اونيۏرسيتي مليسيا ڤهڠ
UNIVERSITI MALAYSIA PAHANG

ORIGINAL ARTICLE

A Customizable Controller for 3 Axis Modular CNC Machine

Jagathesvaran A/L Veeramony[1], Muhammed Nafis Osman Zahid[1]*

[1]Faculty of Manufacturing and Mechatronic Engineering Technology, Universiti Malaysia Pahang, 26600 Pekan, Pahang, Malaysia.

ABSTRACT –. Numerical control is a useful technology to control a machine that utilize numeric values and codes. The machine comes with various sizes and built-in with its own controller. However, most of the commercial CNCs controllers are expensive and work only on the dedicated machine. With recent advancement of automation, a custom conrtoller for CN machine operation is possibly to be build. In this project, the three axis motion is controlled by Arduino Uno board with Atmega 328p microcontroller. In this case, the Arduino serves as an open source burner that burns the microcontroller using the provided programmes. Stepper motor drivers and the CNC shield accept signals from the Arduino board and transmit power to the stepper motors. Using Inkspace software, the user must first convert any image or text file into G-code before feeding it to the machine. The G-code file is then browsed in Universal G-code Sender software connected to a GRBL shield v0.9i and Arduino board through a laptop serial interface. The Universal G-code Sender programme has a G-code visualizer feature that allows to see the manufacturing processes alongside the simulation.

INTRODUCTION

The movement and operation of system gear can be controlled using computer numerical control, an advanced type of automation. A numerical control device, which uses coded layouts, digits, and characters, was developed in the nineteenth century to lessen the workload on machinists. High-creation, multiple-axle screw machine assembly has always been a time-consuming process. In any case, over the past ten years, controls for CNC multi axle screw machines have dramatically improved [1]. The technology benefits include high performance, excessive flexibility, high manufacturing rate, low manufacturing value, significantly reduced running time, and significantly reduced manufacturing losses [2]. It consists of three basic steps: gathering information, analysing and creating the toolpaths and executing the operations. Primarily based on a sequential procedure or coded commands, the machinery systems are capable to produce certain parts with distinctive sizes. After that, the programme is translated into an electric signal that is fed as input to the motors and executes the tool movements [3]. In such a way, the CNC machine system can be classified into two terms:

Non-modular system

Any machine that is complex to repair can be deemed not to be a modular system. Additionally, it is their purpose to only allow the manufacturers to perform any maintenance.

Modular system

Machine made of independent components with common interfaces. This will eventually enable designs to be upgraded, changed, mended and reused. Additionally, modular design can be advantageous to its user and they lessen the need for rework and upgrades rather than replacing the entire unit. In the meantime, it develops several customization cultures.

New industrial requirements including high precision, exceptional first-rate, excessive manufacturing rates and cheap production costs are becoming more and more important in manufacturing as a result of ongoing technological advancements. Many of these requirements, including dimensional precision, compliance to tolerances and production rate, can be satisfied with more advanced device controllers. However, there is still a lack of emphasis on the controller's role in adapting to various needs and highly configurable machines due to the sharp expansion in industry types and rising demand for small goods. Therefore, the design and configuration of the controller portion for a modular CNC machine is the main focus of this project. The brain of a CNC machine is the CNC controller. In a CNC system, a controller completes the crucial connection between a computer device and the mechanical system. The main task of the controller is to translate conditioned signals from a laptop or indexer into mechanical movement using motor output. A controller is made up of several parts, and each one works together to provide the desired motor action [4].

The number of axes, configuration and features are further parameters for CNC controllers. Rack-hooked up CNC controllers have tabs for mounting controller components on vertical rails inner a popular rack. Pendant controllers are attached to the device's arm. Pedestal controllers are mounted on top of a system-connected arm [5]. There are additional products with necessary screens, touch monitors, keypads, or touch pads. Alarms and event tracking, diskette or floppy storage, tape storage, zip disc storage, multiple software storage and simultaneous control are features for CNC controllers. Additionally, tape readers, behind tape readers (BTR), and self-diagnostic features may be included in CNC controllers [2].

