Abstract- The Computer Numerical Control (CNC) development was an important step for the modern industry. These systems can do micrometric positioning of a cutter on space by controlling several mechanical axes. An application example of these systems is on printed circuit boards (PCB) production.

On PCB’s production, the CNC does the positioning of a cutter which removes the copper layer with a drill. However, this cutting method has some disadvantages such as the resolution limitation due to the drill diameter and the high maintenance costs due to drill wear.

The replacement of the conventional cutter with a laser is a solution for these problems. The laser can remove the copper layers without direct contact with the material, eliminating the wear problem. However, the high cost of lasers and their danger level are the main reasons for the low usage of this type of solutions.

This document presents the development of an electronic controller system for a laser CNC. All system development is based on a Colinbus CBR-40 CNC, where only its mechanical structure and motors are effectively used. All electronic control of the motors and laser is developed on this project and efficient discrete controlling algorithms are used with an ARM microcontroller.

I. INTRODUCTION

The modern industry has been involving its production methods. An example of this evolution is the Computer Numeric Control, best known as CNC. A CNC controls the position of a cutter by moving mechanical axes, making possible the execution of high resolution tasks. In general, a CNC works in a tridimensional space and uses an electric drill to cut materials such as metal. One example of this kind of systems are the 3D printers that are very popular nowadays.

A computer sends to the CNC a several instructions that are translated in cutter’s motion to cutting or just engraving. However, the use of a drill has several disadvantages such as the high noise due the drill friction, the necessity of lubes usage to avoid cutting errors and the regular maintenance due the drill wear.

One of possible solutions is the replacement of the electrical driller by a laser cutter. Lasers can cut materials without direct contact, eliminating the high noise levels, the maintenance and the necessity of lubes usage. On the other hand, this solution is more expensive than the electric drill, making this fact one of the main reasons for the low usage of a laser CNC.

The objective of the present project was the development of the hardware and firmware for a laser CNC. The system was based on a Colinbus CNC CBR-40 and only the original mechanical structure and CNC motors were used for the development of this project.

The original version of this Colinbus model uses a parallel port for the communication with the computer (PC), uses stepper motors controllers with only 0.5 step fraction and does not have any kind of controller to the cutting tool. All the system is controlled by a H8/3003 microprocessor from Renesas [1].

In this project, the CNC’s control is made by a STM32F446 microcontroller which communicates with a PC through a Universal Serial Bus (USB) port. Also, the motor controllers were replaced by a modern controller which offers a 1/32 step fraction, allowing a significant resolution increase of the CNC.

The developed system also has a control panel that allows the management of CNC configurations and the execution of some dedicated functions without the need to use a computer.

The communication between the CNC and the PC is made by G-Code commands [2]. These commands are represented in ASCII format and are very common in this kind of systems.

The developed system uses a 2500 mW semiconductor laser, emitting light with a wavelength centered at 440 nm. As there is not much information on the characteristics of this laser, several tests were carried out to obtain the fundamental characteristics so that the integration of this system in the CNC was possible. The carried tests have led to the electrical and optical characteristics of the laser, such as its light spectral width, optical power, cutting diameter and current vs voltage curves.

A driver to the laser to control the light intensity by limiting the current flow was also developed. This circuit also controls the laser’s operation mode, continuous or pulsed. All these functions are controlled by the STM microcontroller.

Briefly, the hardware developed in this project has four boards: main board, microprocessor support board, control panel and laser control board. The main board, laser control board and control panel were developed on this project. Only the microcontroller support board was acquired.

The firmware was developed in C and C++ and the FreeRTOS operating system was used. This operating system provides management of the processor occupancy, allowing the execution of several tasks in real time. Thus, this CNC is composed by four different tasks: the “cncSystemControlTask” which makes the management of CNC’s system, the “parserPlannerTask” that’s responsible for the G-Code commands interpretation and processing, the “runnerTask” that controls the motors and laser to execute the received G-Code commands. The last task is the “menuTask” which, as its name
suggests, is the responsible for the management of the CNC’s control panel. Which task has its execution priority and thus, its execution order, is controlled by semaphores mechanisms. The use of FreeRTOS allowed the execution of these tasks as an assembly line, minimizing the waiting time between tasks and making the CNC more efficient.

