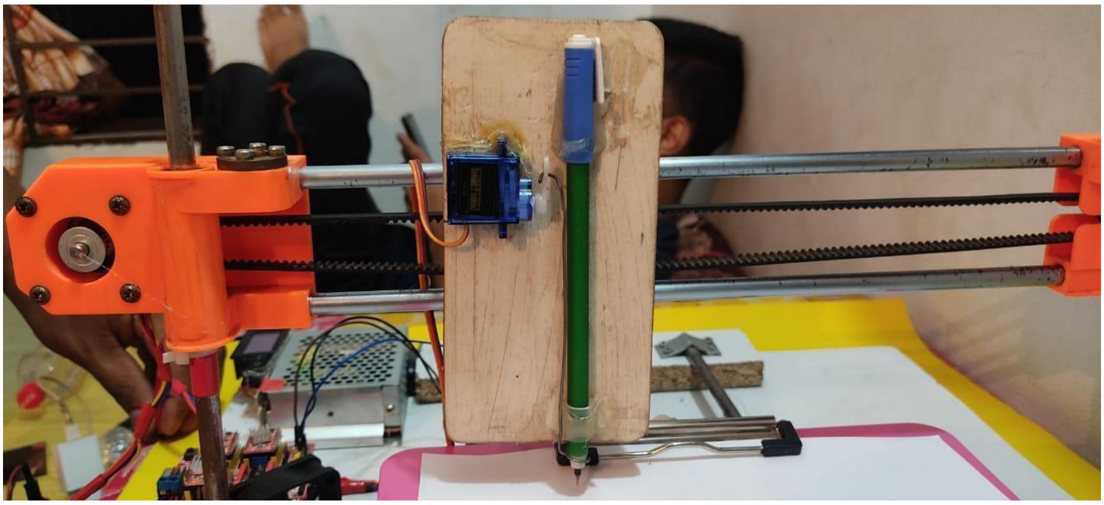


Fig: 5.5: Setup with pen holder with X-Axis.

5.2 Resources and Cost Management

Resource and cost management are essential aspects of any project, including the development of an Arduino-based 2D plotting machine. In this endeavor, resources encompass various tangible and intangible assets, while cost management involves the effective allocation and utilization of financial resources to ensure project success within budgetary constraints.

Firstly, let's delve into resource management. Resources for the Arduino-based 2D plotting machine project include hardware components, software tools, and human resources. Hardware components such as the Arduino board, stepper motors, guide rods, pulleys, pen holders, motor drivers, limit switches/end stops, and power supply form the foundation of the machine. These components must be carefully selected based on quality, compatibility, and reliability to ensure optimal performance and durability of the plotting machine.

Software tools play a crucial role in programming and controlling the Arduino-based plotting machine. The Arduino Integrated Development Environment (IDE) provides a platform for writing, compiling, and uploading code to the Arduino board. Additionally, libraries such as Accel Stepper simplify the control of stepper motors, while custom code tailored to the project's specific requirements enables the implementation of desired functionalities and features.

Human resources are equally important in the development of the plotting machine. A skilled team with expertise in hardware engineering, software development, testing, documentation, and project management is essential for successful project execution. Assigning roles and responsibilities based on team members' strengths, skills, and experience ensures efficient collaboration and communication, leading to the smooth progression of the project.

Moving on to cost management, it involves the effective allocation and utilization of financial resources to ensure project success within budgetary constraints. Developing a comprehensive budget that encompasses the costs associated with hardware components, software tools, and human resources is the first step in cost management. This budget should be carefully planned and monitored throughout the project lifecycle to ensure adherence to financial constraints and identify any deviations or overruns.

Vendor selection plays a crucial role in cost management, as choosing reliable suppliers or vendors offering competitive prices for hardware components can significantly impact project costs. Additionally, leveraging open-source or free software tools whenever possible helps reduce software licensing costs. Regular monitoring and adjustment of budget allocations based on changing project requirements, priorities, and constraints are essential to effective cost management.

Risk management is also intertwined with resource and cost management. Identifying potential risks and uncertainties that could impact the availability or utilization of project resources allows for the development of contingency plans and mitigation strategies to address these risks proactively. Proactive risk management ensures smooth progress and successful outcomes while minimizing the impact of unforeseen challenges on project resources and costs.

In conclusion, effective resource and cost management are crucial for the development of an Arduino-based 2D plotting machine. By carefully managing hardware components, software tools, human resources, budget allocations, and risks, stakeholders can optimize resource utilization, control project costs, and ensure the successful achievement of project objectives within budgetary constraints.

5.3 Lesson Learned

The development of an Arduino-based 2D plotting machine provided several valuable lessons. One prominent lesson was the importance of meticulous attention to detail in both hardware and software components. Even seemingly minor discrepancies or errors could significantly impact the functionality and performance of the machine. Therefore, thorough testing and debugging became crucial to identify and rectify any issues early on in the development process.

Another key lesson learned was the significance of scalability and modularity in design. As the project progressed, it became apparent that designing the plotting machine with scalability in mind allowed for easier integration of additional features or upgrades in the future. Modularity enabled components to be replaced or upgraded without requiring significant redesign or disruption to the overall system.

Additionally, the project highlighted the importance of documentation and knowledge sharing. Comprehensive documentation of hardware assembly, software development, and testing procedures proved invaluable for troubleshooting issues and transferring knowledge to team members. Sharing this documentation with the broader community also fostered collaboration and facilitated the exchange of ideas and insights.

Furthermore, the development process underscored the importance of patience and perseverance. Building and fine-tuning the Arduino-based 2D plotting machine required time, effort, and persistence. There were inevitably setbacks and challenges along the way, but maintaining a positive attitude and persevering through difficulties ultimately led to success.