METHODOLOGY

The basic concept of this project is to know how does a CNC machine works especially the G Code and controller interpretation to move in the X, Y and Z axis. In order to understand this, quite a number of books, articles, journals and research paper was referred. The invention of the modular CNC machine has made it possible to comprehend the machine's operation system quickly. Modular machines can be modified as technology advances and their requirements change. In such a case, the design of a controller for a modular CNC machine would be very beneficial and efficient. The controller is invented with of standard and customized interfaces without many complications. Furthermore, it can be upgraded depending on the requirements using different and reworked components or modules [6]. Moreover, changes can be done easily and quickly as the controller is detached from the machine. On the other side, this type of controller can be installed or 'plug and play' to a system. In addition, it can provide better consistency and accuracy that is desired in the operations [8].

Mechanical Design

There is only one type of motor used which is the stepper motor. For the X, Y and Z axis stepper motor is used because the bearing in the stepper motor can be customized and attached to the lead screw shaft. This enables movement in the clockwise and anticlockwise direction of the shaft in the X and Y axis. Moreover, for the Z axis the stepper motor can lift the tool to up and down position. Meanwhile, it requires a high torque but at minimum speed so that the rotation of the lead screw shaft is consistent and stable [9]. Furthermore, it is designed so that the motor can operate with excellent repeatability where no adjustment is required. The stepper motor can rotate approximately 360 degrees in both directions. The stepper motor does not rely on an encoder to run because we are using a closed loop stepper motor whereas the usage of the encoder can malfunction at times. Besides that, stepper motors can work without any configuration or setup where they can be connected to the drivers in order to run those [10]. X, Y, and Z axis stepper motors must be fitted to the lead screw shaft using a universal hub or coupling in order for them to rotate simultaneously with the motor according to the generated code. There is a motor driver used to connect the stepper motors and servo motors which is mounted to the Arduino at the beginning [9]. In this case, Arduino can supply 12V to the stepper motor and the data is transferred in the form of signals to the motor as it controls the lead screw rotation and movement of the tool.
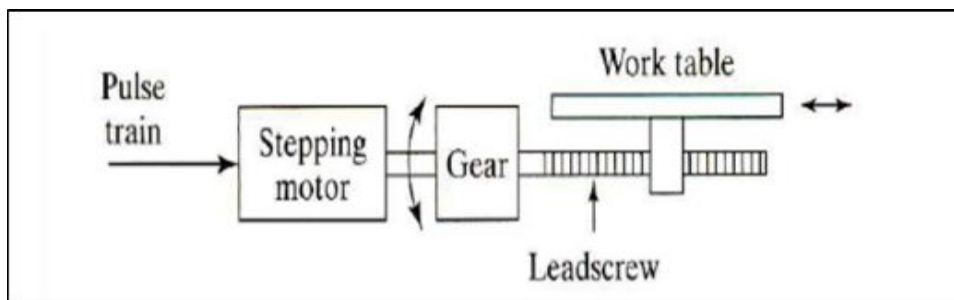

Figure 1. Stepper motor orientation in the machine

Electrical Design

Arduino is the core of a modular CNC machine as the microcontroller either drives or controls the position and movement of a stepper with the command that is derived to the program (Figure 3). As an open-source platform for prototyping, Arduino is essentially functional hardware and software [6]. Arduino enables the development of electronic software programmes and their interoperability with one another. A CNC machine's motions can be controlled by a variety of open-source programmes, such as GRBL. The GRBL programme for Arduino is simple to install and operates straight away to create a low-cost, high-performance CNC controller. G-code is the input signal utilised by GRBL, and an output signal is used by Arduino to control motion [7]. Meanwhile, it has both analog and digital I/O pins which are able to interface with disparate expansion boards, different circuit systems and microcontrollers with other assisted components together to run a program and integrate with various circuits [11].
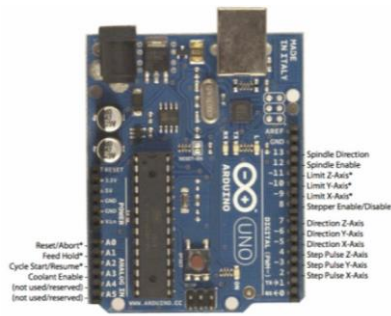
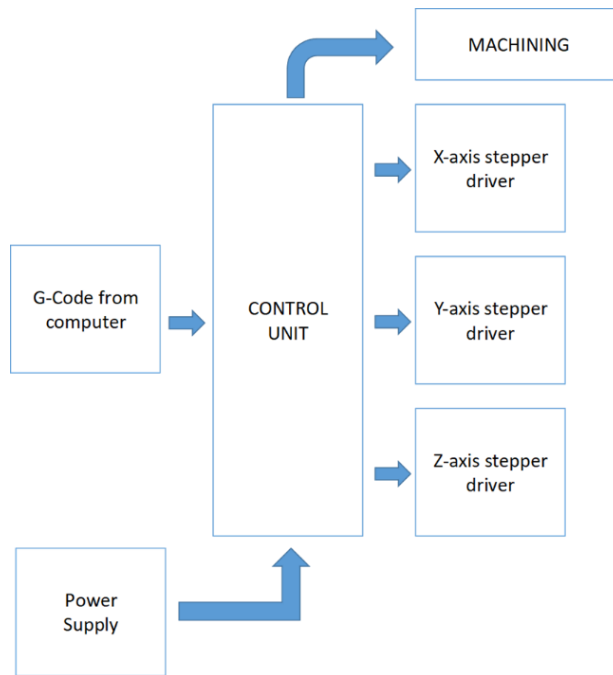**Figure 2.** CNC V3 Shield expansion bard with Arduino UNO