The CNC can perform rectilinear and curvilinear movements, for which the Bresenham algorithms are used. Because of the inertia of the movements, were also developed algorithms to control the speed of movements in order to preserve the quality of the print.

In the scope of this project, the interpretation of the G-Code commands results in execution blocks. Thus, a block represents a certain action that contains a series of characteristics to be executed by the CNC. A block can represent a motion, a request to turn the laser on, or a modification of some CNC definition.

In a print job, it is frequent to receive several blocks for CNC movement. In order to optimize the execution of these type of blocks, the "multi blocks look ahead” philosophy is used. This technique consists in the interconnection of several blocks forming a super block. In this way, the speed profile planning is done considering the whole super block and not block by block. This method minimizes the number of stops periods during printing, increasing the efficiency of the CNC.

The laser’s power control is done using the speed planning control. Since one of the characteristics that defines the cut profile made by the laser is the exposure time, the speed of the movements during the cut has a direct influence on the result.

This paper presents all the procedures, tests and algorithms developed during the realization of this project. It also presents the state of the art of this system which was the basis for this project.

II. HARDWARE

As referenced before, this project was based on a Colinbus CBR-40 CNC [3] (Figure 1) and only the mechanical structure and motors of this CNC were used on the development of the project.

![Figure 1 - Colinbus CBR-40.](image)

The replacement of the electric cutter by a laser was one of the requirements of this project. The use of the laser, among other advantages, improved the maximum resolution of the system, reducing noise levels. Support for calibration of the CNC’s axes and a USB communication interface to replace the parallel port of the original model were also developed.

A. MECHANICAL CHARACTERISTICS

The structure of the CNC Colinbus CBR-40 (Figure 1) is made of solid aluminum, which makes this system more immune to oscillations, but has the disadvantage of having a lower speed. This CNC has three axes (X,Y and Z) where which axes is moved by a screw worm directly connected by a stepper motor Nanotec ST4118M1206 [4] (Figure 2).

Measuring the distance between the successive blades of the screw, the CNC’s resolution was obtained as in equation (1) where the R is the resolution in millimeters, the d is the distance between the blades (also in millimeters) and the N the number of steps for each motor’s rotation.

\[
R = \frac{d}{N} [\text{mm}] 
\]

![Figure 2 - Mechanical structure of Y’s axes.](image)

In this project, the number of steps per rotation is 6400 and the blade’s distance is 3mm which gives a resolution of 0.5\(\mu\)m. Since the original resolution was 7.5 \(\mu\)m [3], an improvement of 15 times was achieved in this project.

The origin of the number of steps presented above comes from the replacement of the original motor’s driver by a new driver. The original drive is the STMicroelectronics L6208PD [5] that can work with a maximum step fraction of half-step. This driver was then replaced by the DRV8825 from Texas Instruments which offers a maximum step-fraction of 1/32. This driver was acquired in a support board from the Pololu Corporation [6]. Thus, once the motor presents 200 steps per rotation in full step mode, this drive can perform 32 times more steps, that is 6400 steps. In Table 1 a general comparison of this CNC before and after the hardware modifications is presented.

<table>
<thead>
<tr>
<th>CNC</th>
<th>Colinbus</th>
<th>Present project</th>
</tr>
</thead>
<tbody>
<tr>
<td>Work frame (X,Y,Z) [mm]</td>
<td>300,400,100</td>
<td>300,400,100</td>
</tr>
<tr>
<td>Mechanical resolution [(\mu)m]</td>
<td>7.5</td>
<td>0.46875</td>
</tr>
<tr>
<td>Max.speed [mm/min]</td>
<td>3600</td>
<td>1546.0875</td>
</tr>
<tr>
<td>Step fraction</td>
<td>0.5</td>
<td>0.03125</td>
</tr>
<tr>
<td>Software</td>
<td>Colin Drive, CAM package</td>
<td>CNC Pathfinder</td>
</tr>
</tbody>
</table>

Table 1 - Characteristics of the CNC before and after modifications.
B. SYSTEM CONTROL ARCHITECTURE

In the development of the control system, a STMicroelectronics STM32F446 microprocessor (Figure 3) was used. This microcontroller is based on the ARM Cortex-M4 processor which features a 32-bit RISC-like architecture and operates at a maximum clock of 180 MHz. The core in this device incorporates a floating-point processing unit and a vectorized interrupt system. In addition, this microcontroller has 512 kB of flash memory, 128 kB of SRAM memory, 3 ADCs and 2 12-bit DACs, 14 timers, 4 I2C interfaces, SPI and UART, among others.

