

Exploring Creative applications in the Digital Age: A review of Computer Numerical Controlled Plotter

Joshua Ayanlade

Department of Mechanical Engineering
University of Ilorin
Ilorin, Nigeria
17-30gd024@students.unilorin.edu.ng

Victor Ayanwunmi

Department of Mechanical Engineering
University of Ilorin
Ilorin, Nigeria
18-30gd106@students.unilorin.edu.ng

Peter Omoniyi

Department of Mechanical Engineering
University of Ilorin
Ilorin, Nigeria
omoniyi.po@unilorin.edu.ng

TC. Jen

Department of Mechanical Engineering
Science.
University of Johannesburg
Johannesburg, South Africa
tjen@uj.ac.za

Abstract— Computer Numerical Controlled (CNC) plotters have a wide range of applications, including computer-aided drawing, architectural drawing, and printed circuit board printing. These applications can also be extended to serve person with disabilities class by printing signatures and stamps. CNC plotters are developed for diverse uses. However, cost-effectiveness remains a priority. This review examines different CNC plotters designed for various applications, investigating input data types, generating and streaming geometric code (G-code) methods, and communication modes with microcontrollers. Open-source and custom-built software is commonly used for G-code generation, streaming, and parsing. They are fast and contribute to cost reduction. However, a significant drawback is the reliance on a personal computer to run this software. Additionally, the plotting process can become cumbersome and time-consuming when the input data needs frequent modifications. In order to overcome these challenges, implementing a suitable streaming protocol between a transmitting microcontroller setup and a receiving microcontroller that is interfaced with a secure digital (SD) card module could potentially eliminate the need for streaming software. Similarly, generating polar geometric coordinates for simple shapes like circles, straight lines, inclined lines, and arcs based on user-provided parameters can reduce the dependency on G-code generation software. This approach not only saves time but also maintains cost-effectiveness.

Keywords— Automation, Computer Numerical Controlled Plotter, Engineering Drawing, Person with Disabilities, Streaming Protocol.

I. INTRODUCTION

Computer Numerical Controlled (CNC) plotters are designed for various applications, such as printing printed circuit board (PCB) designs, computer-aided drawing (CAD) drawings, electronic signatures, texts, and images. These machines benefit many users, such as person with disabilities (PWD) class, manufacturers, architects, and students. CNC plotters operate using geometric code (g-code), a programming instruction that determines the machine's movement direction, speed, and distance. The plotting process involves several steps, including preparing the input file, generating the g-code, streaming the g-code, and parsing the g-code [1].

Open-source software is commonly used for controlling CNC plotters because of their accessibility and contribution to

achieving cost-effectiveness, but they are difficult to customize for specific user needs [2]. Customized software is built for robust control of the CNC plotting operation and specific applications such as electronic signature processing, speech input reception, streaming, and parsing of g-code [3], [4]. One common challenge with using this software is the requirement of a PC or mobile phone to run the applications.

Although this software helps to reduce the cost of developing a CNC plotter, a major challenge is the need for personal computers to run the software and the need for software to generate and stream input data g-code even at the slightest changes to the dimensions of the input data, thereby making the plotting process cumbersome, time-consuming, and often restricting the user input to only images.

This literature review examines various software options used in CNC plotting processes, including open-source and custom-built. It explores the input data, g-code generation, g-code streaming, and g-code parsing methods, comparing and contrasting them. This study cut across different CNC plotters used for different applications and controlled by various controllers, including the Atmega 238, STM32, programmable logic controller (PLC), and Raspberry Pi. The paper provides suggestions on how the frequent need for CAD and g-code generation software for some simple geometries can be addressed and, by extension, the reliance on PCs for g-code streaming. A schematic circuit diagram and simulation results are included.

II. CONCEPT OF CNC PLOTTER

Computerized Numerical Control (CNC) machines revolutionize modern manufacturing processes by using codes and instructions to automate operations and produce precise parts. These instructions are translated into electrical signals that drive the machine's motors and control its movements [5]. The workflow typically begins with computer-aided design (CAD) software for creating the desired drawing, followed by computer-aided manufacturing (CAM) software to generate machine-readable code. This code, containing alphanumeric instructions represented by G and M codes, guides the machine's movements and basic electrical control functions [6].