Figure 3. Block Diagram

A modular CNC machine that was developed can sketch out designs with lines and curves. A data wire is used to communicate with the Arduino microcontroller from a computer. Once it has the code, the machine uses stepper motors for its X, Y, and Z axes [13]. The CNC shield will be mounted on top of Arduino where it will deliver current to the common Arduino board and then converts the G code into a digital signal for steppers. The motor moves left and right for X axis, back and front for Y axis, and up and down for Z axis [14]. Figure 4 shows the control system of the modular CNC machines.
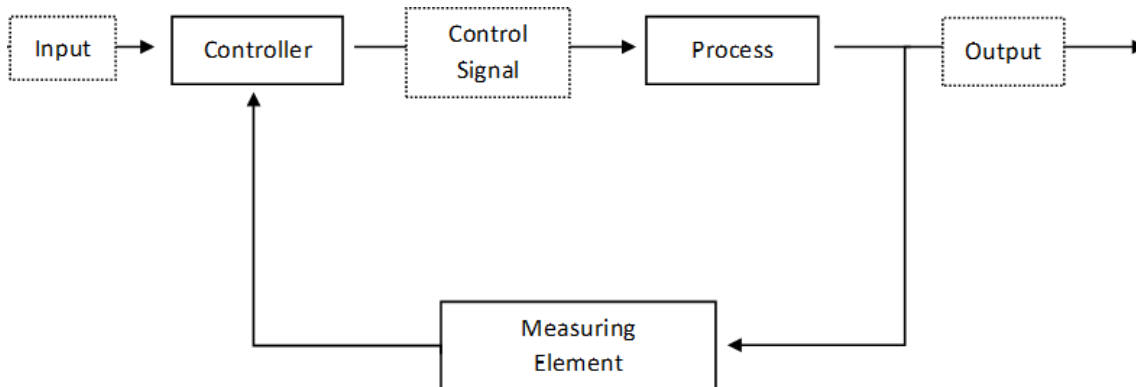

Figure 4. Modular CNC controller control system

Programming Software Development

The primary step in this project is having the Arduino IDE software. This software is a platform to derive commands, a toolbar, a text panel with switches for functions which is common and a sequence of instructions [12]. It interfaces with Arduino and other legitimate hardware to send data and interact with them. Commands written in Arduino IDE are called sketches. The message compartment allows feedback during exporting and saving sketches whereas it can also display if there is an error in the program [5]. Then, the downloaded GRBL library file is included in Arduino IDE and connected to the microcontroller. There is a library designed inside this software especially for 3-axis machines as they support well with G code instruction and are prepared for simple production works. In addition, Universal G-Code Sender (UGS) would be compatible for GRBL as an exchange platform as a G-Code sender [11].

A stepper motor must have a programme sent into it in order to be controlled by an Arduino, but in a CNC machine, all of the motors in the X, Y, and Z axes operate together simultaneously as the step reading changes based on measurements in the G code form [14]. A modular CNC machine is a combination of software, mechanical and electrical parts. A key component of its creation is programming, which begins with the GRBL's interface with Arduino, where it responds to X, Y, and Z axis movement. First of all, the Arduino system has identified the GRBL firmware by adding GRBL into the library of the Arduino microcontroller. Secondly, supporting software like Universal G-code sender is used. The G codes are generated using CAD and CAM software in the numerical code format and is accepted in the software [11].