Of the available timers, 3 were used for motor speed control, pulsed laser mode control and watchdog timer. An I2C communication interface was used to communicate with the control panel display and the external EEPROM memory to save CNC’s settings. Two UART interfaces were used for communication with the computer (UART-USB converter) and for programming/system debug mode. For the laser electrical characterizer, two analogue/digital (ADC) converters were used to read the current and voltage and a digital/anologue converter (DAC) for adjusting the laser power. The remaining devices (Figure 4) were connected through a General Purpose Input Output (GPIO) port.

C. CNC’S BOARDS

The developed CNC’s main board supports the Nucleo’s board and the three motor drivers (X, Y and Z). In addition, an ATM24 AT24C02 flash memory [7] and an FTDI FT232R USB / UART model bidirectional converter [8] were placed on this board (Figure 5).

The control panel consists of a 4x16 Liquid Cristal Display (LCD) [9], a rotary encoder for menu navigation and a buzzer for sound indication (Figure 5). The LCD uses a controller that receives 8-bit or 4-bit parallel commands to perform writing or other dedicated functions on the LCD [9]. In order to reduce the number of signals used by the LCD, an I2C converter for 8-bit words, model PCF8574 [15] was used.

The CNC’s interface also has an emergency stop button (Figure 5) to cut off the power of the motors and the laser (12V) and activate a high priority interruption routine that requires the CNC’s control system to stop printing and block all the functionalities.

D. LASER

A laser (Light Amplification by Stimulated Emission of Radiation) is a device that has the ability to produce and emit electromagnetic radiation. Compared to other devices, the radiation emitted by the laser has special characteristics such as the emission of coherent monochromatic radiation.

Due to their characteristics, lasers are widely used in industrial areas to perform high precision tasks. Measurement of physical quantities, cutting, drilling and recording materials are some of the tasks performed by lasers in the several areas of modern industry [10].

The focus of this project is on the use of lasers to cut and engrave various materials such as wood, plastic and cardboard.

In the development of this project, a 2.5 W laser with a wavelength of 445 nm was used (Figure 6). In addition to this information, no other characteristics of this laser were known.

In order to be able to integrate this laser into the CNC, it was necessary to perform several tests to obtain some essential characteristics for the production of the final prototype.

The entire electronic laser control system was also developed in this project.
E. **LASER DRIVER**

Similar to an LED, the light intensity of a semiconductor laser is best controlled by the current flow. In the course of this project the circuit shown in Figure 7 was developed.

![Figure 7 - Laser's driver developed in this project.](image)

By controlling the voltage applied to the transistor gate, it is possible to control the amount of current flowing between the collector and the emitter of the transistor. The transistor used in this project (Q1 - Figure 7) was an N channel mosfet model IRF540 [11]. This transistor was chosen because it supports a maximum pulsed mode current of 110 A and a VGS (th) between 2.2 and 4 V.

A DAC, integrated on microprocessor chip, was used to perform the control of this voltage. An operational amplifier model LM358 [12] and a resistance of 1 Ω were also added to achieve the balance between the DAC’s voltage and the current circulating in the laser and, thus, to linearize the control of that current.

As pulsed mode was one of the modes of operation of the laser, another N-MOS (Q2) transistor, model AO3400 [13], was added to operate as a switch commanded by a square wave signal generated by a PWM which is also available in the microcontroller used.

F. **CHARACTERISTICS OF THE LASER**

Analyzing the results represented in Figure 8, one of the most evident characteristics is the increase of the current with the voltage. It was verified that the current of this laser is limited when the voltage increase. This is a characteristic of this type of semiconductor laser [14].

