

Universidad Tecnológica Nacional

Proyecto Final

Cortadora de cartón automatizada

Autor:

- *Ariel David Costanzo*

Director:

- *Ing. Mag. Sergio Burgos*

*Proyecto final presentado para cumplimentar los requisitos académicos
para acceder al título de Ingeniero Electrónico*

en la

Facultad Regional Paraná

Abril 2024

Declaración de autoría:

Yo declaro que el Proyecto Final “Cortadora de cartón automatizada” y el trabajo realizado son propios.

Declaro:

- Este trabajo fue realizado en su totalidad, o principalmente, para acceder al título de grado de Ingeniero electrónico, en la Universidad Tecnológica Nacional, Regional Paraná.
- Se establece claramente que el desarrollo realizado y el informe que lo acompaña no han sido previamente utilizados para acceder a otro título de grado o pre-grado.
- Siempre que se ha utilizado trabajo de otros autores, el mismo ha sido correctamente citado. El resto del trabajo es de autoría propia.
- Se ha indicado y agradecido correctamente a todos aquellos que han colaborado con el presente trabajo.
- Cuando el trabajo forma parte de un trabajo de mayores dimensiones donde han participado otras personas, se ha indicado claramente el alcance del trabajo realizado.

Firmas:

-
-
-

Fecha:

Agradecimientos:

Agradezco enormemente el apoyo incondicional de mi familia especialmente mi papá y mi mamá que me sostuvieron económicamente y me motivaron a no bajar nunca los brazos en todo este tiempo.

A mi compañero que me acompañó durante gran parte de la carrera y estuvo presente en cada momento en que necesité un oído o una palabra para seguir en marcha.

Al Ing. Sergio Burgos, cuya ayuda fue clave para la conclusión de este proyecto y un gran docente a lo largo de la carrera, quien siempre estuvo a disposición cuando le solicité ayuda.

A la Facultad Regional Paraná, todo su equipo docente que me instruyó en estos años, y a la Universidad Tecnológica Nacional, gratuita y pública.

Finalmente agradecer a todos aquellos amigos y amigas, quienes han estado en todo o parte de este proceso y me han alentado y sostenido para culminar mi formación.

Ariel David Costanzo

Universidad Tecnológica Nacional

Abstract

Facultad Regional Paraná

Ingeniero en Electrónica

Cortadora de cartón automatizada

Ariel David Costanzo

Abstract:

Design and construction of a low-cost, compact automated cardboard cutting machine for the manufacturing of highly customized cardboard and paper products.

Built using additive manufacturing and Open-Source CAD design tools such as Inkscape and FreeCAD, developed with microcontrollers featuring built-in wireless connectivity from the ESP-32 series and the FluidNC firmware specifically configured for this application.

Resulting in a machine capable of precisely cutting multiple materials taking as input two-dimensional designs created by software.

Keywords:

Cutting machine, 3D printing, Open Source, Fluid NC, ESP32.

Resumen:

Diseño y construcción de una máquina de corte de cartón automatizada de bajo costo y porte para la manufactura de productos de cartón y papel altamente personalizados.

Construida mediante la fabricación aditiva, utilizando herramientas de diseño CAD Open-Source como Inkscape y FreeCAD, desarrollada con microcontroladores con conectividad inalámbrica incorporada de la serie ESP32 y el firmware FluidNC configurado específicamente para dicha aplicación.

Obteniendo una máquina capaz de realizar el corte preciso de múltiples materiales a partir de diseños bidimensionales creados por software.

Palabras Clave:

Máquina de corte, Impresión 3D, Software libre, Fluid NC, ESP32.

Dedicado a:

Sonia y Rubén,
con quienes soñamos juntos
este momento.

Índice:

Capítulo 1: Introducción.....	1
1.1 Idea que dio origen al proyecto.....	1
1.2 Descripción del proyecto.....	1
1.3 Objetivos.....	1
1.4 Análisis de mercado.....	1
Capítulo 2: Desarrollo.....	3
2.1 Diagrama en bloques del proyecto.....	3
2.2 Desarrollo del sistema mecánico.....	4
2.2.1 Diseño y fabricación de la herramienta de corte.....	6
2.2.2 Diseño y fabricación de la estructura de la máquina.....	7
2.3 Desarrollo del sistema electrónico.....	9
2.3.1 Desarrollo del prototipo del sistema electrónico.....	9
2.3.2 Desarrollo del sistema electrónico final.....	11
2.3.3 Desarrollo de la interfaz UART física:.....	19
2.4 Desarrollo del firmware.....	21
2.4.1 Fluid NC.....	21
2.4.2 Firmware de Interfaz UART.....	24
2.5 Desarrollo del software.....	25
Capítulo 3: Resultados.....	27
3.1 Resultados generales por sistema.....	27
3.2 Desempeño en el corte de distintos grosores de cartón.....	29
3.3 Características finales del producto.....	30
Capítulo 4: Análisis de Costos.....	32
4.1 Costos de materiales y mano de obra.....	33
4.2 Gestión de financiamiento Proyecto “Encartonate”.....	37
4.3 Amortización de la inversión.....	38
Capítulo 5: Discusión y Conclusión.....	40
5.1 Análisis de resultados obtenidos.....	40
5.2 Trabajos futuros.....	40
Capítulo 6: Literatura Citada.....	42
Anexo.....	43
Anexo 1: Detalle de las piezas diseñadas para la herramienta de corte.....	43
Anexo 2: Detalle de las piezas diseñadas para la estructura mecánica de la máquina.....	45

Lista de Figuras:

Figura 1. Diagrama en bloques de la máquina.....	3
Figura 2. Croquis 3D de la máquina.....	4
Figura 3. Detalle de herramienta de corte.....	6
Figura 4. Sistema mecánico.....	8
Figura 5. Diagrama en bloques de circuito prototípico.....	10
Figura 6. Arduino MEGA.....	11
Figura 7. Ramps 1.6.....	11
Figura 8. Distribución de pines y fotografía del módulo "Nodemcu ESP32-S2".....	12
Figura 9. Esquemático interno del módulo "Nodemcu ESP32".....	13
Figura 10. Esquemático del circuito del microcontrolador.....	15
Figura 11. Esquema del bloque de alimentación.....	17
Figura 12. Módulo DRV8825.....	17
Figura 13. Driver de motor DC con mosfet IRLR2905.....	18
Figura 14. Lector de tarjetas microSD.....	19
Figura 15. Diagrama divisor resistivo.....	20
Figura 16. Circuito esquemático de la interfaz física.....	21
Figura 17. Arduino Pro Micro y Display Full-Graphics utilizados.....	21
Figura 18. Interfaz Web de Fluid NC.....	22
Figura 19. Configuración general.....	23
Figura 20. Configuración SPI y micro SD.....	23
Figura 21. Configuración de tipo de control de motores.....	23
<i>Figura 22. Configuración de driver pwm.....</i>	<i>23</i>
Figura 23. Configuración de ejes de motores PAP.....	23
Figura 24. Configuración de interfaz UART.....	24
Figura 25. Definición de la clase "Procesador_mensaje".....	25
Figura 26. Cadena de software para generar un archivo de corte.....	26
Figura 27. Pruebas de corte.....	29
Figura 28. Detalle de bordes cortados con la máquina (izquierda) y con corte láser (derecha).....	30
Figura 29. Detalle de borde cortado por la cuchilla.....	30
Figura 30. Fotografía de la máquina de corte finalizada.....	30
Figura 31. Productos realizados por el emprendimiento Kanchay para el proyecto "Encartonate".....	38