Universal G-Code Sender is the platform used to send the g-codes. The correct COM port needs to be selected on the software interface and set the baud rate to 115200 as the firmware used is GRBL [14]. Once finished, the machine is operated by uploading the G-codes. A platform that visualizes the tool movement allows us to guarantee that the motors are operating in the direction that has been programmed while the operation is being done. The G-Codes transmitter platform and the machine control terminal are shown in Figures 5 and 6.
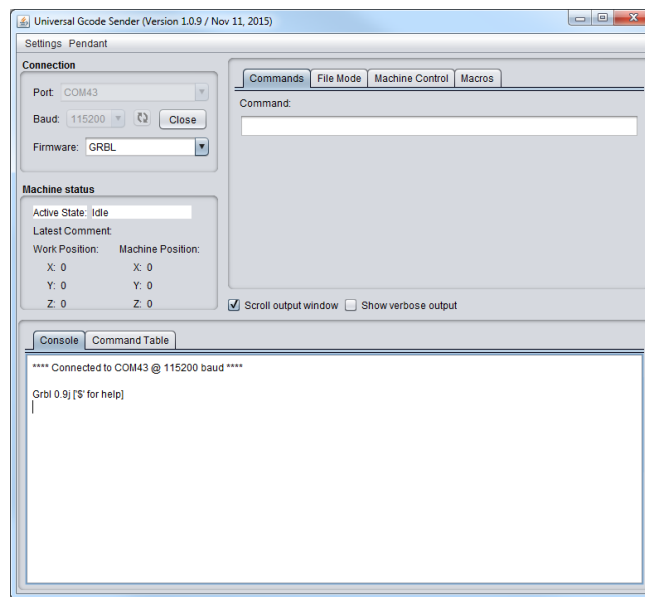
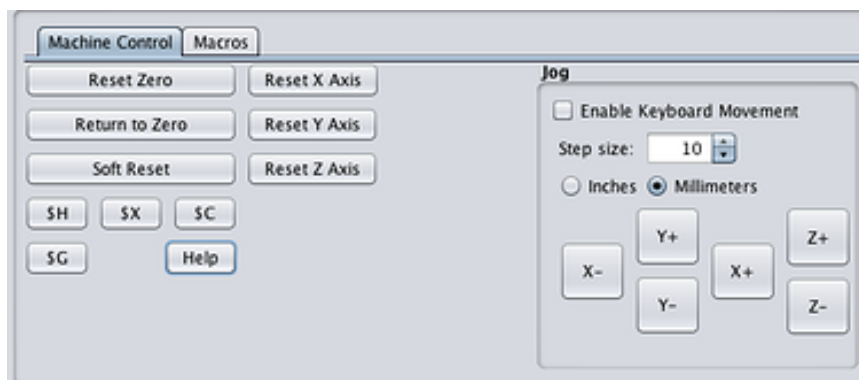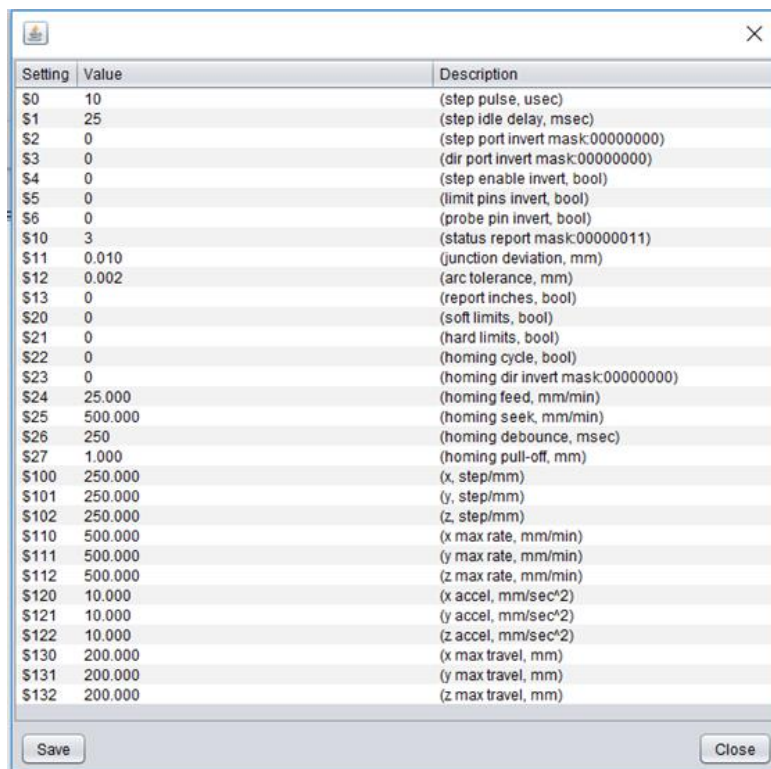