Lastly, the project reinforced the benefits of interdisciplinary collaboration and learning. Bringing together individuals with diverse backgrounds and expertise, including hardware engineers, software developers, and project managers, enriched the project with different perspectives and ideas. Collaborative problem-solving and knowledge exchange facilitated a more holistic approach to development and enhanced the overall quality of the final product.

In conclusion, the development of an Arduino-based 2D plotting machine taught valuable lessons about attention to detail, scalability and modularity, documentation and knowledge sharing, patience and perseverance, and interdisciplinary collaboration. These lessons not only contributed to the successful completion of the project but also provided valuable insights and skills that can be applied to future endeavors in electronics, engineering, and project management.

CHAPTER 6

IMPACT ASSESSMENT OF THE PROJECT

6.1 Economical, Societal and Global Impact

The development of a 2D CNC plotting machine driven by Arduino has the potential to affect society, the economy, and the world at large. Let's examine every one of these:

Economic impact:

Accessibility and Affordability: The cost of entering the CNC technology market is significantly reduced by Arduino-based 2D CNC plotting devices. Due to their high cost, traditional CNC machines are out of reach for low-income individuals, small enterprises, and educational institutions. These devices become substantially more economical by using publicly available components and open-source technology, such as Arduino, democratizing access to sophisticated manufacturing techniques.

Innovation and Entrepreneurship: Arduino-based CNC machines' low cost encourages innovation and entrepreneurship. Prototype development, product customization, and even small-scale manufacturing operations can now be carried out by startups, small firms, and individuals without requiring a substantial initial investment. This facilitates the conversion of ideas into tangible goods and services, which promotes entrepreneurship, job creation, and economic progress.

Localized manufacture: By enabling localized manufacture, Arduino-based CNC machines lessen reliance on distant supply networks and centralized production facilities. Local production of customized parts, components, and goods by communities lowers shipping-related carbon emissions and transportation expenses. By generating jobs and promoting economic independence, this decentralization of manufacturing boosts local economies and increases resilience to shocks from around the world.

Education and Skill Development: Using and building Arduino-based CNC machines fosters the development of manufacturing, electronics, and programming skills. These devices can be included into courses at educational institutions to give students real-world experience in STEM subjects. Developing nations can attract investment in innovative manufacturing sectors and become more competitive in the global market by producing a workforce with the necessary skills to operate CNC equipment.

Impact on Society:

Education and Learning Opportunities: CNC projects with an Arduino foundation offer learning experiences to people of all ages. These devices can be used to teach design principles, mechanical engineering, electronics, and programming in elementary schools and universities. Through practical instruction, students develop their creativity, problem-solving abilities, and technological comprehension, laying the groundwork for future jobs in STEM-related disciplines.

Creativity and Innovation: Makers, artists, designers, and inventors may unleash their creativity with Arduino-based CNC machines. These machines offer a diverse platform for bringing ideas to life, whether it's creating exquisite artwork, prototyping novel products, or making custom parts. Arduino-based CNC technology democratizes access to the means of production by making manufacturing tools more accessible, enabling people to follow their passions and express themselves.

Community Building: The creation and application of CNC machines powered by Arduino create stronger bonds between neighbor's and promote teamwork. Hackerspaces, maker spaces, and do-it-yourself communities offer forums where enthusiasts can exchange knowledge, work together on projects, and encourage one another's creative endeavors. This cooperative mindset fosters diversity, inclusivity, and support for one another, resulting in thriving communities that are focused on creation and innovation.

Employment Creation and Economic Empowerment: At the local level, the use of Arduino-based CNC technology generates employment possibilities and promotes economic empowerment. These machines may be used by startups and small businesses to produce customized goods, which will boost local economies and create jobs. In order to further promote economic growth, people who have access to CNC machines can also work as independent designers, fabricators, or consultants.

Global Impact:

Localization and Resilience: CNC machines powered by Arduino encourage localized production and lessen dependency on international supply networks. These devices improve communities' ability to withstand disruptions brought on by pandemics, natural disasters, or geopolitical unrest by facilitating local production of goods. By reducing the risks connected with centralized production facilities and long-distance transportation, manufacturing is becoming increasingly decentralized, strengthening and extending supply networks.

Sustainability in the Environment: The use of CNC technology with an Arduino foundation encourages environmentally friendly production methods. These devices support environmental conservation initiatives by maximizing material utilization, decreasing waste, and using the least amount of energy possible. Furthermore, localized production lessens the carbon footprint brought on by long-distance shipping, hence lessening the negative environmental effects of international trade.

Open-Source Collaboration: Global cooperation and knowledge exchange are encouraged by the open-source nature of CNC technology based on Arduino. Global enthusiasts, scholars, and programmers foster innovation and ongoing enhancement by contributing to the creation of open-source software and hardware. This open, transparent, and collaborative attitude advances technology faster and helps communities all around the world.

Access to Resources and Technology: Advanced manufacturing equipment and resources are made available to communities and individuals in underserved places through the use of Arduino-based CNC technology. These devices enable people to create, invent, and address local problems by democratizing access to CNC technology. Communities may undergo radical change as a result of this access to technology, which can promote economic growth, skill development, and general well-being.

To summaries, the creation and implementation of 2D CNC plotting machines powered by Arduino have significant effects on the world economy, society, and environment. These machines empower people, strengthen communities, and support a more equitable, resilient, and sustainable global economy by democratizing access to manufacturing tools, encouraging creativity and innovation, establishing community, and improving localization and sustainability.

6.2 Environmental and Ethical Issues

Environmental Issues

Energy Consumption: 2D CNC plotting machines powered by Arduino are still powered by electricity, even though they are typically more energy-efficient than conventional industrial CNC machines. Energy use may have an impact on the environment, depending on usage levels and energy sources. Innovations in energy-efficient architecture and the utilization of renewable energy sources, however, help allay these worries.