Lista de Tablas:

Tabla 1. Tabla de componentes estructurales, mecánicos y electromecánicos de la máquina.....	5
Tabla 2. Resumen de piezas impresas en 3D de la herramienta de corte.....	7
Tabla 3. Resumen de piezas impresas en 3D de la estructura de la máquina.....	9
Tabla 4. Lista de componentes electrónicos del prototipo.....	10
Tabla 5. Detalle de pines del MCU ESP32.....	14
Tabla 6. Lista de componentes electrónicos del desarrollo final.....	16
Tabla 7. Características eléctricas ESP32.....	16
Tabla 8. Inflación porcentual por mes.....	32
Tabla 9. Índice de inflación acumulada desde febrero de 2022.....	32
Tabla 10. Costos de componentes mecánicos.....	33
Tabla 11. Costos de componentes electrónicos del prototipo.....	33
Tabla 12. Costos de componentes electrónicos del desarrollo final.....	34
Tabla 13. Costo de piezas impresas en 3D.....	35
Tabla 14. Costos de mano de obra.....	36
Tabla 15. Resumen de costos totales del proyecto.....	36
Tabla 16. Costos totales por unidad.....	37
Tabla 17. Productos con características similares.....	37
Tabla 18. Lista de piezas para impresión 3D de Herramienta de corte.....	44
Tabla 19. Lista de piezas para impresión 3D de la estructura de la máquina.....	47

Lista de Abreviaciones y Símbolos:

UART: Universal Asynchronous Receiver and Transmitter.

SPI: Serial Peripheral Interface.

MCU: Micro Controller Unit, Micro-controlador.

CNC: Computer Numerical Control, Control numérico por computador, hace referencia a un conjunto de máquinas que poseen como característica común el control a través de una computadora o microcontrolador de su movimiento y operación.

ESP32: es parte del código alfanumérico que representa a un conjunto de microcontroladores de la empresa Espressif, en este documento se hace referencia usando este nombre al modelo ESP32-S2.

string: cadena de caracteres.

float: tipo de variable que permite almacenar un valor decimal con punto flotante.

PWM: Pulse Width Modulation

RMT: Remote control transceiver.

DC: Direct Current. Corriente directa o corriente continua.

PAP: Paso a Paso, hace referencia a los motores de tipo paso a paso.

DIY: Do It Yourself o Hazlo tú mismo.

Capítulo 1: Introducción

1.1 Idea que dio origen al proyecto

La idea del proyecto surgió a partir del interés particular por las impresoras 3D y las máquinas de tipo CNC que permiten fabricar objetos a partir de un diseño digital; por otro lado participo hace más de 6 años en un emprendimiento llamado “Kanchay” [1] que creamos en conjunto con una amiga (Cecilia Illescas), en el cuál trabajamos con cartón de descarte y lo reutilizamos y reciclamos, a través de un proceso manual en objetos utilitarios para el hogar como cuadernos, luminarias y mobiliario de bajo porte.

A partir de estos intereses surgió la idea de diseñar y fabricar una máquina que permita automatizar el corte de cartón y utilizar como herramienta de diseño los programas de tipo CAD, con el objetivo de incorporarla al emprendimiento y así acelerar la producción y facilitar la creación de diseños más complejos.

1.2 Descripción del proyecto

El proyecto consiste en el diseño y fabricación integral de una máquina capaz de cortar cartón corrugado doble y otros materiales similares a partir de un diseño bidimensional generado en un software de diseño CAD.

1.3 Objetivos

Objetivo principal: diseñar y construir una máquina que permita el corte automático de cartón corrugado doble.

Objetivos secundarios:

- Incorporar la máquina en el proceso productivo del emprendimiento Kanchay para mejorar la calidad de sus productos y facilitar la producción en serie.
- Crear diseños complejos de corte y realizarlos con la máquina.
- Hacer una serie de pruebas con distintos tipos de materiales similares al cartón como papeles de distintos gramajes, cartulinas, cartones con diferentes grosores y estructuras y caracterizar el desempeño de esta en cada tipo de material.

1.4 Análisis de mercado

Público objetivo: El producto está orientado a emprendimientos y PYMES nacionales pequeñas, principalmente aquellas que su proceso productivo implique la necesidad de realizar cortes precisos, complejos y autónomos de cartón corrugado y papeles.

La cortadora de cartón al igual que los CNC están enfocados en la realización de productos altamente personalizados, no es recomendable para la producción en serie, ya que existen en el mercado otros productos que los superan en las características de velocidad y repetibilidad.

Prueba de concepto: La estructura básica de la cortadora, permite que su herramienta, la cuchilla oscilante, pueda desplazarse por un espacio que estará definido por las dimensiones de la estructura. Este mecanismo es ampliamente utilizado por otras máquinas como ser: impresoras 3D, CNC, plotters de corte, cortadoras láser; esto hace que su funcionamiento y uso sea fácilmente asimilado por el usuario, y en un corto período de aprendizaje pueda obtener resultados de esta.

Esta similitud con las máquinas listadas también abre la posibilidad de que la cortadora se convierta con pequeñas modificaciones en una máquina multipropósito con la capacidad de funcionar como una impresora 3D del tipo FDM, una máquina de corte por abrasión para materiales blandos, un grabador y cortador láser o un plotter gráfico.

Si no se realizan modificaciones a su herramienta o estructura, y nos enfocamos en la variación de los parámetros de corte como la velocidad de corte, la profundidad y la velocidad de movimiento, podemos generar distintos perfiles de corte y ampliar la capacidad de corte hacia distinto tipo de materiales como: vinilos, papel, cartulina, cartón compacto, polyfan fino, entre otros. Esto aumenta en gran medida su campo de aplicación.

Prueba de producto: Para cortar cartón corrugado con precisión y alta personalización, existen en el mercado actualmente dos opciones: corte láser o corte con cuchilla utilizando un CNC industrial.

El corte láser tiene una gran ventaja en la precisión del corte y el extremo detalle de sus trazos. Su desventaja principal es su elevado costo de este y la vida útil del láser, que al gastarse debe reemplazarse y es compleja tanto su disposición final como la adquisición de un nuevo tubo, ya que su costo es grande y sólo se consiguen fuera del país.