Figure 5. Universal G-code Sender Platform

Figure 6. Manual Machine Control Terminal

RESULT AND DISCUSSION

The brain of this machine is the microcontroller as it is a perfect platform that spans across software and hardware. Besides that, the ideal firmware which runs through this microcontroller as a G-code interpreter is GRBL where it is economical and user-friendly for running a modular CNC machine, especially for milling and 3D machines. GRBL operates primarily through the Arduino serial port interface, which accepts G-Code orders over serial and converts them into motor signals. It also requires a steady stream of G-Code commands to be delivered via computer. In order to achieve exact timing and asynchronous operation, the controller is built in comparatively optimised C employing every intelligent feature of the AVR chips. The general idea of how GRBL works internally in the microcontroller is based on the programs running simultaneously. In order to receive the G-code commands, the main programme reads the serial port. It then transmits the information to a federate planner and sends the entire programme into a ring buffer (FIFO) while other programmes are operating in the background. As it delivers step pulses and bit directions to the stepper, it is also capable of controlling the stepper motors via the CNC shield. Furthermore, it will continuously accept G-code blocks until then there is space in the buffer. Once it is full, the GRBL will send a signal not to send a command until the program ends.

Once the GRBL firmware is flashed into the Arduino Uno microcontroller another important component that makes this modular controller possible is CNC V3 Shield. The shield can be used as a driver expansion board. There're 4 slots in the board for stepper motor drive modules, which can drive 4 stepper motors, and each step stepper motor only needs two I/O port, whereas 6 I/O ports can quite well to manage three stepper motors making it more convenient to use. Moreover, the V3.0 shield is added with support for the GRBL with PWM spindle.

Setting	Value	Description
$0	10	(step pulse, usec)
$1	25	(step idle delay, msec)
$2	0	(step port invert mask:00000000)
$3	0	(dir port invert mask:00000000)
$4	0	(step enable invert, bool)
$5	0	(limit pins invert, bool)
$6	0	(probe pin invert, bool)
$10	3	(status report mask:00000011)
$11	0.010	(junction deviation, mm)
$12	0.002	(arc tolerance, mm)
$13	0	(report inches, bool)
$20	0	(soft limits, bool)
$21	0	(hard limits, bool)
$22	0	(homing cycle, bool)
$23	0	(homing dir invert mask:00000000)
$24	25.000	(homing feed, mm/min)
$25	500.000	(homing seek, mm/min)
$26	250	(homing debounce, msec)
$27	1.000	(homing pull-off, mm)
$100	250.000	(x, step/mm)
$101	250.000	(y, step/mm)
$102	250.000	(z, step/mm)
$110	500.000	(x max rate, mm/min)
$111	500.000	(y max rate, mm/min)
$112	500.000	(z max rate, mm/min)
$120	10.000	(x accel, mm/sec^2)
$121	10.000	(y accel, mm/sec^2)
$122	10.000	(z accel, mm/sec^2)
$130	200.000	(x max travel, mm)
$131	200.000	(y max travel, mm)
$132	200.000	(z max travel, mm)