Material Waste: When surplus material is taken away during CNC machining to create the desired shape, waste material is naturally produced. Environmental issues may arise from this, especially if the material being chopped is not recyclable or if the waste is not disposed of appropriately. Material waste and its environmental impact can be reduced by putting measures like recycling scrap material, cutting paths optimized and eco-friendly products to use into practice.

Chemical Pollution: In order to enhance cutting performance and extend tool life, some CNC machining procedures call for the use of lubricants or cutting fluids. However, if not managed and disposed of correctly, these fluids may contain dangerous substances that endanger the environment. The effects of chemical pollution on the environment can be lessened by putting best practices for the safe use, storage, and disposal of cutting fluids into practice.

Carbon Emissions: The production and delivery of the parts needed to assemble Arduino-based 2D CNC plotting machines might result in carbon emissions, even if the devices themselves may not produce much carbon dioxide when in use. Indirect carbon emissions may also arise if the electricity used to run these devices is produced using fossil fuels. Carbon emissions related to CNC machining can be decreased by employing energy-efficient designs, obtaining components locally, and using renewable energy sources.

Electronic Waste: Arduino-based CNC machines have a lifespan, just like any other electronic item, and may thereafter be disposed of as electronic garbage (e-waste). Hazardous substances including lead, mercury, and cadmium can contaminate the environment when e-waste is improperly disposed of. Reducing the negative effects of electronic waste on the environment can be achieved by putting into practice responsible e-waste management techniques like recycling and appropriate disposal.

Ethical Issues

Laboure Practices: The fabrication of parts for Arduino-based CNC machines may entail unethical labor practices like low pay, hazardous working conditions, and worker exploitation. Ensuring that the production process conforms to ethical standards can be achieved by sourcing components from vendors who follow transparent and ethical labor practices.

Intellectual Property Rights: Since open-source hardware and software are frequently used in Arduino-based CNC machines, there are moral questions about intellectual property rights. Open-source ideas encourage cooperation and creativity, but they also bring up issues with intellectual property ownership, crediting, and fair use. Encouraging contributors with due acknowledgement and honoring licensing agreements are two ways that the open-source community can support ethical norms.

Equity and Accessibility: Although Arduino-based CNC machines are more affordable and accessible than traditional CNC machines, some people or communities may still face obstacles to using them. Access to CNC technology may be restricted by elements including infrastructure, money, and technical expertise, which could exacerbate already-existing disparities and inequities. These ethical issues can be addressed in part by fostering accessibility, diversity, and inclusivity in maker communities and STEM education.

Environmental Responsibilities: Broader environmental responsibilities are included in the ethical analysis, which goes beyond the direct effects of CNC machining on the environment. This entails taking into account the effects of CNC machines throughout their entire lifecycle, from manufacture to disposal, and putting sustainable methods into place to reduce environmental damage. Upholding moral principles and advancing environmental sustainability can be achieved by stressing environmental stewardship and accountability in CNC machining procedures.

Safety and Risk Management: There are dangers associated with CNC machining by nature, such as mechanical, electrical, and ergonomic risks. Aside from putting risk management strategies into place to stop accidents and injuries, ethical issues also include making sure that operators and onlookers are safe. In CNC machining processes, risks can be reduced and moral standards can be upheld by following safety procedures, offering proper training, and following regulations. In summary, 2D CNC plotting machines based on Arduino provide moral and environmental concerns that need to be carefully considered and addressed proactively. The environmental impact of CNC machining can be reduced and ethical standards upheld by stakeholders through the implementation of sustainable practices, the promotion of ethical Laboure practices, the promotion of inclusivity and accessibility, and the prioritization of safety and risk management.

6.3 Utilization of Existing Standards or Codes

Arduino-based 2D CNC plotting machines utilize various existing standards and codes to ensure their safety, reliability, and compliance with industry best practices. These standards cover a wide range of aspects including electrical safety, material specifications, software development, machinery safety, environmental impact, open-source principles, regulatory compliance, quality management, and accessibility. Here's a detailed exploration of how these standards and codes are applied in the design, development, and operation of Arduino-based 2D CNC plotting machines:

Electrical Safety Standards: Since Arduino-based CNC machines use electrical circuits and parts, they must adhere to electrical safety regulations. For electrical equipment used in measurement, control, and laboratory contexts, safety criteria are outlined in standards such the International Electrotechnical Commission (IEC) 61010. Following these guidelines protects users' and operators' safety by reducing the possibility of electrical hazards.

Material and Component Standards: Components used in Arduino-based CNC machines must meet relevant industry standards to ensure quality, performance, and durability. Organizations such as the American Society for Testing and Materials (ASTM) and the International Organization for Standardization (IOS) provide specifications and testing procedures for various materials and components. Compliance with these standards ensures that CNC machines are built with high-quality, reliable components.

Software Development Standards: To guarantee dependability, security, and interoperability, software used to operate CNC machines based on Arduino must follow certain guidelines. Guiding principles for creating high-quality software products are provided by standards like the ISO/IEC 25000 series for software quality requirements and evaluation. Code stability and maintainability are ensured by adhering to coding standards, such as the MISRAC recommendations for embedded software development.

Safety Standards for Machinery: Arduino-based CNC machines are considered machinery and must comply with relevant safety standards to protect operators and bystanders from mechanical hazards. Standards such as the ISO 13849 series for the design of safety-related parts of control systems provide guidelines for assessing and mitigating risks associated with machinery operation. Adhering to these standards ensures that CNC machines are designed and operated safely.

Environmental Standards: The energy consumption, material waste, and pollutants that might result from CNC machining operations can have an influence on the environment. Sustainably produced goods are guaranteed and these effects are lessened when environmental regulations are followed. Greenhouse gas emissions from manufacturing activities can be minimized by following the recommendations provided by standards like the ISO 14000 series for environmental management.