Los CNC industriales son máquinas muy robustas, que necesitan de gran espacio físico y consumen una gran cantidad de energía y están diseñadas para el moldeado por abrasión de una gran variedad de materiales, como metales y madera, además con una herramienta adecuada llamada cuchillo por arrastre, o con un cuchillo oscilante, puede cortar materiales como el cartón o realizar corte sobre tela, papel, plásticos, etc. Estas características hacen que su costo sea mucho más grande que otras máquinas y para la aplicación específica del corte de cartón están ampliamente sobredimensionados.

La principal ventaja de realizar un CNC a medida es que su costo se puede disminuir ampliamente y su tamaño puede ser ajustado de acuerdo con la aplicación o a la disponibilidad de espacio del cliente. Diseñando cuidadosamente la máquina, es posible disminuir la cantidad de materiales importados requeridos para su construcción y mantenimiento; disminuyendo aún más su costo y beneficiando además a la industria nacional.

Capítulo 2: Desarrollo

El desarrollo del proyecto se presenta primero a través de un diagrama de bloques general y luego en tres partes que están en relación con los sistemas que componen la máquina y que interactúan para determinar su funcionamiento: el sistema mecánico, el sistema electrónico y el sistema de software y firmware.

2.1 Diagrama en bloques del proyecto

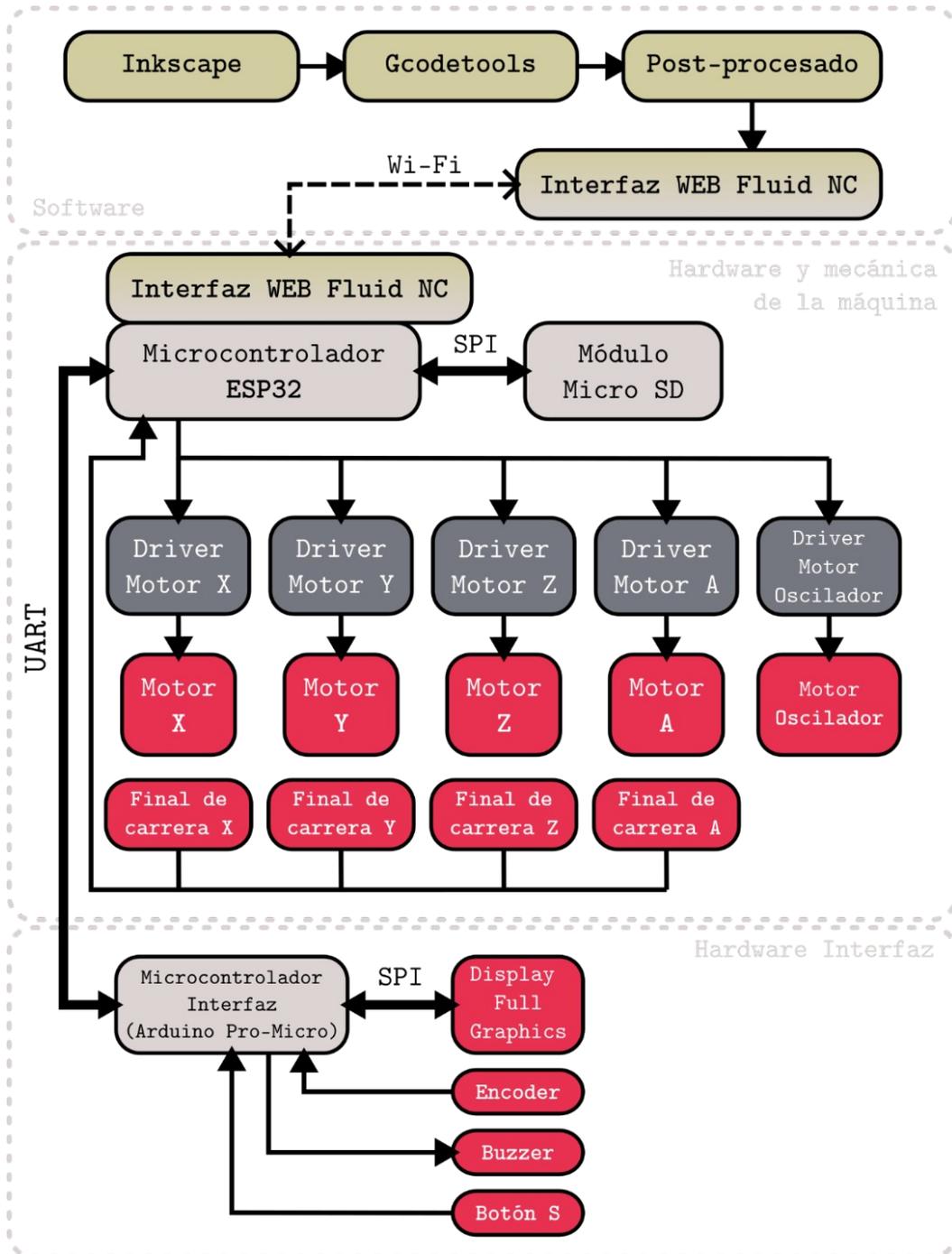


Figura 1. Diagrama en bloques de la máquina

2.2 Desarrollo del sistema mecánico

El sistema mecánico de la máquina se diseñó usando de referencia, por su similitud, las estructuras típicas utilizadas en el diseño de las impresoras 3D, muchos de éstos se encuentran liberados en la web y son de fácil acceso. De esta manera, utilizando el software libre FreeCAD [2], se diseñaron cada una de las piezas que la componen y que luego fueron impresas en 3D.

Para realizar este proceso en primer lugar se realizó un boceto a mano alzada tomando decisiones en cuanto a los componentes principales que debería tener para lograr el funcionamiento deseado y las medidas máximas que se deseaba para la misma, teniendo en cuenta que la máquina pueda colocarse sobre una mesa o escritorio mediano, se decidió que las medidas fueran de: 90 [cm] de largo, 60 [cm] de ancho y 20 [cm] de alto.