Figure 7. Firmware settings

Universal G-code Sender is a full-featured G-code platform used for interfacing with advanced CNC controllers like GRBL. The Java application is self-contained and includes all external requirements, so if the Java Runtime Environment is installed, Universal G-code Sender will take care of the rest. However, it won't generate any G-code or convert images into commands, therefore its best use is for sending commands to machines. There is a lot of external software available where it can convert any images into G-code commands thus the firmware which is used here is INKSCAPE. The simulation of the generated G-Code is compared between the CATIA V5 and the Universal G-Code (UGS) Sender software to make sure both do the correct tool path movement. Universal G-code sender software takes this file as input to run the stepper motors incorporated into a mechanical system and draw the desired drawing of the featured block. A similar G-code generation process is also carried out for other features. The tool path generated is visualized in universal G-code sender software as shown in Figures 8 and 9. Meanwhile, the region where the part has been machined is represented in red lining whereas the path where the tool has not passed through is remain white. This enables to identify the machining process as per the designed model.
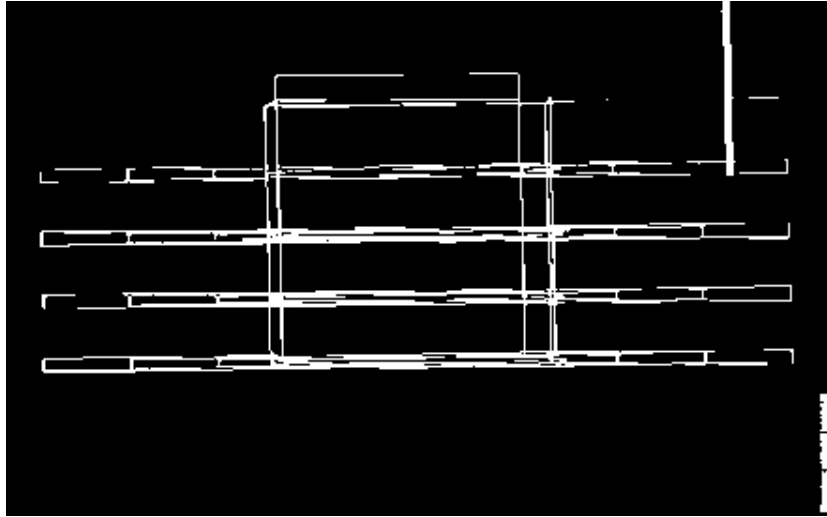
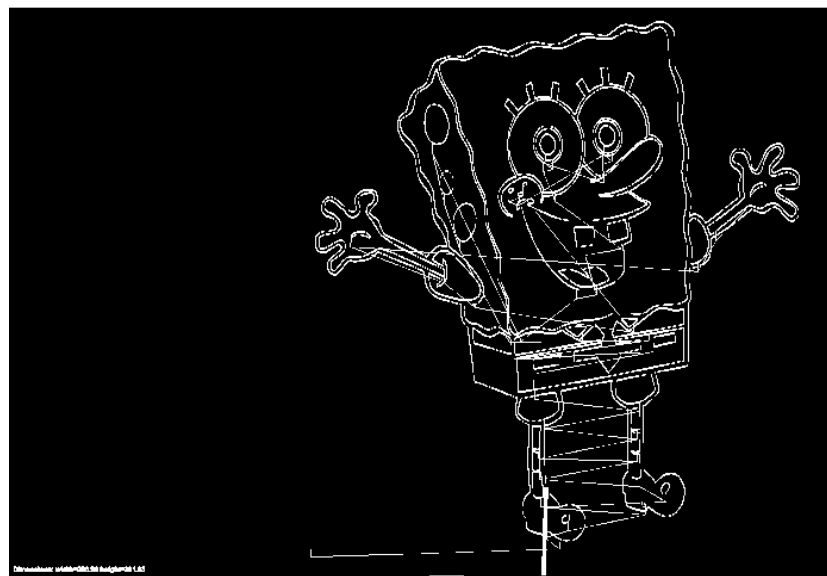**Figure 8.** 3D Simulation in UGS Visualizer

Figure 9. 2D Simulation in UGS Visualizer

Based on the analysis, G-Codes from CATIA V5 and Inkscape is compatible to run with Universal G-Code Sender. This is verified when the movement of the tool path act to be similar in the visualizer of both the softwares. Besides that, the coordinates of X, Y and Z axis are the same even though it is drawn in 2D or 3D models. Both codes can be edited before being transferred to the system if there is any mistake found in the visualizer. Otherwise, the only issue encountered from comparing both the codes is Universal G-Code Sender has limited to interpret as it is using GRBL software to transfer to the G-code command and it only supports the certain modal group in GRBL. Furthermore, GRBL is unable to interpret the G-code line number (N104 G91 G28 Z0) thus codes that are generated from CATIA V5 or other software which have these line numbers have to be removed before being transferred to the system. In addition, the status of codes can easily be reviewed where the codes which do not pass through the system will be stated as error; or unsupported commands, whereas those codes which can be integrated will be stated as OK. Table 1 listed the G codes that compatible with the GRBL software.