Hardware and Software Open-Source Standards: CNC devices powered by Arduino frequently make use of open-source hardware and software, necessitating adherence to pertinent open-source licenses and standards. Guidelines for open-source hardware projects are provided by standards like the Open-Source Hardware Association (OSHWA) certification programmed. These standards cover things like attribution, licensing, and documentation requirements. Respecting these guidelines guarantees adherence to open-source ideals and encourages cooperation and creativity among makers.

Regulatory Compliance: Depending on their intended usage and region, Arduino-based CNC machines may be subject to regulatory regulations. CNC machines are guaranteed to comply with rules pertaining to safety, health, and environmental protection, such as the European Machinery Directive (2006/42/EC) or the Occupational Safety and Health Administration (OSHA) standards in the United States. Respecting legal regulations is crucial for both liability protection and market access.

Quality Management Standards: Complying with QMSs guarantees the dependability and uniformity of CNC machines powered by Arduino. Guidelines for putting in place quality management systems and continuous improvement procedures are provided by standards like the ISO 9000 series for quality management. Adherence to these guidelines aids producers in upholding the caliber of their output, fulfilling consumer demands, and improving their competitiveness in the industry.

Accessibility Standards: Accessibility should be taken into consideration throughout the design and construction of Arduino-based CNC machines to guarantee that people with disabilities may operate them. While standards like the Americans with Disabilities Act (ADA) provide principles for guaranteeing physical accessibility, standards like the Web Content Accessibility Principles (WCAG) provide guidelines for building accessible digital interfaces. By adhering to accessibility guidelines, CNC machines are made inclusive and usable by a wide variety of users.

6.4 Other Concerns

In addition to the utilization of existing standards and codes, there are several other concerns associated with Arduino-based 2D CNC plotting machines:

Accuracy and Precision: CNC plotting machines must achieve high levels of accuracy and precision in order to regularly deliver high-quality outputs. Precise movements and accurate tool paths are contingent upon calibration, appropriate maintenance, and high-quality components.

Speed and Efficiency: Maximizing production requires optimizing the speed and efficiency of CNC plotting machines. In order to reduce machining time without compromising quality, this entails striking a balance between speed and accuracy and putting effective toolpath generation algorithms into practice.

Maintenance and Upkeep: To keep CNC plotting machines functioning effectively and avoid unplanned downtime, regular maintenance is required. To maintain optimum efficiency, this entails regular inspections, lubricating moving parts, and replacing worn-out components.

Software Updates and Compatibility: It's critical to make sure that the software used to operate CNC plotting machines can produce precise tool paths and is compatible with the hardware of the device. It could be necessary to install new features, correct bugs, or boost speed with frequent software upgrades.

User Support and Training: The effective use of CNC plotting machines depends on users receiving the necessary training and continuing assistance. Users must be able to load and run programs, perform maintenance, and troubleshoot problems in an efficient manner. They must also understand how to use the equipment safely.

Material Handling and Compatibility: Proper material selection and handling are critical to the success of CNC plotting operations. Appropriate material selection, fixturing, and handling are essential since different materials have different qualities that might impact machining performance.

Cost Considerations: When making an investment in CNC plotting machines, cost is a crucial consideration. To evaluate the total affordability and return on investment, consideration must be given to elements other than the initial purchase cost, such as operating, maintenance, tooling, and training costs.

Safety Procedures and Rules: When using CNC plotting machines, it is crucial to protect both the operators and any onlookers. Minimizing the danger of accidents and injuries requires strict adherence to safety rules and regulations, which include the installation of protective barriers and emergency stop systems.

Environmental Impact: Sustainability requires minimizing the effects of CNC plotting operations on the environment. Reducing the machine's environmental impact can be achieved by using energy-efficient designs, recycling waste, and using fewer toxic chemicals.

Workflow Integration: In manufacturing or fabrication settings, CNC plotting machines must be able to function in unison with the current workflows and procedures. For seamless operation and interoperability with other tools and systems, compatibility with CAD/CAM software, data interchange formats, and communication protocols is necessary.

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

7.1 Conclusions

The Arduino-based 2D CNC plotting machine offers a flexible and approachable platform for both beginners and experts in the field of CNC (Computer Numerical Control) projects. This project provides an affordable solution for a number of tasks, such as sketching, engraving, and prototyping, by combining the adaptability of Arduino microcontrollers with the accuracy of CNC technology.

Three basic parts make up the CNC plotting machine: software, electronics, and mechanical. The building of the machine's frame, which contains the rails, pen holder, and motors, is the mechanical component. To guarantee precise and seamless operation, this frame needs to find a balance between weight and rigidity. Furthermore, the machine's precision and longevity are greatly influenced by the materials and building methods used. The Arduino microcontroller, which functions as the system's brain, is at the heart of the CNC plotting machine's electrical component. In conjunction with sensors, power supply, and stepper motor drivers, the Arduino controls the exact movements of the apparatus in response to input commands. Wiring and circuit design need to be carefully considered in order to reduce the possibility of problems like signal interference and motor overheating. Furthermore, careful consideration and testing are required because the selection of stepper motors and drivers directly affects the machine's speed, torque, and resolution. Software-wise, the CNC plotting machine interprets G-code commands and converts them into synchronized motor movements using a combination of control software and firmware. Firmware that is compatible with Arduino, like GRBL (G-code Real-time Adaptive Controller), offers a stable base for motion control and may be tailored to meet the needs of individual projects. In addition, control software that is easy to use, like Universal

G-code Sender (UGS), makes it easier for users to communicate with machines by providing functions including file management, real-time monitoring, and manual control. A number of crucial factors came to light during the design and implementation phases, influencing the CNC plotting machine's general layout and operation. First off, the machine's precision, speed, and durability are greatly influenced by the selection of mechanical parts and building methods. As a result, attaining peak performance requires careful attention to detail and fine craftsmanship. Furthermore, the machine's performance and dependability are strongly impacted by the choice of electronic parts, especially stepper motors and drivers. To guarantee smooth functioning, factors including torque, step resolution, and compatibility with the Arduino platform need to be carefully considered. Additionally, software dictates the CNC plotting machine's simplicity of use, real-time control capabilities, and compatibility with a variety of file formats. Thus, to provide a complete and user-friendly solution, developers must give equal weight to hardware design and software development. In addition, it is vital to take into account factors like scalability, maintenance, and safety to guarantee that the CNC plotting machine can fulfil the varied requirements of its customers.