The CNC concept extends to developing CNC plotters and three-dimensional-controlled plotting machines capable of drawing text or images on solid surfaces using a pen [8]. In the CNC plotter, a microcontroller is employed to process

logical instructions received from a computer. These instructions can be in code, text, or images and are transformed into machine language for execution by the CNC plotter [8]. A typical CNC plotter is shown in Figure 1, and the commonly used hardware components are microcontroller, stepper motors, motor drivers, CNC shield, and servo motor.

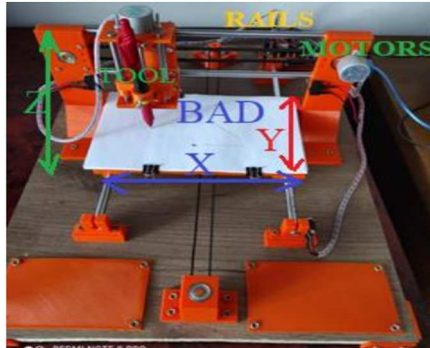


Fig.1: CNC Plotter Machine. Source:[7].

CNC plotters operate based on G-code and find applications in various plotting types, including text, images, PCB designs, and signatures. The CNC plotter workflow is shown in Figure 2. These include generating the G-code from an input file and streaming the G-code to the controller, which then generates control signals for the stepper drivers. The controller decodes the instructions received, and the stepper drivers generate the necessary pulses to move the stepper motors to the desired locations [9], [10]. Parsing software such as Grbl is often employed to decode the g-code and is uploaded to the microcontroller [1].

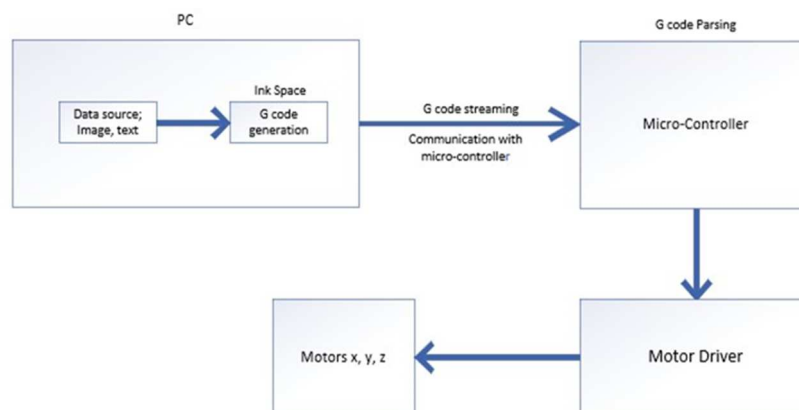


Fig 2: Workflow for a basic CNC Plotter

A. Data Source

When operating a computer numerical controlled (CNC) plotter, the first step involves inputting data into the computer [9]. This data serves as the basis for designing the desired pattern to be drawn. The input data can take various forms, including images, text, speech, electronic signatures, and computer-aided design (CAD) drawings. The specific type of data source CNC plotters require varies depending on their intended application.

Desai et al. [11] developed a CNC-based handwriting machine for person with disabilities (PWD) class, which inputs sound. The system uses Python programming and

libraries such as librosa, time, serial, and speech recognition to convert speech into text. These libraries assist in differentiating word frequencies, extracting words, and forming sentences. The use of a bidirectional Kalman filter enhances real-time speech-to-text conversion, particularly in noisy environments. This method is simpler and more robust to noise than the HMM-based speech recognition system [12]. For multilingual speech-to-text conversion, Ghadage & Shelke [13] found that utilizing the Mel-Frequency Cepstral Coefficient (MFCC) feature extraction technique along with the Minimum Distance Classifier and Support Vector Machine (SVM) methods for speech classification achieves higher accuracy compared to using only the MFCC-feature extraction technique and CDHMM-classifier. The reported accuracy for the Marathi language was 93.625%, and for English-Marathi mixed languages, it was 90.625%. To facilitate communication between deaf and blind individuals and those unfamiliar with sign language or Braille, deep learning techniques, including convolutional neural networks (CNNs), max pooling, and multiple dense layers of neural networks, can be employed. These techniques can be used to develop a sign language recognition model and a Braille recognition model, with a reported accuracy of more than 90% [14].