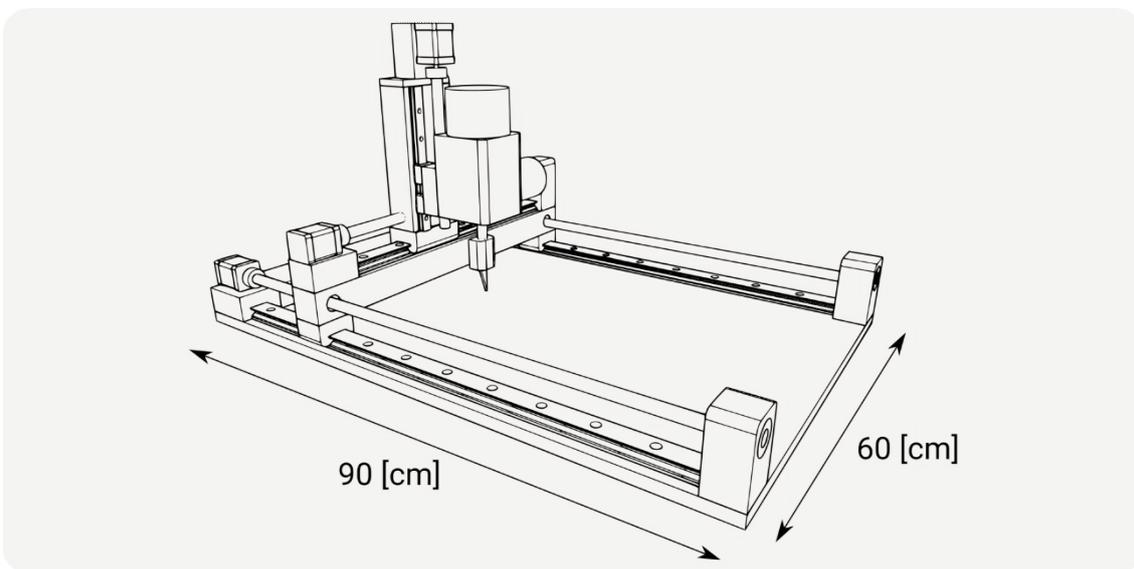


Figura 2. Croquis 3D de la máquina

Una vez terminado el boceto se comenzó el proceso de selección de los componentes mecánicos que se requerían para su fabricación, para ello se investigó sobre máquinas existentes que tuvieran funcionalidades similares, y especialmente aquellas no industrializadas sino realizadas y compartidas por la comunidad “maker” o la corriente de DIY (Hazlo tú mismo), cuyos diseños en su mayoría están gratuitamente ofrecidos en distintas páginas web y a menudo hacen uso de la fabricación aditiva como método constructivo principal.

Así se confeccionó la lista de materiales que figura a continuación.

Lista de componentes estructurales, mecánicos y electromecánicos	
Descripción	Cantidad
Base de madera melamina 90 [cm] x 60 [cm] x 18 [mm]	1
Barra Soportada Templada p/ sbr12 x 900 [mm]	2
Soporte SK8	8
Barra Templada y Cromada Ø8 x 600 [mm]	2
Barra Templada y Cromada Ø8 x 200 [mm]	2

Rodamiento SCS8LUU	4
Rodamiento SBR12UU	4
Tuerca THSL Ø8 [mm] Paso 2 [mm] Antibacklash	1
Varilla THSL Ø8 [mm] Paso 2 [mm] x 200 [mm]	1
Motor NEMA 17 0.9°	5
Motor DC RS-795	1
Correa GT2	3
Rodamiento 608	1
Rodamiento 625	3
Acople Flexible 5x8 mm	1
Polea GT2 Lisa 10mm x 5mm	6
Tuerca manguito para madera M6 x10u.	1
Repuesto para cutter bisturí x10u.	1

Tabla 1. Tabla de componentes estructurales, mecánicos y electromecánicos de la máquina

Se realizó la compra de los componentes indicados y se comenzó el proceso de diseño de las piezas para imprimirlas en 3D, para esto se buscaron las hojas de información de aquellas piezas que disponían de ella y para las que no, se utilizó un calibre para medir las dimensiones necesarias e incorporarlas en el diseño CAD.

El diseño se fue haciendo en partes, imprimiendo las piezas diseñadas y probando su funcionamiento, luego, en varias ocasiones se tuvo que realizar modificaciones al modelo y reimprimir las piezas. Se inició desde abajo hacia arriba, comenzando por la base y el eje Y, el cuál es el eje más largo de la máquina y el que soporta el resto de los ejes de movimiento, luego siguiendo con el eje X y el eje Z. La herramienta de corte se diseñó por separado y luego se incorporó con un adaptador a la estructura base de la máquina.

2.2.1 Diseño y fabricación de la herramienta de corte

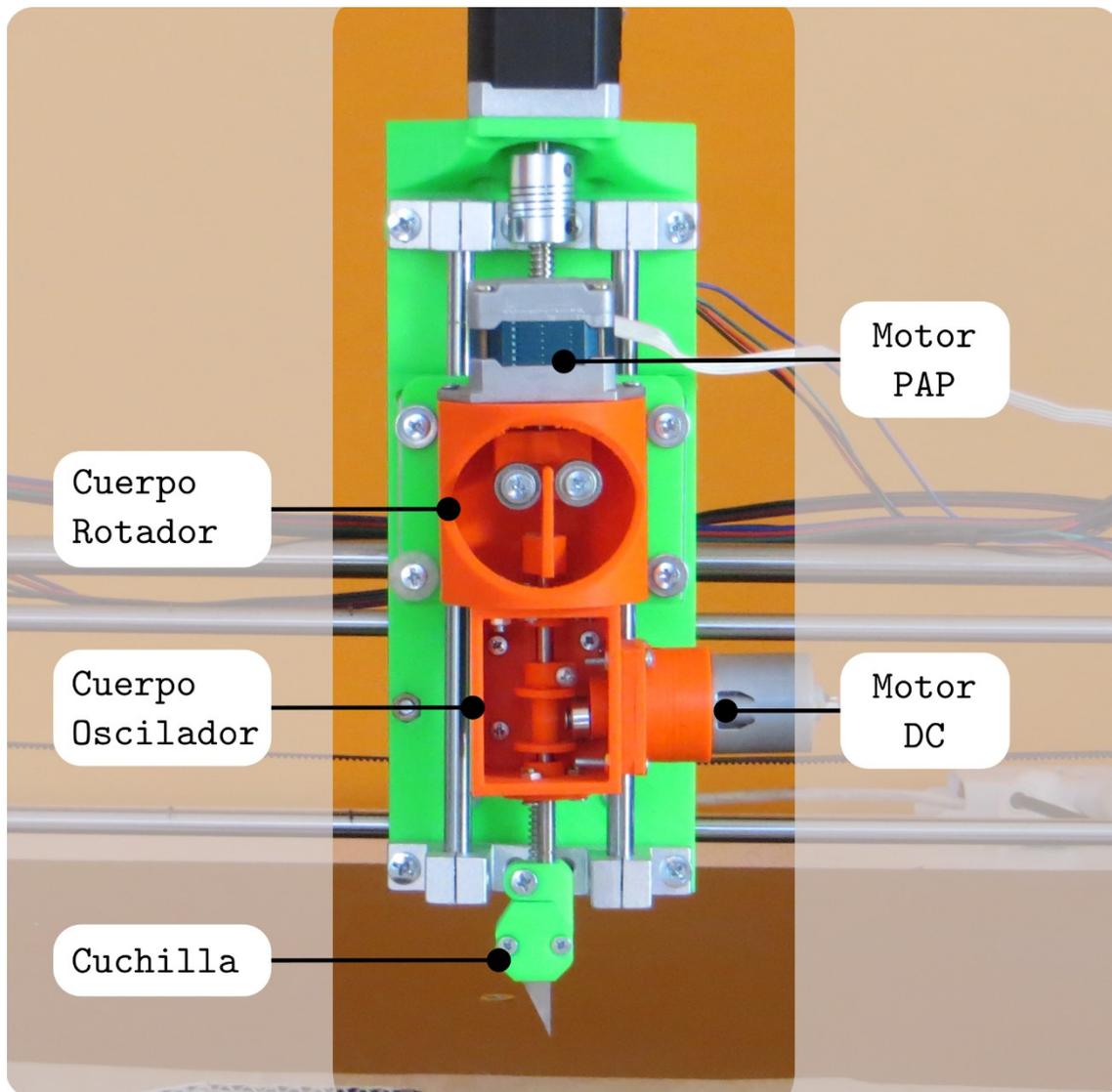


Figura 3. Detalle de herramienta de corte

Según puede observarse en la Figura 3, la herramienta de corte está compuesta por al menos tres bloques, la cuchilla, el cuerpo del elemento oscilador, y el cuerpo del elemento rotador. El diseño de esta [3] se realizó de forma modular para acoplar los dos movimientos que debe realizar en dos bloques individuales cada uno con su motor.