![Figure 8 - Electric characteristics of the laser.](image)

An optical sensor model 818-SL [15] and an optical power measure system, model 835 [16], from Newport, were used to perform the optical characterization of the laser. In order to control the luminous intensity of the laser, the laser controller developed in this project was used. To prevent damage of the optical sensor, it was only possible to measure up to 90% of the maximum laser power. Figure 9 shows the optical power of the laser as a function of the wavelength.

![Figure 9 - Optical power characteristic of the laser.](image)

It is important to note that the optical sensor is not capable of performing readings below 400 nm [15], so the graph depicted in Figure 9 may not coincide with reality for λ < 400 nm. From the information gathered it was possible to conclude that, considering a band width of 20 nm, the average power is approximately 963.65 mW. This power is far from the expected 2500 mW, however, because there is no information about the measurement methods used by the manufacturer of this laser, it was not possible to have a reference of the analysis performed.

The laser used in this design (Figure 6) can be focused by rotating the lens. Even with a constant diameter, the characteristics of the cut made by the laser depend on several factors such as the reflection of the material, the time of exposure and the energy of the radiation. To characterize these factors, several tests were performed on the laser cutting profile in wood and paperboard as presented in Figure 10.

![Figure 10 - Speed and power relation for a cutting of 0.2mm in paperboard.](image)

As expected, the cutting diameter increases with power and decreases when speeds increase. The graph also shows that the variations of diameter are not linear, mainly for powers above 2000 mW where the diameter tends to increase more. It is also possible to identify an area, of purple color, in which there is no cut.
III. Firmware

The Numerical Control Kernel (NCK) is the system responsible for the operations of the CNC, such as the interpretation of commands sent by the computer, interpolation and execution of movements. This system can be implemented using a real-time operating system (RTOS) or a FSM-based (Finite State Machine) architecture.

The project presented in this document is based on the FreeRTOS [17] operating system which manages 4 different tasks in this system: "cncSystemControlTask", "parser/planner task", "runner task" and "menu task". The FreeRTOS has an implicit task, called “idle task”, which its purpose is the management of system’s memory and other tasks. The implemented NCK’s architecture is presented in Figure 11.

![NCK system architecture.](Image)

The system has four different semaphore systems that allow the control of the execution order of the four different tasks without polling systems. A watchdog timer is also used to avoid some accident with the laser if some system anomaly locks the CNC’s system. The use of independent tasks managed by the scheduler allow the execution of different functions in real time.

Table 2 presents an example of the system’s execution timeline.

<table>
<thead>
<tr>
<th>Task</th>
<th>SystemControl</th>
<th>Runner</th>
<th>Parser/Planner</th>
<th>Menu</th>
<th>Idle task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 11 - NCK system architecture.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A computer sends G-Code [2] commands to the CNC’s NCK system. These commands are received by UART module that unlock the parser/planner semaphore when the entire ASCII command is received. The command is then interpreted by the parser/planner task and transformed into an internal data structure called “block”. These blocks are placed in a FIFO (First In First Out) queue that also unlocks the runner task when a new block is available. Finally, the blocks are executed by the runner task which is responsible for the control of the motors and laser. The system task is responsible for managing high priority operations such as the emergency stop and the menu task is responsible for managing the graphical interface of this CNC.

G. The “cncSystemControlTask”

The purpose of this task is to control the states of the remaining tasks that compose this system. The programming basis for this task consists of a state machine through the use of an infinite loop and a switch-case structure that is activated by the release of the control semaphore from this task.

When started, the cncSystemControlTask starts by initializing some CNC features such as the LCD and reading the CNC settings stored in the external EEPROM memory. After the features are initialized, this task is locked. This task is only unlocked when other tasks or interrupt routines request it. An example of this is the beginning of a print: When runnerTask requests to activate the CNC print mode, it suspends the menu task, turns off control panel button interrupts, emergency stop and activates the watchdog.

This task is also called when a print job is finished, when the emergency stop or the pause/resume option is activated during a print or a critical fault, such as motors overcurrent fault, occurs.