Software is used to prepare files for engineering drawings, images, text, and printed circuit board (PCB) designs. Khan Prince et al. [15] used Fritzing software, an open-source circuit design software, for PCB design. Fritzing offers a user-friendly interface and can automatically generate schematic circuit diagrams and PCB layouts from breadboard component connections. The PCB drawing is exported in "Etchable SVG" format and then uploaded to Inkscape software for generating the G-code. Abdalla Milad Faraj et al. [16] used Proteus software for PCB layout design. Proteus offers more functionality and a wider range of components

compared to Fritzing. Additionally, Proteus can be used for simulation to verify the correct component connections. CAD software, such as those mentioned by Kumar et al. and Panchal et al. [17], [18], is suitable for engineering drawings where precise dimensions are essential.

In addition to accessible open-source software, custom software can be created to receive digital input. An Android application with a graphical user interface was designed by Kharade et al. [6] to accept electronic signatures. Android Development Studio, the official integrated development environment for developing Android applications, was used to create the application. A quick and feature-rich emulator, a

unified environment for development for all Android devices, and the ability to make changes to an app that is already running without creating a new APK is just a few of the productivity-enhancing features that Android Studio offers [6], [19].

In summary, different data sources are used in CNC plotters, most of which are openly sourced and originally written by developers. However, these software sets still have limitations due to flexibility and adaptations to other CNC plotting applications.

B. Generation of G-code

Generating G-code is essential for CNC plotters, as it is the language they understand. Open-source or custom-developed software is commonly used for this purpose. However, in some cases, G-code may not be required, and instead, the geometric data of the input image is utilized. David et al., Hasan et al., Kamble et al. [10], [20], [21] used CAMotics, Benbox, and EleksCAM as G-code generating software for the input data. The functionalities of this software are not as robust as Inkscape.

Inkscape is a widely used free and open-source vector graphics editor employed for creating and modifying vector designs. It supports various file formats and provides extensive tools for creating, manipulating, and editing shapes, paths, and text. Inkscape can generate G-code using external plugins, effectively functioning as both "CAD and CAM" software. Its versatility is one of the reasons for its common usage. Additionally, Inkscape can help to configure microcontroller plotting parameters such as speed and steps. Abdalla Milad Faraj et al., Khan Prince et al., Kumar et al. [15]–[17] used Inkscape for generating G-code for PCB and CAD drawings. Details on how to use Inkscape are explained by Anjanah et al. [8].

Researchers like [2], [6], [22] have bypassed the G-code generating software. Kharade et al. [6] map the pixel coordinates of the electronic signature received via the GUI to the CNC plotting space, which is then used by the microcontroller to generate the motion control signals for the stepper motors. Shivakumar et al. [2] generated coordinate points using the parametric form of the geometric data, which are then fed to a motion control unit to produce corresponding signal pulses for controlling the stepper motors. However, proper synchronization using the Handshaking protocol is required due to differences in computer clock frequency, which is around 2.2 to 3.6 GHz compared to the Arduino UNO's maximum clock frequency of 16 MHz, as it can result in data loss. Additionally, decoding the data available in the serial register of the Arduino is necessary to determine the encoding type and execute the corresponding functionality.

In a recent study by Deekshitha et al. [22], they developed a standalone handwriting machine. This machine can convert various types of input data, such as audio files, speech, images, or text, into text format. The resulting text is then stored as an image, and the text coordinates of this image are generated using Python modules. Although this machine eliminates the dependency on external software and computers, the Raspberry Pi used with Arduino is more expensive than the Atmel microcontroller.

C. Streaming of G-code

Hasan et al., Zammani et al. [10], [23] utilized open-source software like Processing and EleksMaker to send control instructions to the plotter. In the case of receiving input data as sound, Ramya & Tippannavar [24] employed Benbox software, an open-source tool, to stream an image file derived from a text file. This text file was generated using a Google API designed to detect and recognize speech captured by a microphone. To achieve robust CNC control, custom-built software is used.