La cuchilla es un repuesto de cutter tipo bisturí, que se seleccionó debido a su amplia disponibilidad, ya que puede conseguirse en librerías, y también porque posee un ángulo cerrado que hace posibles cortes muy precisos. Está adosada al eje principal a través de una pieza dónde se fija con dos tornillos accesibles al frente para poder cambiarla fácilmente.

El cuerpo oscilador es el bloque encargado de generar el movimiento rápido oscilatorio que actuará sobre la cuchilla cortando efectivamente el material, al ser un movimiento rápido y unidireccional, se seleccionó un motor del tipo DC que puede alcanzar gran velocidad y es sencillo el control a través de PWM de su velocidad. Una pieza solidaria al eje del motor con un rodamiento excéntrico genera el movimiento oscilante sobre el eje

principal, que tiene fijada una pieza encargada de transformar el movimiento rotativo del motor en una oscilación lineal.

El cuerpo del rotador posee en la parte superior un motor Paso a Paso del tipo bipolar, que se encarga de fijar el ángulo de la cuchilla a través de una pieza que permite girar el eje mientras oscila libremente, ya que posee dos rodamientos que sostienen una pieza adosada al eje, pero a su vez habilitan el movimiento oscilatorio vertical sin fricción. El rotador se ocupa de orientar la cuchilla tangencialmente a la dirección de movimiento de la cuchilla cuando se realiza el trabajo de corte.

Resumen de piezas impresas en 3D de la herramienta de corte			
Nombre	Tiempo de impresión x u. [min]	Consumo de material x u. [gr]	Cantidad
Cuerpo del sistema rotador	125	37	1
Pieza central del rotador	16	4	1
Pieza móvil del rotador	12	2	1
Cuerpo central de la herramienta	86	22	1
Buje	7	2	2
Oscilador	13	2	1
Soporte del Motor Oscilador	27	8	1
Separador del Motor Oscilador	9	2	1
Excéntrico del Motor Oscilador	8	2	1
Pieza de sostén de la cuchilla de corte	17	4	1

Tabla 2. Resumen de piezas impresas en 3D de la herramienta de corte.

En el Anexo se incluye la lista piezas en conjunto con imágenes de estas.

2.2.2 Diseño y fabricación de la estructura de la máquina

La estructura mecánica de la máquina se diseñó pensando en maximizar el área de trabajo teniendo una altura baja, de tal manera de que se pueda colocar cómodamente en un escritorio mediano a grande.

Posee tres ejes de movimiento lineales controlados por motores paso a paso; el eje más largo (eje Y) de 90 [cm] posee dos motores que se deslizan utilizando correas tensadas y soportados por rodamientos abiertos sobre una varilla de acero templado y cromado de 12 [mm] de diámetro colocada sobre un soporte de aluminio; sobre este eje se mueve el eje X, que posee una longitud de 60 [cm], creado con varillas de acero de 8 [mm], más una varilla de refuerzo de 12 [mm] de diámetro, en el caso de este eje se utilizan rodamientos cerrados, y el movimiento se transfiere al igual que en el eje Y, a través de una correa tensada y generado por un solo motor PAP; el eje Z mueve la herramienta de arriba abajo, y posee una longitud total de 20 [cm], el movimiento se genera por un motor, una varilla roscada de perfil rectangular y una tuerca del mismo tipo solidaria al carro de la herramienta transforma este movimiento rotativo del motor en un movimiento lineal.