Table 1. List of Codes compatible with GRBL

MODAL GROUP	CODES
Motion Mode	**G0**, G1, G2, G3, G38.2, G38.3, G38.4, G38.5, G80
Coordinate System Select	**G54**, G55, G56, G57, G58, G59
Plane Select	**G17**, G18, G19
Distance Mode	**G90**, G91
Arc IJK Distance Mode	**G91.1**
Feed Rate Mode	G93, **G94**
Units Mode	G20, **G21**
Cutter Radius Compensation	**G40**
Tool Length Offset	G43.1, **G49**
Program Mode	**M0**, M1, M2, M30
Spindle State	M3, M4, **M5**
Coolant State	M7, M8, **M9**

Based on the result obtained in Figures 10 and 11, the measurement is the same compared to the one from Inkscape and the drawing created by the machine. There is little imperfection in the drawing because the position of the drawing instrument is not accurate and precise on the drawing surface. Otherwise, the machine able to perform drawing for various shapes and designs without faulty. The safety height is set to be 5mm and the tool diameter used here is 3mm. This can hold a drawing instrument more than 10mm in diameter.

Figure 10. Measurement taken from Inkscape before converting to G-code

Figure 11. Measurement based on the drawing of the machine

CONCLUSION

In this study, the outcomes of the design and development of a programmable controller for a three-axis modular CNC machine are discussed. The controller's primary function is to draw letters and graphics more accurately and with greater mobility. The workload is decreased by this hardware and software configuration. G code makes it simple to locate the locations of every stepper motor that is moving because the status of the moving motor is directly visible on the computer, allowing it to start or stop the machine whenever needed. The developed controller for modular CNC machines has brought flexibility and increased the adaptability for various kind of works.

ACKNOWLEDGEMENT

REFERENCES

[1] Sarkar, B., Khan, A.I., Rana, A.K. and Kundu, S., 2020, March. Quantification of the phase fraction in multiphase steel and 2D design using mini CNC plotter. In IOP Conference Series: Materials Science and Engineering (Vol. 810, No. 1, p. 012066). IOP Publishing.

[2] Alan Overby (2010), CNC Machining Handbook, New York, USA, McGraw-Hill Professional Publishing.

[3] V. V Athani (2008), Stepper motors, New Delhi, New Age International (P) Ltd., Publishers.

[4] Hiroshi KUWATA (1985), The Conversational CNC, Journal of the Society of Mechanical Engineers

[5] James Floyd Kelly Patrick Hood-Daniel (2010), Build Your Own CNC Machine, Apress

[6] Pritesh Runwal, 2017, Design and Manufacturing of Mini CNC Plotter Machine, International Journal for Research in Applied Science and Engineering Technology

[7] Tung, T.T., Quynh, N.X. and Minh, T.V., 2021. Development and Implementation of a Mini CNC Milling Machine. Acta Marisiensis. Seria Technologica, 18(2), pp.24-28.

[8] P Hood-Daniel, JF Kelly, 2009, Build Your Own CNC Machine, Springer.

[9] Aravind Mangudi (2003), Design of a stepper motor driver.

[10] Driving a Stepper | All About Stepper Motors | Adafruit Learning System, Learn.adafruit.com, https://learn.adafruit.com/all-about-stepper-motors/driving-a-stepper

[11] Mohammad Kamruzzaman Khan Prince, 2017, Implementation of a Low-cost CNC Plotter Using Spare Parts, International Journal of Engineering Trends and Technology

[12] Arduino GCode Interpreter – RepRapWiki, Reprap.org, http://reprap.org/wiki/Arduino_GCode_Interpreter

[13] 2000, Mechatronic stepper motor drive, Assembly Automation.

[14] I. Nae and T. Andrei, 2010, "Designing and building a CNC router using stepper motors", Serial Technical