Nasser & Hassan [3] developed custom-built processing software with a Graphical User Interface (GUI) that allows users to choose a G-code file and send instructions. The software was built using the Visual Basic language due to its flexibility in controlling the stepper motors and the possibility of exporting it as execution files. One of the robust functionalities of this custom-built software, which is not present in Processing and CAMotics software, is the ability to send single characters such as reset, feed hold, resume, and status report, which can override ongoing operations. This gives users direct CNC control, especially during emergency periods.

The custom-built software Singh et al. [4] developed allows manual control of the CNC plotter using a joystick and sending G-code files via the GUI. An STM32F103 microcontroller was used for faster computation, and the concept of interrupt-driven programming and manipulation of register bits was employed in programming the microcontroller, which is responsible for generating pulses to move the stepper motors.

The basic features of open-source g-code software like g-code streaming and visualization, manual control, console, Log, Jogging, and Homing are adequate for simple to moderate tasks. Custom-built software will be needed if more robust control functionality is essential.

D. Communication with Microcontroller

The mode of communication with the microcontroller often depends on the application of the CNC plotter. While most CNC plotters are developed for short-range communication, others are designed for distant communication with the plotter [1]. Most wired communication utilizes a USB cable to interface the microcontroller unit with the PC. However, when the microcontroller development board lacks an onboard USB-to-Serial Converter chip, a separate converter module is needed to convert the USB data to serial data that the controller can understand. Madekar et al. [25] used an FTDI module to interface the PC with the controller, through which instructions are sent to the controller. According to [26], "FTDI" USB-serial converters can help address time synchronization issues between Gb1 and the host PC commonly encountered when using "flow control" streaming protocols.

For distant communication with the CNC plotter, wireless communication is employed. Raja [27] utilized a Bluetooth module (HC05) for wireless communication between the PC and the CNC plotter. The Bluetooth module fetches commands from the processing software running on the PC and sends them to the Arduino using the serial communication protocol. Alsalam [28] achieved longer communication

distances by connecting a PHPoC Wi-Fi shield to the microcontroller development board, which hosts the CNC software. A connection is established with online website software running on a mobile device via an IP address. This setup can be used for applications such as distance signatures.

Wired communication is more economical and reliable for applications where proximity to the plotter is not a concern. It offers lower latency and faster data transfer, especially when sending complex instructions or large amounts of data.

III. SUMMARY AND RECOMMENDATION

The CNC plotting process involves generating G-code of input data, streaming, and parsing these codes in a form the controller understands. Open-source software is commonly employed to execute this process because it is readily available and easy to use. Custom-built software is used for robust CNC plotter control and user needs. It is noteworthy that the use of this software, whether open source or custom-built for plotting simple geometries like circles, straight and inclined lines, arcs, and bounding boxes, would imply the need for the software even when the slightest modification is made to the initial design, thus making it unsuitable for real-time applications like printing signatures and stamps. Another limitation of open-source software is that it cannot be easily modified for users' needs. This is because their open-source code is mainly developed in Java or Visual Basic, which are very challenging to understand [2], while custom-built software is not accessible to all. Additionally, using open-source and custom-built software often results in a dependency on a personal computer (PC), as it is required to run the software and establish a communication connection with the controller.

To address the need for a PC in G-code streaming while keeping the associated cost low, it is recommended that a second microcontroller, such as the ATmega 2560, should be used as the transmitting controller, while the ATmega 128p can serve as the receiving controller. Since both microcontrollers operate within a compatible processing speed range, synchronization becomes unnecessary during data transmission. This alleviates a challenge commonly encountered when utilizing a PC [2].

According to Naokes [26], there are three common methods for interfacing with Grbl (an open-source firmware that can be used to control the motion of the CNC plotter). These are the simple send-response, flow control, and character-counting methods. The Send-Response method stands out as the simplest means of streaming a G-code program to Grbl [26]. This method entails the host PC sending a line of G-code to Grbl and awaiting an 'ok' or 'error' response before sending the subsequent G-code line. This protocol has been implemented between the transmitting (TX) and receiving (RX) controllers, as depicted in Figure 3, and simulated using Proteus. The result is displayed in Figure 4.