H. Parser/Planner Task

The parser/planner task is responsible for receiving the G-Code commands that are then interpreted and converted to be executed by the runner task. This task is blocked by a control semaphore that can be unlocked by the UART module when a new command is received. The received commands are of type G-Code and, as such, this system follows the protocol referring to G-Code [18]. G-Code has two letters to define its operations: G and M. The G commands are related directly with CNC’s operation actions such as motion and some print settings. M codes are reserved for CNC’s system settings like adjustments in motors behavior. The motions commands supported by this system are G01 for linear motions and G02 and G03 for clockwise or counter clockwise arc motion.

As mentioned before, this task receives commands from the UART module and parses them. The interpretation system parses the commands by reviewing syntax errors in the command, if the command is supported by the system, and whether the command has all the necessary information. If a wrong command is received, this task has the responsibility of informing the computer by sending the command “er”. Otherwise, if the command is accepted an “ok” command is sent by the CNC.

After the parser process, the command is converted into an internal data structure called “block”. The block has the necessary data for proper execution by runner task. In the case of motion commands, the interpolation functions are responsible for filling the block with the motion data.

I. Linear Interpolation

To create any kind of movement, the CNC must have the ability to execute at least vertical, horizontal, diagonal and arcs. Because it is a digital system, CNCs recognize their work area as a plane or space, discrete and non-continuous, making it impossible to represent perfectly linear lines. The CNCs use interpolation algorithms that perform the conversion of perfect lines into discrete lines. In the case of this project, the Bresenham algorithms [19] were used.

5
Considering a two-dimensional line with origin \((x_1; y_1)\) and destination \((x_2; y_2)\), the Bresenham method starts by analyzing the line by its variation in the x, y axes:

\[
\begin{align*}
\Delta x &= x_2 - x_1 \\
\Delta y &= y_2 - y_1
\end{align*}
\] (2)

The axis that presents the greatest variation is called the driver axis and the other is called the driven axis. Thus, if \(|\Delta x| > |\Delta y|\), the x-axis is represented as the driver and the y-axis as the driven, otherwise the order is reversed. In this way, the driving axis is always moved in all iterations of the algorithm, whereas the driven axis is only moved through the following rule:

\[ e_y = e_{y-1} + \frac{\Delta y}{\Delta x} \] (3)

From the previous expression, the y-axis is only moved if \(e_y > 0\), where \(1 - \frac{\Delta y}{\Delta x}\) is its initial value. The positive or negative movement of the axes depends on the value of \(\Delta y\) and \(\Delta x\). If \(\Delta y > 0\), the y-axis is moved in the positive direction, otherwise the movement is in the negative direction. The same occurs on the x-axis in case this is the driver axis.

In order to optimize the performance of this algorithm, the division operations were canceled by multiplying the expression of the error by the conductor variation \((\Delta x)\) [20]. The pseudo code of this algorithm can be represented as follows:

\begin{verbatim}
Get \((x_1; y_1)\) e \((x_2; y_2)\)
\Delta x = x_2 - x_1
\Delta y = y_2 - y_1
j = y_1
\delta = \Delta y / \Delta x
For \(i = x_1\) to \(x_2 - 1\)
    Move \((i; j)\)
    If \(e \geq 0\)
        \(j++ = 1\)
        \(e = e + \delta\)
    End if
    \(e += \Delta y\)
End for
\end{verbatim}

\section{Circular Interpolation}

In the same way as the linear method, it is necessary to identify the driver that represents the axis with the greatest variation. It is considered a circular area divided into eight equal parts (octants):

\begin{figure}[h]
\centering
\includegraphics[width=0.4\textwidth]{octants.png}
\caption{Circumference octants.}
\end{figure}

Considering the 1st octant represented in the figure above, it is possible to verify that the y-axis varies more than the x-axis, making y the driver and x the driven. The same happens for the 4th, 5th and 8th octant. In the case of the 2nd, 3rd, 6th and 7th octant, the driver is the x-axis. As previously mentioned, the driver is always moved at each iteration of the algorithm, knowing if the driver is moved or not.