Finally, for enthusiasts, hobbyists, and professionals alike, the Arduino-based 2D CNC plotting machine presents an alluring combination of accessibility, versatility, and performance. Through the utilization of Arduino microcontrollers, precise stepper motors, and user-friendly software interfaces, this project enables users to accurately and efficiently realize their imaginative ideas. Ongoing improvement and optimization, together with close attention to mechanical, electrical, and software design issues, are necessary for a successful implementation, nonetheless. The Arduino-based CNC plotting machine has the potential to revolutionize a number of sectors and inspire a new generation of makers and entrepreneurs with continuing invention and community engagement.

7.2 New Skills and Experiences Learned

The process of building a 2D CNC plotting machine powered by Arduino is a life-changing opportunity to learn new skills and gain invaluable experiences. People gain competence in programming, troubleshooting, electronics integration, mechanical assembly, and project management throughout the process. Mechanically speaking, people learn how to design and build the CNC machine's frame structure while accounting for rigidity and weight distribution. While assembling components, they hone their precision and workmanship with the use of tools like drills, saws, and measuring devices. Additionally, overcoming obstacles like synchronizing linear motion systems develops problem-solving skills and attention to detail.

People are introduced to component integration and circuit design through the project's electronic component. Students gain knowledge of how to wire and assemble parts such as power supply, sensors, drivers, and stepper motors. Solving electrical problems, such as signal interference or overheating motors, improves flexibility and problem-solving abilities. Comprehending notions such as voltage, current, and impedance enhances one's understanding of electronic systems.

The Arduino microcontroller's programming is essential to the operation of the CNC machine. Learning to write code in languages like Java and C/C++ allows people to control machines in real time and customize their behaviors. They are introduced to motion planning methods through working with firmware like GRBL, which also offers insights into control theory and software-hardware interface.

People are introduced to numerical control and computer-aided manufacturing (CAM) through the creation and interpretation of G-code. Spatial reasoning and meticulous attention to detail are necessary while creating toolpaths with CAM software. Learning digital manufacturing techniques improves one's creativity and comprehension of workflows from design to production. Overcoming obstacles during the project fosters critical thinking and resilience. Whether dealing with software glitches, electrical connectivity problems, or mechanical misalignments, people learn to identify problems methodically and apply workable solutions. This practical approach to problem-solving enhances confidence and flexibility, two qualities that are critical in a variety of work environments. Project management entails organizing, organizing, and documenting the work. Important project management tasks include obtaining components, drafting a bill of materials, and recording assembly procedures. Coordination of resources and tasks, together with progress reporting, fosters the development of time management and teamwork abilities that are applicable to various projects and situations. To sum up, creating a 2D CNC plotting machine with an Arduino platform offers a thorough learning experience that covers mechanical, electrical, programming, troubleshooting, and project management abilities. Beyond technical mastery, people acquire creative, collaborative, and problem-solving skills that are useful in a variety of disciplines. This project encourages innovation and a mindset of constant learning in addition to increasing knowledge.

7.3 Future Recommendations

In the future, Arduino-based 2D CNC plotting machines can be improved and enhanced in a number of ways as technology advances and user needs change. These suggestions address potential drawbacks and new developments in digital fabrication, with the goal of improving performance, usability, and versatility:

Enhanced Precision and Accuracy: To raise the general precision and accuracy of the CNC machine, spend more money on higher-quality mechanical parts such ball screws, high-resolution stepper motors, and precision linear rails. Adding feedback mechanisms or closed-loop control systems can help reduce errors and increase positioning precision. Expanded Material Compatibility: Make changes to the machine's design so that it can cut, engrave, or draw a greater variety of materials. This can entail adding tool changers, additional tool heads appropriate for various materials including ceramics, plastic, metal, and even wood, as well as customizable spindle speeds.

Integration of Advanced Control Algorithms: To maximize machining performance, minimize vibrations, and shorten cutting times, investigate the integration of advanced control algorithms such adaptive feed-rate control and PID (Proportional-Integral-Derivative) control. These algorithms have the potential to improve surface finish quality and overall efficiency.

Integration of Advanced Control Algorithms: To maximize machining performance, minimize vibrations, and shorten cutting times, investigate the integration of advanced control algorithms such adaptive feed-rate control and PID (Proportional-Integral-Derivative) control. These algorithms have the potential to improve surface finish quality and overall efficiency. Closed-loop control of the machine's movements can be achieved by integrating feedback sensors, such as encoders or optical sensors. This results in increased precision and repeatability by guaranteeing more precise positioning and compensating for any deviations or mistakes detected during operation.

User-Friendly Software and Interface: Provide a software package and interface that makes it easier to import designs, generate toolpaths, and operate the machine. To improve user experience and expedite productivity, include tools like real-time monitoring, simulation capabilities, and automatic toolpath optimization.

Wireless Connectivity and Remote Control: To facilitate remote control and monitoring of the CNC machine, incorporate wireless connectivity options like Bluetooth or Wi-Fi. This enables users to remotely run the machine, wirelessly submit designs, and receive alerts or status updates on their PCs or mobile devices.