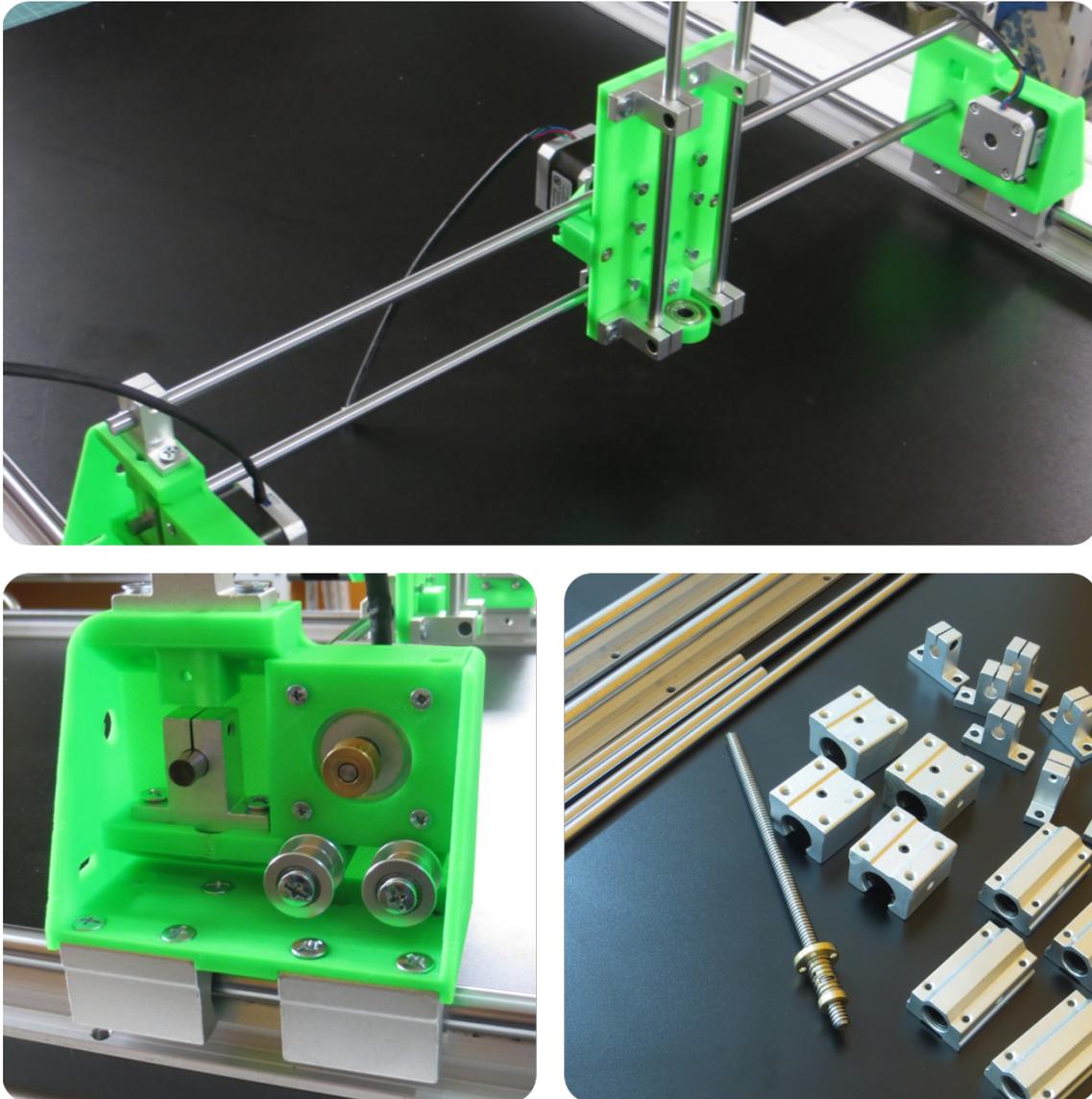


Figura 4. Sistema mecánico

Resumen de piezas impresas en 3D de la estructura de la máquina			
Nombre	Tiempo de impresión x u. [min]	Consumo de material x u. [gr]	Cantidad
Carro Eje Y	254	71	2
Soporte de tensores Eje Y	63	20	4
Separadores Eje Y	4	1	2
Tensores móviles de correas	12	3	6
Soporte Eje Auxiliar del Eje Y	54	17	2
Pieza Central del carro del Eje X	177	55	1
Soporte del motor del Eje X	51	14	1
Separador de poleas del Eje X	2	1	2
Soporte Eje Auxiliar del Eje X	74	23	1
Soporte del motor del eje Z	171	55	1
Acople del soporte del motor del eje Z al Eje X	16	5	1

Carro Eje Z	79	23	1
Soporte inferior varilla roscada del Eje Z	16	5	1
Pieza de acople de herramienta de corte al Eje Z	55	16	1
Pata de sujeción de material	13	4	8
Perilla de sujeción de material	9	3	8

Tabla 3. Resumen de piezas impresas en 3D de la estructura de la máquina.

En el Anexo se incluye la lista piezas en conjunto con imágenes de estas.

2.3 Desarrollo del sistema electrónico

Para diseñar el sistema electrónico se diagramó un prototipo más simple y, posteriormente, tras verificar el correcto funcionamiento de las partes electrónicas y mecánicas, se construyó la placa de circuito que se corresponde con el diagrama de bloques presentado anteriormente. En los siguientes puntos se detallarán ambos circuitos y se expondrán las características y resultados encontrados con ambos.

2.3.1 Desarrollo del prototipo del sistema electrónico

Al igual que el diseño mecánico, para realizar el diseño electrónico del prototipo se comenzó tomando de referencia los esquemáticos de los circuitos de las impresoras 3D, y a partir de ellos se diagramó un circuito con el que se hicieron las pruebas de funcionamiento de la máquina, utilizando para esto módulos típicos que se utilizan para estas máquinas y también un firmware open-source de uso libre, comúnmente utilizado llamado Repetier [4].

Una de las ventajas de utilizar como firmware Repetier frente a otros firmwares conocidos como Marlin, es que el software con el que se puede manipular dispone de un servidor web (Repetier Server) que se puede instalar en una computadora y así controlar la máquina y subir archivos a través de una red local, lo que supone mayor facilidad a la hora de controlar, gestionar y monitorear varias máquinas conectadas. Particularmente brinda la ventaja de permitir transferir los archivos de modo inalámbrico.