By keeping the value of the radius, the x-axis is only moved if:

\[ |e(x - 1, y + 1)| < |e(x, y + 1)| \] (4)

If the error in \(x\) is greater than the error in \(x - 1\), it means that the x-axis is moved, otherwise it remains in the same location. The same calculations are made in case \(x\) is the driver but taking into account the error in \(y\) [21]. This cycle repeats until the arc reaches the desired location. The following pseudocode represents this algorithm for the case of y-axis is the driver:

\begin{verbatim}
Get \((x_1; y_1)\) \((x_2; y_2)\) e r
\(x = x_1\)
\(y = y_1\)
While \(y \neq y_2\) and \(x \neq x_2\) do
    Move\((x; y)\)
    If\(|x| > |y|\)
        If\(|e(x - 1, y + 1)| < |e(x, y + 1)|\)
            \(x -- = 1\)
        End if
        \(y++ = 1\)
    End if
End if
End While
\end{verbatim}

\section{Speed Profile of a Block}

The velocity profile developed in this project is based on the first order polynomial profile [45]. This profile is defined by an acceleration and deceleration slope, the step number at which the acceleration and deceleration period ends and the maximum speed (Figure 13).

\begin{figure}[h]
\centering
\includegraphics[width=0.4\textwidth]{speed_profile.png}
\caption{Speed profile of a block}
\end{figure}

The acceleration and deceleration periods, represented in Hz / Stp, have a constant slope that can be modified in the CNC settings menu. Once the slope is set, it is not always possible to reach the maximum speed. If the number of steps required to perform the acceleration and deceleration periods exceeds the total number of steps of the movement, the CNC will never reach the maximum speed. Thus, the policy of lowering the
maximum speed was defined in order to ensure that the CNC accelerates and decelerates safely.

As shown in Figure 13, if the point of intersection of the acceleration and deceleration lines is located below the maximum speed of the block, the maximum speed becomes equal to the value of the velocity resulting from the intersection. However, this process is valid in case the initial and final velocities of the block are equal. It may be the case that these speeds differ in such a way that the intersection point is located on the sides of the original profile of the block and outside it. In these cases, the speed profile of the block will only present an acceleration or deceleration period.

Note that the profiles presented in this document represent the velocity of the movements and, therefore, the graphs do not present any information about the direction of the movements. The direction of the movements is thus controlled by the interpolation algorithms.

**L. MULTI-BLOCKS LOOK AHEAD ALGORITHM**

Instead of processing and executing block by block, the multi-block look ahead method (MBLA) identifies the blocks that can be aggregated to be processed and executed together, increasing the efficiency of the CNC. The aggregation of a set of blocks is called a super block (Figure 15). If a new block is unable to be aggregated into the current super block, the current super block is sent to runner task and a new super block begins as illustrated on the red example in Figure 12.

Directly related to the block’s speed profiler, in this project three modes of operation of the MBLA method were developed: fast, normal and high quality.

Based on existing methods [22], what differentiates these modes is how the MBLA analyzes the blocks. In the case of the high-quality method, only blocks whose direction of movement is equal are aggregated. The speed of each block is also taken into account and can be modified to ensure the print quality (Figure 16).

On the other hand, the fast mode does not take into account the direction of the movements nor the speed of each block. This method receives a set of blocks and assigns the maximum speed contained in that set of blocks to all blocks. Thus, in a print job, there is only one period of acceleration and deceleration, reducing the printing time but increasing the probability of printing errors.

The normal mode works like the high-quality mode, with the difference that it tries to respect the maximum velocity of each block and does not take into account the direction of the movements.

The graph of Figure 63 illustrates the print times for the MBLA modes for the three planning stages of the super blocks profile. These results were obtained from a simulation of some blocks with similar characteristics of a print job using Zig-Zag pattern.

As previously mentioned, the high-quality mode has a longer execution time followed by the normal mode as shown in examples a) and b) in Figure 18.
At first glance, there are no obvious differences between the high-quality mode and the normal mode. However, in a more careful analysis, it is possible to verify that the cuts made by the laser have less failures in some zones.

In spite of these results, it was possible to verify that depending on the characteristics of the figure to be printed and the programmed speed and power, the fast mode can have a print quality equivalent to normal and even high-quality mode as shown in example a) in Figure 19.