For convenient storage and access to G-code files, an SD card can be connected to the transmitting controller using an SD card module. Communication between the two controllers can be established through UART (universal asynchronous receiver-transmitter) serial communication by connecting the TX and RX pins of one microcontroller to the RX and TX pins of the other, as depicted in Figure 3. Implementing this setup would enable users to navigate through files on the SD card

and execute them. This approach helps to reduce the time associated with uploading each file, as seen in the case of open-source and custom-built software. Its applicability can be extended to people with disabilities (PWB) by integrating a microphone with the transmitting controller. This microphone would receive commands in sound, facilitating the execution of signature G-code files and similar tasks.

Similarly, the transmitting controller can be programmed to generate parametric coordinates for geometric shapes like straight lines, inclined lines, circles, and arcs [2]. These coordinates can be appended in G-code format to a named file inside the SD card, which can then be saved and called by the microcontroller for execution. This file can either be overwritten, or a new file can be created, as the case may be. A robust programming approach would involve receiving input parameters such as plotting start and stop coordinates, plotting X and Y speed, geometric dimensions, and even incorporating the ability to pause and stop the plotting operation in an emergency. IoT-enabled microcontrollers should be employed as transmitting controllers to extend their application over long distances.

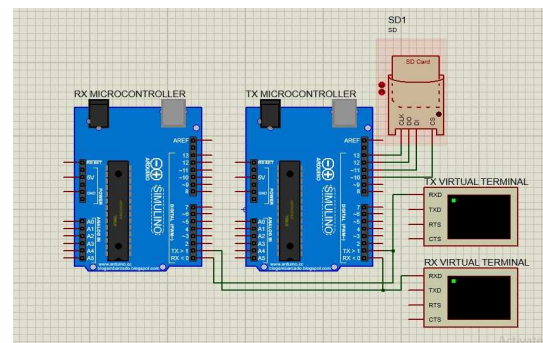


Fig 3: Micro-controller setup to replace host PC.

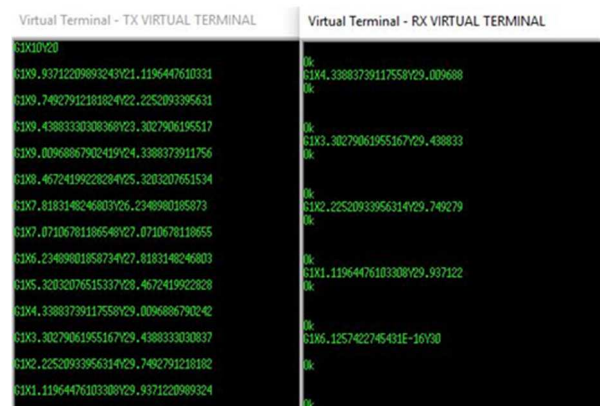


Fig 4: Simulation of Send-Response Streaming Protocol

IV. CONCLUSION.

This literature investigates various CNC plotting processes. Open-source software is commonly used to generate G-code from various input data because of its user-friendly interface and ease of access; however, its open-source code cannot be easily modified for a user's specific needs. Custom-built software is used for robust control over the plotting operation, but it is not accessible to all. Using open-source and custom-built software would often require a PC to run this software, and the overall plotting process can be

laborious and time-consuming if the input data changes over time. Thus, they are not suitable for real-time applications. An appropriate communication setup between two microcontrollers, using an SD card, implementing a suitable streaming protocol, and generating a parametric form of input data can help address the above limitations while keeping the associated costs low.

V. NEW TRENDS AND SOLUTIONS IN THE DEVELOPMENT OF LOW-COST CNC PLOTTERS.

CNC plotters have been developed for diverse applications; however, keeping the associated cost low has been the priority. Although open-source and custom-built software have been employed to achieve this, this approach is faced with the limitation of dependency on personal computers to run this software. An approach that has been put forward to eliminate this is the use of an high-end microcontroller such as the Raspberry Pi, which is programmed to generate the text coordinates of the input data and send instructions to Arduino using a pyserial connection [22]. However, the cost of the Raspberry Pi is substantial compared to the Atmega microcontroller (which is relatively cheaper, smaller in size and consumes less power). If the Atmega microcontrollers can be effectively programmed using the concept of interrupt-driven programming and manipulation of register bits [4] to generate text coordinates of input data, dependency on PCs for streaming g-code can be eliminated at a lower cost.