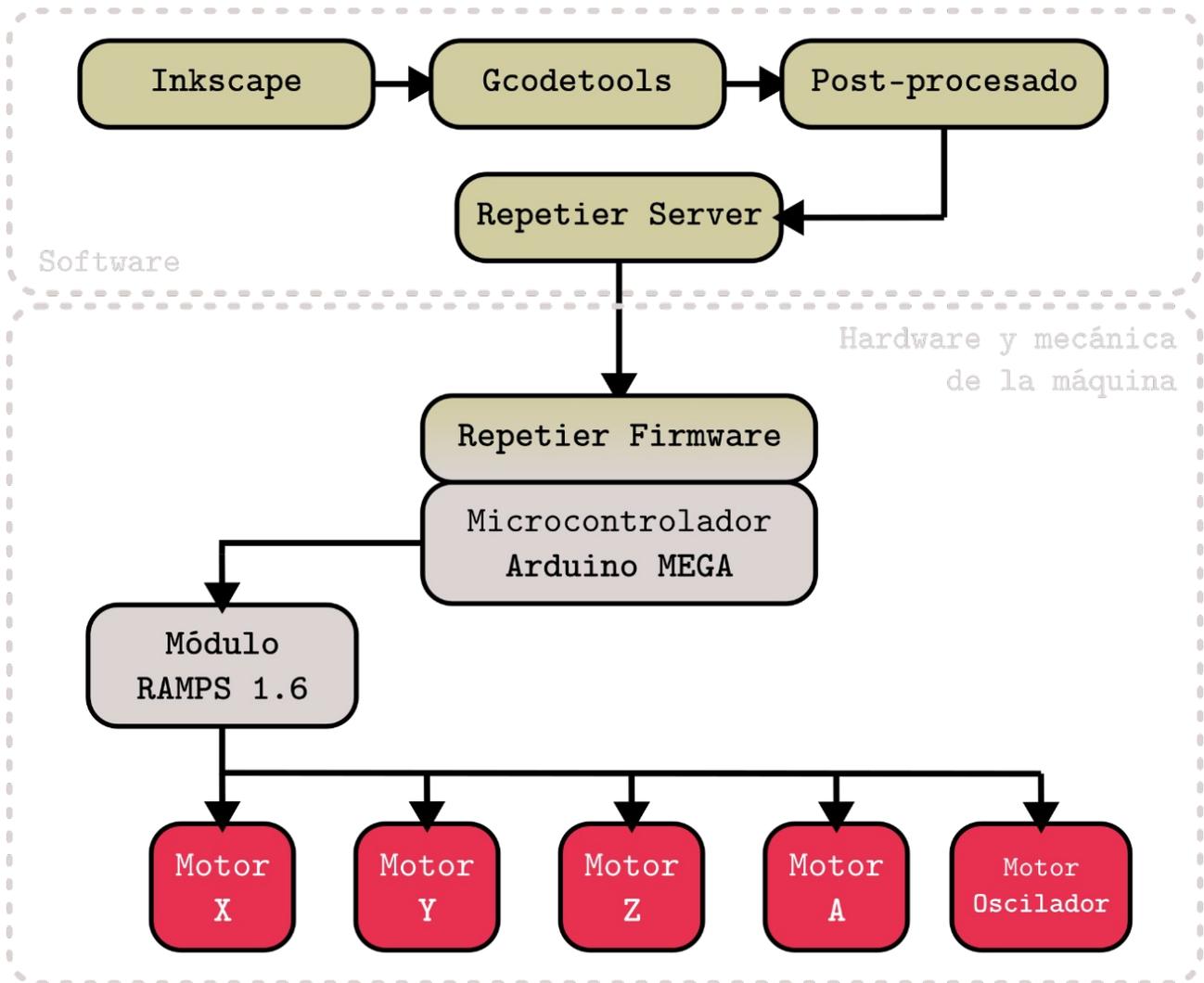


Figura 5. Diagrama en bloques de circuito prototípico

Lista de componentes electrónicos del prototipo	
Descripción	Cantidad
Arduino MEGA	1
Drivers DRV8825	5
Módulo RAMPS 1.6	1
Fuente 12 [V] 20 [A]	1
Cable 4 polos x 24 AWG x10m.	1

Tabla 4. Lista de componentes electrónicos del prototipo

Microcontrolador Arduino Mega:

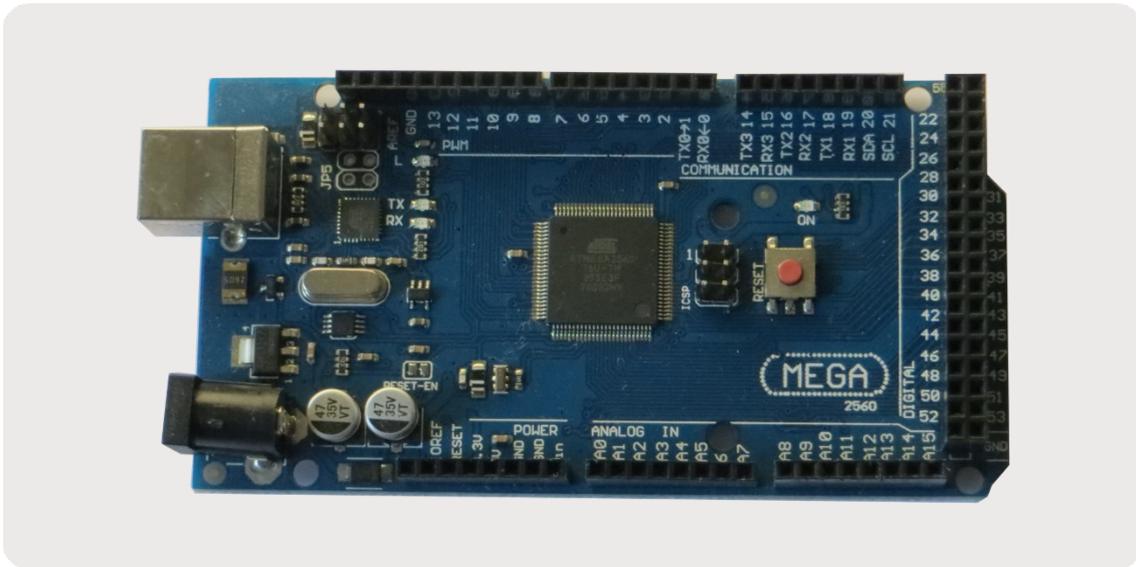


Figura 6. Arduino MEGA

RAMPS 1.6:

El “Ramps 1.6” [5] es un módulo diseñado para el Arduino MEGA que puede insertarse sobre el mismo ya que la disposición de pines coincide exactamente y contiene sockets para colocar drivers de motores paso a paso bipolares del tipo DRV8825 o A4988, también dispone de tres drivers con mosfets de potencia y una terminal con tornillos que permite conectar la fuente de tensión.

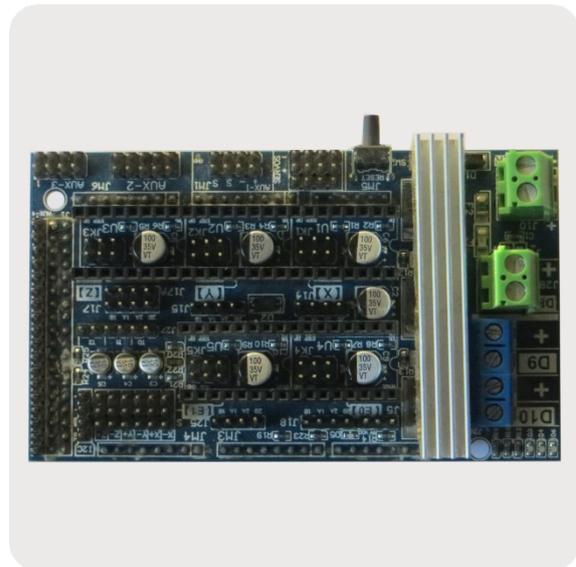


Figura 7. Ramps 1.6

El Arduino MEGA junto con el módulo Ramps en cualquiera de sus versiones, es comúnmente utilizado en la fabricación de impresoras 3D DIY y se encuentra ampliamente disponible en el mercado local a bajo costo.

2.3.2 Desarrollo del sistema electrónico final

Para el desarrollo del circuito final se tuvieron en cuenta muchas de las características observadas en el circuito prototipo, así se tomaron las siguientes consideraciones en cuanto al diseño de este:

- **Conexión inalámbrica:** Se desea mantener la misma capacidad de gestión, monitoreo y control inalámbricos por red local que posee el prototipo.

- **Independencia de otros sistemas:** Es necesario añadir una tarjeta de memoria que permita almacenar los archivos de corte en la misma máquina para así no depender de una computadora conectada a la misma para que funcione.
- **Interfaz de control y monitoreo física:** Para lograr la independencia detallada en el punto anterior, es necesario añadir una interfaz que permita observar el estado de la máquina y también realizar acciones como realizar un corte o mover la herramienta.
- **Reutilización de la mayoría de los componentes utilizados para el prototipo.**

Teniendo en cuenta estas características, se decidió utilizar para el proyecto el microcontrolador ESP32, principalmente debido a sus características inalámbricas y su mayor velocidad de procesamiento, y también por su previa utilización en otros proyectos personales.

Placa de desarrollo NODEMCU ESP32:

El MCU ESP32 es una serie de microcontroladores que su fabricante *Espressif* ha comercializado a lo largo de estos años, y también ha mejorado en nuevas versiones. La versión utilizada en el proyecto se denomina ESP32 WROOM 32, este nombre hace referencia a un módulo que contiene en su interior el chip ESP32-D0WDQ6, una memoria EEPROM y una antena entre otras cosas. Este módulo posee un formato de soldadura superficial que habitualmente se comercializa soldado en placas de desarrollo que contienen otros componentes como una interfaz USB-Serie, y un circuito de alimentación para el microcontrolador.