The b) and c) results shows that the balance between power and print speed has an important role in print quality. As demonstrated in example b), excessive power may lead to decreased print quality. On the other hand, example c) illustrates the various faults that can occur in the motors in case of a too high speed.

IV. RUNNER TASK

The runner task is responsible for the execution of the blocks and thus, controls the motors and the laser. This task has the second highest priority and is suspended until no blocks or super blocks are placed in the queue.

The motor controllers receive a square signal from the microcontroller in order to make the motors move. In this way, the speed of the motors depends directly on the frequency of this signal. To implement this function a timer already integrated in the microcontroller, which triggers an interrupt routine was used. When the interrupt is triggered, a GPIO port is active for a period of 5 μs. To control the frequency of this signal two control registers of this timer, prescaler and period are used. The period is given by the following equation [23]:

\[
\text{Period} = \frac{\text{Timer}_{\text{clk}}}{(\text{motorFreq}) \times (\text{Prescaler} + 1)} - 1 \text{ [Hz]}
\]

where \(\text{motorFreq}\) is the frequency of the signal sent to the motor controllers and \(\text{Timer}_{\text{clk}}\) is the control frequency of the timers which in this case is 48 MHz. Thus, by setting a constant value for the prescaler register it is possible to obtain a function that directly controls the speed of the motors.

After calculating the period value, the timer is then triggered, and a semaphore system is activated by suspending the operation of the runner task. This solution allows the other available tasks to run and therefore, in the case of this project, suspending the runner task allows the parser/planner task to continue to receive and process commands. Thus, the runner task does not have to suspend their functions to wait for parser/planner to prepare new blocks.

The laser control system developed in this project has the objective of controlling the intensity of the light emitted by it in order to modulate the profile of the cut. In most cases, the CNCs uses an electric mill for printing. Since different materials have different characteristics such as density and thickness, it is necessary to adapt the speed of the electric cutter to the characteristics of the material used.

As previously mentioned, in the Laser driver, an electronic circuit was developed that allows the digital control of the luminous intensity of the laser. In this way, it is possible to control the power emitted by the laser through the voltage variation of one of the DACs that the microcontroller has (Figure 20).

Since the system developed in this project has an acceleration and deceleration controller, it is important to maintain the uniformity of the laser cut at these times so that the ratio of the laser power and the speed of the movements is maintained. In the case of electric drills, one of the G-Code parameters is the number of revolutions per minute of the electric cutter [18]. In this project this value refers to the power that the laser has to emit. Thus, the parser/planner task is responsible for acquiring this value and calculating the power/speed ratio for each block. For example, if a command indicates a motion to be executed with a maximum speed of 1000 Hz and a power of 1000 mW, the power/speed ratio is 1. In this way, modulation of laser power is done by multiplying the value of speed by this value.

As the velocity modulation system is already implemented, the laser power profile will present similar characteristics to the velocity profile of the blocks (Figure 21). However, the laser power is limited to 2500 mW so that the ratio of power/speed can fail if the speed is higher than that used in the power factor calculation.
V. Menu Task

The menu task is responsible for the management of the CNC’s native interface. The interface developed (Figure 22) consists of a 4-line and 16-column LCD with, a buzzer and a rotary encoder, which also includes an “OK” button for menu navigation.

Regarding the LCD, its control is made through an I2C serial communication [24]. The pulses coming from the encoder are received via GPIO which activates an interrupt routine. This routine is used to identify the direction of rotation of the encoder and to indicate to the system that a new input has occurred at the CNC interface.

When a new input is received, the menu’s control semaphore is unlocked and thus the menu task is able to execute the proper function for the received input.

Since each menu option has its functions and the signals from the encoder have different effects on them, each menu option is represented by an object. Thus, the menu task is composed by several objects stored in a table where each one knows which object to call depending on the encoder inputs (Figure 23).

As shown in Figure 23, each black dot represents a menu option which, as mentioned earlier, represents an object. For instance, in the case of the “Laser” option this object knows that when the "OK" button is pressed it must call the "On / Off" object. When the "On / Off" object is executed, it sends to the LCD controller a character table containing its own graphical interface. Therefore, each object has stored in its structure the graphic interface for its menu. This graphical interface is sent to the LCD controller that updates it whenever a new signal is received from the encoder.