Scalability through Modular Design: Create a CNC machine with a modular architecture that facilitates simple expansion or modification. This could entail using modular tool heads, replaceable parts, or expansion slots to allow for the installation of more features or accessories as needed.

Energy Efficiency and Sustainability: Use features like variable speed drives, regenerative braking systems, or power-saving modes to optimize the machine's energy efficiency. If you want to reduce your impact on the environment, think about utilizing eco-friendly products and manufacturing techniques.

Community Coordination and Open-Source Development: Encourage the sharing of information, concepts, and resources about Arduino-based CNC machines among makers, developers, and enthusiasts by creating a cooperative community. Promote open-source software development and documentation to promote best practices adoption and innovation.

User Education and Training: To equip users with the abilities required to properly operate, maintain, and troubleshoot the CNC machine, offer thorough user education and training materials. Provide online discussion boards, courses, and workshops to encourage lifelong learning and skill improvement. These suggestions for the future can help Arduino-based 2D CNC plotting machines develop into more powerful, effective, and approachable digital fabrication tools that let users express their creativity and easily realis their ideas.

REFERENCES

1. Patil, P., Lidhade, P., Khamkhar, S., Mane, Y., Dhole, A. and Patil, S., DESIGN AND DEVELOPMENT OF CNC PLOTTER MACHINE.

2. Hasan, M.M., Khan, M.R., Noman, A.T., Rashid, H., Ahmed, N. and Reza, S.T., 2019, February. Design and implementation of a microcontroller based low cost computer numerical control (CNC) plotter using motor driver controller. In 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE) (pp. 1-5). IEEE. Briefly state what he intends to highlight in this study.

3. Park, S., et al. (2017). "Development of a low-cost Arduino-based CNC machine for educational purposes." 2017 IEEE Global Engineering Education Conference (EDUCON).

4. Sarguroh, S.S. and Rane, A.B., 2018, January. Using GRBL-Arduino-based controller to run a two-axis computerized numerical control machine. In 2018 International Conference on Smart City and Emerging Technology (ICSCET) (pp. 1-6). IEEE.

5. Martínez, A., et al. (2017). "Low-cost Arduino-based CNC machine." 2017 IEEE 15th International Conference on Industrial Informatics (INDIN).

6. Chen, Z., et al. (2019). "Design and implementation of an Arduino-based CNC system." 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC). Briefly state what he intends to highlight in this study.

7. Kywe, T., Pwint, H.N.Y. and Aung, T.T.E., 2019. ARDUINO BASED AUTOMATIC MINI CNC MACHINE FOR DRAWING. International Journal Of All Research Writings, 2(3), pp.51-58.

8. Prajhwal, K.S., Sabarish, K.S., Karan, S., Subhash nan, A. and Ragunath, K., 2020. 2D Plotter Machine.

9. Girhe, P., Yenkar, S. and Chirde, A., 2018. Arduino-based cost effective CNC plotter machine. International Journal of Emerging Technologies in Engineering Research, 6(2), pp.6-9.

10. Circuit Designer Software.

APPENDIX A

TURNITIN REPORT

ORIGINALITY REPORT

12%	10%	5%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	5%
2	Submitted to Arab American University - Jenin Student Paper	1%
3	www.coursehero.com Internet Source	1%
4	repository.sustech.edu Internet Source	1%
5	www.researchgate.net Internet Source	<1%
6	docplayer.net Internet Source	<1%
7	Submitted to Oxford Brookes University Student Paper	<1%
8	Submitted to American University of the Middle East Student Paper	<1%
9	Submitted to Ishik University Student Paper	<1%

Appendix B

Complex Engineering Problem Solving and Engineering Activities

Complex Engineering Problems (P) Solving		
	Attributes	Statement from students
P1	Range of resources	Resources for a 2D CNC plotting machine encompass hardware like stepper motors, linear guides, belts, and Arduino controllers. Software includes CAD/CAM for toolpaths, firmware, and libraries for interfacing. Access to tutorials, documentation, and online communities for support is vital.
P2	Level of interaction	The level of interaction with a 2D CNC plotting machine involves setting up the design parameters, initiating the cutting or plotting process, and monitoring the operation for any errors or issues. This can be done through software interfaces or manual control interfaces depending on the complexity of the machine and user preferences.
P3	Innovation	Innovation in 2D CNC plotting machines can involve advancements in precision, speed, automation, or versatility. This might include implementing new control algorithms for smoother motion, integrating vision systems for automatic material alignment, or designing modular systems that can adapt to various applications. Additionally, innovations in software interfaces, such as user-friendly CAD/CAM software or cloud-based control platforms, can enhance the usability and accessibility of CNC plotting technology.
P4	Consequences of society and environment	2D CNC plotting machines have both positive and negative impacts. They enable rapid prototyping and small-scale manufacturing, fostering innovation. They reduce waste by optimizing material usage but may lead to job displacement due to increased automation. Energy consumption and proper disposal of electronic waste are environmental concerns.
P5	Familiarity	Familiarity with a 2D CNC plotting machine involves understanding its components, operation, and software interfaces. Users need to be proficient in tasks like setting up designs, initiating operations, and troubleshooting issues that may arise during plotting or cutting processes.

P6	Extent of stakeholder involvement and conflicting requirements	Stakeholder involvement in a 2D CNC plotting machine includes users, designers, manufacturers, and possibly regulatory bodies. Conflicting requirements may arise between precision and speed, simplicity and complexity, or cost and performance. Balancing these needs requires collaboration among stakeholders to achieve an optimal solution.
P7	Interdependence	The interdependence of a 2D CNC plotting machine refers to how its various components, such as motors, controllers, software, and mechanical parts, rely on each other to function properly. Any change or malfunction in one component can affect the overall performance and accuracy of the machine. This underscores the importance of ensuring compatibility and proper calibration among all parts for optimal operation.