Efforts are also being made to extend the CNC application to person with disabilities (PWD) class by developing a plotter machine that receives voice as input and executes the writing operations of what is being said. Google speech-to-text API or library is commonly used for converting the input speech to text [11], [24]. This is an improvement to the older speech recognition technique, which requires a bidirectional Kalman filter for filtration of noise from input speech [12]. Similarly, Google language translator library can be incorporated, which can help in translating the generated text into various language scripts [24]. This is less cumbersome compared to Ghadage & Shelke's [13] Mel-Frequency Cepstral Coefficient (MFCC) feature extraction technique, used along with the Minimum Distance Classifier and Support Vector Machine (SVM) methods, for multilingual speech-to-text conversion.

REFERENCES

- [1] J. Raval, H. Sable, A. Singh, S. Shah, and S. Mhatre, "Voice Controlled Mini CNC Plotter," *SSRN Electron. J.*, May 2021, doi: 10.2139/SSRN.3867635.
- [2] T. Shivakumar, M. S. Sravan, and K. Selvajothi, "Python based 3-Axis CNC plotter," *PECON 2016 - 2016 IEEE 6th Int. Conf. Power Energy, Conf. Proceeding*, pp. 823–827, 2017, doi: 10.1109/PECON.2016.7951672.
- [3] B. Nasser and A. Hassan, "CNC Software Control System Using Visual Basic," 2020, doi: 10.1088/1757-899X/928/3/032069.
- [4] P. Singh, R. Kumar, R. Dhammi, and C. Gaur, "Design of a Low-Cost ARM-Based CNC Plotter Machine," 2021, pp. 813–827. doi: 10.1007/978-981-15-9678-0_68.
- [5] S. Raad Qasim, H. Mohammad, and M. Falah, "Accurate and Cost-Effective Mini CNC Plotter," *Int. J. Comput. Appl.*, vol. 178, no. 48, pp. 10–15, 2019, doi: 10.5120/ijca2019919370.
- [6] S. Kharade, A. Mukne, A. Kumar, and S. Rathod, "Development of novel wireless electronic signature stamp," *2020 1st IEEE Int. Conf. Meas. Instrumentation, Control Autom. ICMICA 2020*, 2020, doi: 10.1109/ICMICA48462.2020.9242825.
- [7] L. I. Culda, E. S. Muncut, and G. M. Erdodi, "Transforming 3D printed CNC prototype into a plotter," 2021, doi: 10.1088/1757-899X/1169/1/012013.
- [8] D. Anjanah, A. Kaushal, A. Gautam, and A. K. G., "CNC Plotter Machine," vol. 10, no. 11, pp. 357–360, 2022.
- [9] P. N. Chourasia, "Implementation of Low Cost Cnc Plotter Using Arduino," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 6, no. 4, pp. 20–28, 2018, doi: 10.22214/ijraset.2018.4006.
- [10] M. M. Hasan, M. R. Khan, A. T. Noman, H. Rashid, N. Ahrned, and S. T. Reza, "Design and Implementation of a Microcontroller Based Low Cost Computer Numerical Control (CNC) Plotter using Motor Driver Controller," *2nd Int. Conf. Electr. Comput. Commun. Eng. ECCE 2019*, pp. 7–9, 2019, doi: 10.1109/ECACE.2019.8679123.
- [11] Y. Desai, Y. Rungta, P. Reshamwala, and K. Joshi, "Speech enabled handwriting machine using CNC," *Proc. - 5th Int. Conf. Intell. Comput. Control Syst. ICICCS 2021*, no. Iciccs, pp. 1453–1457, 2021, doi: 10.1109/ICICCS51141.2021.9432107.
- [12] N. Sharma and S. Sardana, "A real time speech to text conversion system using bidirectional Kalman filter in Matlab," *2016 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2016*, pp. 2353–2357, 2016, doi: 10.1109/ICACCI.2016.7732406.