VI. Conclusions

The objective of this project was the development of a CNC that uses a laser as a cutting tool. For this, an existing mechanical structure was used, composed by the step motors and all the mechanisms necessary for the positioning of the cutting tool. A laser already adapted for this purpose was also used.

The development of this project consisted on the development of the CNC firmware and the necessary hardware such as the control panel of the CNC and the main board that supported the microcontroller and the controllers of the motors. The power control circuit of the laser has also designed on this project. This circuit also allows the analysis of the electrical characteristics by controlling the current of the laser and reading its voltage and current.

Since all the optical and electrical characteristics of the laser were not known, it was necessary to perform several tests to classify the laser. The analysis of the electrical characteristics was performed with the help of the circuit developed in this project for this purpose. Optical interaction tests were performed between the laser and different materials: paper, paperboard and wood presented satisfactory results. Also, the relation between the power emitted by the laser and the speed of the movements was found through the performed tests on the materials. In this way it is concluded that this relation can present very different values depending on the type of material, color and thickness.

A CNC is a complex system used for high precision tasks. These tasks are usually sent by a computer to the CNC control kernel (NCK). In general, these systems can use a microcontroller to control their entire structure. In the particular case of this project, the control system of the CNC is made by a microcontroller STM32F446, being a low-cost solution and easy to program. In this way, this system has the responsibility to carry out the several tasks that will be necessary for the accomplishment of tasks by the CNC.

The programming of the developed NCK system was based on the FreeRTOS operating system. This system allows the creation of several independent tasks. The main tasks developed in the project were the parser/planner task and the runner task. These tasks are responsible for receiving commands from the computer that are later interpreted and executed. There is also the general system control task (cncSystemControlTask) and the task that controls the native interface of the CNC (Menu Task).

The parser/planner task is responsible for interpreting the G-CODE commands that are later sent to the runner task to be executed. In the context of this project, the commands are interpreted and converted into a data structure called “block”. This block can then be connected to other blocks creating a super block. This interconnection mechanism allows a set of blocks to be executed by the runner task while new blocks are generated by the parser/planner task without printing being suspended. This mechanism is called by “multi-blocks look ahead”.

The use of stepper motors requires speed control as they have some limitations on startup and top speed, which can cause problems during operation. In this project this control is realized through a mathematical model based on first order
polynomial functions. The main advantages of this model are its simplicity and low computational weight.

With the objective of exploring the potentialities of the mechanisms developed in this project, three printing modes were developed: fast, normal and high-quality mode. These modes differ in the analysis of how the blocks are linked by the multi-blocks look ahead and how the speed profile of these is created. In this way, the CNC offers the possibility for the user to choose between the quality of the print and the time required for its realization.

In a two-dimensional domain, the CNC only needs to perform two types of motion to execute a printing: linear and curvilinear motions. Since the CNCs work in the digital domain, the quality of the interpolation algorithms of these movements is the characteristic that has the greatest influence on the impression.

One of the algorithms most used by CNCs is the Bresenham algorithm. These algorithms perform the discretization of the movements in a simple way and with good levels of quality, having the main advantage of a low processing weight.

In this project a graphic interface for interaction with the CNC was also developed. This interface is composed by an LCD, a rotary encoder, a buzzer and an emergency stop button. The menu of this system has a number of features such as CNC programming and resolution and calibration, modification of CNC characteristics such as its laser features such as power programming and pulse sending for testing purposes.

For future projects related to this system, speed control improvements are recommended. Speed control methods and the method of creating the profile of the blocks can be improved to increase the quality of the prints.

The calibration system can also be improved. Detection of the origin of the axes via pushbuttons can bring some errors. Replacing this system with a distance measurement system such as ultrasound can help with more accurate calibration.

The usage of another type of laser can also improve this CNC. A laser with other characteristics can increase the range of possible materials to work by this CNC.

Finally, the interface of the CNC can be improved by replacing the LCD screen for a color screen with a higher resolution. The use of touch screen technology can also improve the interaction with the CNC interface. Connecting the CNC in a local area network, via wireless or ethernet cable, can also be an advantageous improvement for this system.

VII. REFERENCES