Complex Engineering Problems (P) Solving		
	Attributes	Statement from students
A1	Depth of knowledge required	Building a 2D CNC plotting machine with Arduino requires intermediate knowledge of electronics, C/C++ programming for Arduino, and mechanical assembly. You'll need to understand G-code interpretation, stepper motor control, and basic mechanical design. Familiarity with Arduino IDE, soldering, and wiring is helpful. Also, knowing how to interface components like limit switches and drivers with Arduino is necessary.
A2	Range of conflicting requirements	It takes compromise to build an Arduino-based 2D CNC plotting machine that balances competing needs like accuracy vs. speed, ease of use vs. complexity, and affordability vs. output. In order to accomplish project goals within limitations like budget, available materials, and desired functionality, optimal results require handling these trade-offs successfully.

A3	Depth of analysis required	Building a 2D CNC plotting machine requires a moderate level of analysis. You need to understand motion control, G-code interpretation, electronic interfacing, and troubleshoot system integration to ensure the machine operates accurately and efficiently.
A4	Familiarity of issues	Familiarity with issues in a 2D CNC plotting machine involves understanding potential challenges related to motion control, accuracy, G-code interpretation errors, electronic component malfunctions, mechanical alignment issues, and software bugs. Addressing these issues requires troubleshooting skills and knowledge of the machine's components and operation.
A5	Extent of applicable codes	The extent of applicable codes for a 2D CNC plotting machine encompasses various programming languages like G-code, C/C++, and possibly Python for interfacing with the Arduino controller. These codes are used for generating toolpaths, controlling stepper motors, interpreting commands, and implementing additional functionalities like speed control and homing routines.

REAL TIME WORK AND SOFTWARE FILE:

https://drive.google.com/drive/folders/1xUHDo84b1USj1VlbFWScLJPBLKnX5STe?usp=sharing

YOUTUBE VIDEO LINK:

https://www.youtube.com/watch?v=gTI8IpO-y2k

APPENDIX C

PROGRAM CODE

```
#include <Servo.h>
#include <Stepper.h>

#define LINE_BUFFER_LENGTH 512

const int penZUp = 45;     // change according to the need
const int penZDown = 0;   // change according to the need

Servo penServo;

Stepper myStepperY
Stepper myStepperX

struct point {
  float x ;
  float y ;
  float z ;
};

struct point actuatorPos;

float StepInc = 1;
int StepDelay = 0;
int LineDelay = 50;
int penDelay = 50;

float StepsPerMillimeterX = 6.0;
float StepsPerMillimeterY = 6.0;

float Xmin = 0;
float Xmax = 190;
float Ymin = 0;
float Ymax = 270;
float Zmin = 0;
float Zmax = 1;

float Xpos = Xmin;
float Ypos = Ymin;
float Zpos = Zmax;

boolean verbose = false;

void setup() {
  Serial.begin( 11225 );

  penServo.attach(penServoPin);
  penServo.write(penZUp);
```

```
  delay(200);

  myStepperX.setSpeed(1000 );
  myStepperY.setSpeed(1000);

  Serial.println("T.A.Vignesh's plotter is switched on!");
  Serial.print("X range is from ");
  Serial.print(Xmin);
  Serial.print(" to ");
  Serial.print(Xmax);
  Serial.println(" mm.");
  Serial.print("Y range is from ");
  Serial.print(Ymin);
  Serial.print(" to ");
  Serial.print(Ymax);
  Serial.println(" mm.");
}

void loop()
{
  delay(200);
  char line[ LINE_BUFFER_LENGTH ];
  char c;
  int lineIndex;
  bool lineIsComment, lineSemiColon;

  lineIndex = 0;
  lineSemiColon = false;
  lineIsComment = false;

  while (1) {

    while ( Serial.available()>0 ) {
      c = Serial.read();
      if (( c == '\n') || (c == '\r') ) {
        if ( lineIndex > 0 ) {
          line[ lineIndex ] = '\0';
          if (verbose) {
            Serial.print( "Received : ");
            Serial.println( line );
          }
          processIncomingLine( line, lineIndex );
          lineIndex = 0;
        }
        else {

        }
        lineIsComment = false;
        lineSemiColon  = false;
        Serial.println("ok");
      }
      else {
        if ( (lineIsComment) || (lineSemiColon) ) {
```

```
        if ( c == ')' )  lineIsComment = false;
      }
      else {
        if ( c <= ' ' ) {
        }
        else if ( c == '/' ) {
        }
        else if ( c == '(' ) {
          lineIsComment = true;
        }
        else if ( c == ';' ) {
          lineSemiColon = true;
        }
        else if ( lineIndex >= LINE_BUFFER_LENGTH-1 ) {
          Serial.println( "ERROR - lineBuffer overflow" );
          lineIsComment = false;
          lineSemiColon = false;
        }
        else if ( c >= 'a' && c <= 'z' ) {        // Upcase lowercase
          line[ lineIndex++ ] = c-'a'+'A';
        }
        else {
          line[ lineIndex++ ] = c;
        }
      }
    }
  }
 }
}

void processIncomingLine( char* line, int charNB ) {
 int currentIndex = 0;
 char buffer[ 64 ];                                // Hope that 64 is enough for 1 parameter
 struct point newPos;

 newPos.x = 0.0;
 newPos.y = 0.0;

 while( currentIndex < charNB ) {
  switch ( line[ currentIndex++ ] ) {              // Select command, if any
  case 'U':
   penUp();
  break;
  case 'D':
   penDown();
   break;
  case 'G':
   buffer[0] = line[ currentIndex++ ];          // /!\ Dirty - Only works with 2 digit commands

   buffer[1] = '\0';

   switch ( atoi( buffer ) ){                   // Select G command
   case 0:                                   // G00 & G01 - Movement or fast movement. Same here
```

```
  case 1:
    // /!\ Dirty - Suppose that X is before Y
    char* indexX = strchr( line+currentIndex, 'X' );  // Get X/Y position in the string (if any)
    char* indexY = strchr( line+currentIndex, 'Y' );
    if ( indexY <= 0 ) {
      newPos.x = atof( indexX + 1);
      newPos.y = actuatorPos.y;
    }
    else if ( indexX <= 0 ) {
      newPos.y = atof( indexY + 1);
      newPos.x = actuatorPos.x;
    }
    else {
      newPos.y = atof( indexY + 1);
      indexY = '\0';
      newPos.x = atof( indexX + 1);
    }
    drawLine(newPos.x, newPos.y );
    //        Serial.println("ok");
    actuatorPos.x = newPos.x;
    actuatorPos.y = newPos.y;
    break;
  }
  break;
case 'M':
  buffer[0] = line[ currentIndex++ ];        // /!\ Dirty - Only works with 3 digit commands
  buffer[1] = line[ currentIndex++ ];
  buffer[2] = line[ currentIndex++ ];
  buffer[3] = '\0';
  switch ( atoi( buffer ) ){
  case 300:
    {
      char* indexS = strchr( line+currentIndex, 'S' );
      float Spos = atof( indexS + 1);
      //          Serial.println("ok");
      if (Spos == 30) {
      penDown();
      }
      if (Spos == 50) {
        penUp();
      }
      break;
    }
  case 114:                                // M114 - Repport position
    Serial.print( "Absolute position : X = " );
    Serial.print( actuatorPos.x );
    Serial.print( "  -  Y = " );
    Serial.println( actuatorPos.y );
    break;
  default:
    Serial.print( "Command not recognized : M");
    Serial.println( buffer );
  }
```

```
    }
  }
}

void drawLine(float x1, float y1) {

  if (verbose)
  {
    Serial.print("fx1, fy1: ");
    Serial.print(x1);
    Serial.print(",");
    Serial.print(y1);
    Serial.println("");
  }
   if (x1 >= Xmax) {
    x1 = Xmax;
  }
  if (x1 <= Xmin) {
    x1 = Xmin;
  }
  if (y1 >= Ymax) {
    y1 = Ymax;
  }
  if (y1 <= Ymin) {
    y1 = Ymin;
  }

  if (verbose)
  {
    Serial.print("Xpos, Ypos: ");
    Serial.print(Xpos);
    Serial.print(",");
    Serial.print(Ypos);
    Serial.println("");
  }
  if (verbose)
  {
    Serial.print("x1, y1: ");
    Serial.print(x1);
    Serial.print(",");
    Serial.print(y1);
    Serial.println("");
  }
  //  Convert coordinates to steps
  x1 = (int)(x1*StepsPerMillimeterX);
  y1 = (int)(y1*StepsPerMillimeterY);
  float x0 = Xpos;
  float y0 = Ypos;

  //  Let's find out the change for the coordinates
  long dx = abs(x1-x0);
  long dy = abs(y1-y0);
  int sx = x0<x1 ? StepInc : -StepInc;
```

```
  int sy = y0<y1 ? StepInc : -StepInc;

  long i;
  long over = 0;

  if (dx > dy) {
    for (i=0; i<dx; ++i) {
      myStepperX.step(sx);
      over+=dy;
      if (over>=dx) {
        over-=dx;
        myStepperY.step(sy);
      }
      delay(StepDelay);
    }
  }
  else {
    for (i=0; i<dy; ++i) {
      myStepperY.step(sy);
      over+=dx;
      if (over>=dy) {
        over-=dy;
        myStepperX.step(sx);
      }
      delay(StepDelay);
    }
  }
  if (verbose)
  {
    Serial.print("dx, dy:");
    Serial.print(dx);
    Serial.print(",");
    Serial.print(dy);
    Serial.println("");
  }
  if (verbose)
  {
    Serial.print("Going to (");
    Serial.print(x0);
    Serial.print(",");
    Serial.print(y0);
    Serial.println(")");
  }
  delay(LineDelay);

  Xpos = x1;
  Ypos = y1;
}
void penUp() {
  penServo.write(penZUp);
  delay(LineDelay);
  Zpos=Zmax;
  if (verbose) {
```

```
    Serial.println("Pen up!");
  }
}
void penDown() {
  penServo.write(penZDown);
  delay(LineDelay);
  Zpos=Zmin;
  if (verbose) {
    Serial.println("Pen down.");
  }
}
```

APPENDIX D
DATA SHEET OF COMPONENTS

SERIAL NUMBER	COMPONENT NAME	COMPONENT RATING	NUMBER OF COMPONENT	COMPONENT COST
01	ARDUINO UNO R3 BASIC	5V	01	850
02	CNC SHIELD (V3)	12-36V DC POWER SUPPLY	01	220
03	STEPPER MOTOR (NAMA 17)	1.68A RATED CURRENT	02	1600
04	SERVO MOTOR (SG 90)	3.0V -7.2V	01	140
05	STEPPER MOTOR DRIVER (A4988)	MAX.V-35 V, MIN .V- 8V	02	260
06	POWER SUPPLY (SMPS)	12V, 5A	01	400
07	JUMPER WIRE	…………	AS PER REQUIRED	50
08	PARTEX BOARD	………….	AS PER REQUIRED	200
09	DUAL DIGITAL (VOLT, AMP,METER)	…………	01	100
10	TIMING PULLEY	6 MM	03	30
11	COOLING FAN	12V, 0.85A	01	50
12	DATA CABLE	………..	01	80
13	TIMING BELT	………....	02	200
14	STEEL SMOOTH ROD	…………	04	300
15	SCREW	…………	AS PER REQUIRED	50
16	PEN	…………	AS PER REQUIRED	05