- [13] Y. H. Ghadage and S. D. Shelke, "Speech to text conversion for multilingual languages," *Int. Conf. Commun. Signal Process. ICCSP 2016*, pp. 236–240, 2016, doi: 10.1109/ICCSP.2016.7754130.
- [14] S. V. Shetty, A. Ram, A. Bhargav, A. Harshinee, and K. Pathak, "The Voiceless Assistant for The Specially Abled," *2023 Int. Conf. Appl. Intell. Sustain. Comput.*, pp. 1–7, 2023, doi: 10.1109/ICAISC58445.2023.10200342.
- [15] M. K. Khan Prince, M.-A. Mukaddem Ansary, and A. S. Mondol, "Implementation of a Low-cost CNC Plotter Using Spare Parts," *Int. J. Eng. Trends Technol.*, vol. 43, no. 6, pp. 333–339, 2017, doi: 10.14445/22315381/ijett-v43p256.
- [16] Abdalla Milad Faraj, Hisham Fathi Ali, Abdussalam Ali Ahmed, and Mohammed Khaled Akel, "CNC machine for image and PCB layout drawing," *Glob. J. Eng. Technol. Adv.*, vol. 11, no. 2, pp. 013–024, 2022, doi: 10.30574/gjeta.2022.11.2.0078.
- [17] M. A. Kumar, J. Krishnaraj, and R. B. G. S. Reddy, "MINI CNC 2D sketcher for accurate building drawing,," *Int. J. Civ. Eng. Technol.*, vol. 8, no. 6, pp. 543–549, 2017.
- [18] H. Panchal, M. Vaja, P. Patel, and U. Padia, "Arduino Based CNC Machine," *Int. J. Res. Sci. Eng.*, no. 3, pp. 145–148, 2017.
- [19] "Meet Android Studio | Android Developers." Accessed: Jul. 16, 2023. [Online]. Available: <https://developer.android.com/studio/intro>
- [20] K. E. David *et al.*, "Numerical Control Plotter for Direct-To-Blank Substrate Tracing of Conductive Ink for Electronic Education Purposes," *TALE 2019 - 2019 IEEE Int. Conf. Eng. Technol. Educ.*, 2019, doi: 10.1109/TALE48000.2019.9225945.
- [21] P. Kamble, S. Khoje, and J. Lele, "Implementation of Paper PCB Using Conductive Ink 2D Plotter," *Proc. - 2018 4th Int. Conf. Comput. Commun. Control Autom. ICCUBE 2018*, pp. 1–6, 2018, doi: 10.1109/ICCUBE.2018.8697781.
- [22] K. G. Deekshitha, C. Patange, M. Harshitha, and Y. J. Pavitra, "Automated Handwriting Machine," *2023 IEEE 8th Int. Conf. Conver. Technol.*, pp. 1–4, 2023, doi: 10.1109/I2CT57861.2023.10126330.
- [23] S. Zammani, M. . Mohammed, M. Abdullah, and S. Al-Zubaidi, "A New Developed Technique for Handwriting Robot," *2019 IEEE Int. Conf. Autom. Control Intell. Syst.*, no. June, pp. 889–892, 2015.
- [24] M. V. Ramya and S. S. Tippannavar, "Low-Cost Speech to Text Translation Handwriting Robot using Arduino and Google API," pp. 2–6, 2023.
- [25] K. J. Madekar, K. R. Nanaware, P. R. Phadtare, and V. S. Mane, "Automatic mini CNC machine for PCB drawing and drilling IRJET Journal Automatic mini CNC machine for PCB drawing and drilling," *Int. Res. J. Eng. Technol.*, pp. 3–7, 2016, [Online]. Available: www.irjet.net
- [26] D. Naokes, "Interfacing with Grbl · grbl/grbl Wiki · GitHub." Accessed: Jul. 15, 2023. [Online]. Available: <https://github.com/grbl/grbl/wiki/interfacing-with-grbl>
- [27] C. Raja, "Mini CNC Plotter Using Android," 2018, Accessed: Aug. 29, 2023. [Online]. Available: www.jetir.org
- [28] B. H. Y. Alsalam, "Low-cost 3D-Printed smart robotic system for distance signature and authentication," *Adv. Control Appl.*, vol. 3, no. 3, 2021, doi: 10.1002/ad2.78.