Para el proyecto utilizaremos una placa de desarrollo denominada “Nodemcu ESP32” [6] que se consigue fácilmente en el mercado local. En este documento nos referiremos a esta placa de desarrollo como “Módulo” o MCU indistintamente.

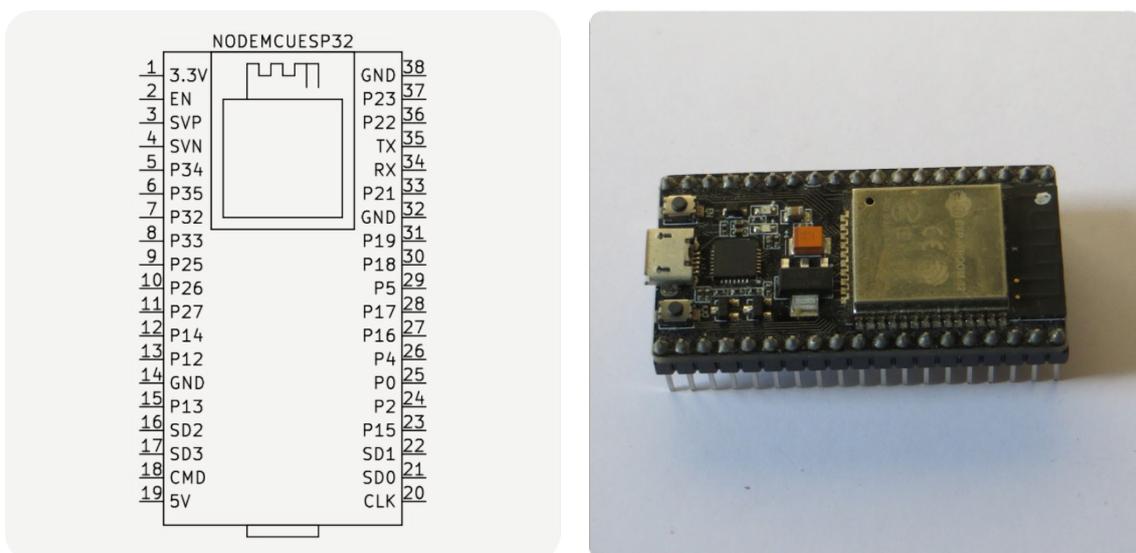


Figura 8. Distribución de pines y fotografía del módulo "Nodemcu ESP32-S2"

Esquema del circuito del módulo (Obtenido de la hoja de datos [7]):

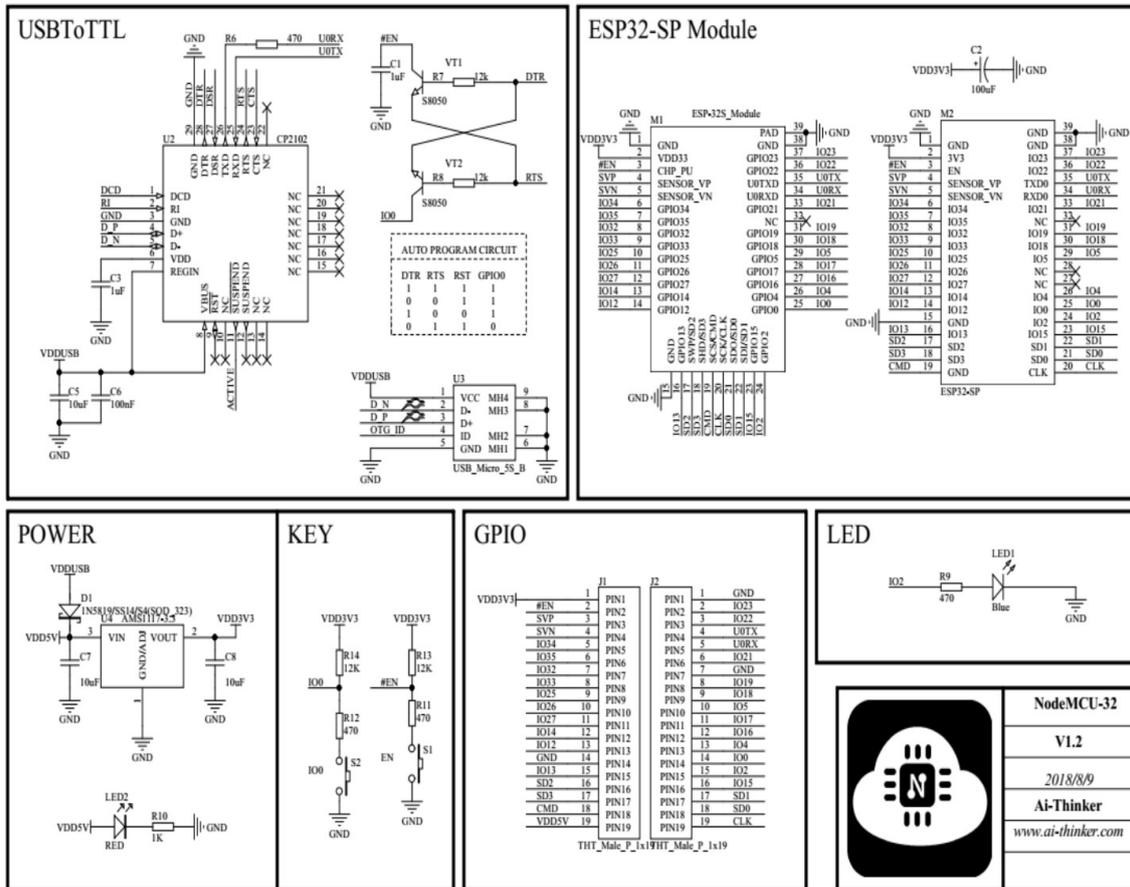


Figura 9. Esquemático interno del módulo "Nodemcu ESP32"

Análisis del esquemático:

Observando el mismo, podemos ver que el módulo contiene todos los dispositivos necesarios para la programación del microcontrolador a través del integrado CP2102 y el botón conectado al GPIO0. También nos otorga la posibilidad de alimentarlo con 5 [V] utilizando el regulador incorporado AMS1117 de 3.3V, además de un led rojo que indica que el módulo está energizado y un led azul conectado al GPIO02 del ESP32, por último, un botón de RESET.

Descripción de pines:

En la siguiente tabla se detallan todos los pines que el módulo posee y sus características, además se indica el tipo, de acuerdo a si puede utilizarse como entrada, salida o ambos.

Nº	Nombre	Tipo	Funciones	Características especiales
1	3.3V	-		
2	RESET	-		
3	gpio.36	INPUT	ADC1_CH0	
4	gpio.39	INPUT	ADC1_CH3	
5	gpio.34	INPUT	ADC1_CH6	
6	gpio.35	INPUT	ADC1_CH7	

7	gpio.32	I/O	ADC1_CH4	
8	gpio.33	I/O	ADC1_CH5	
9	gpio.25	I/O	ADC2_CH8, DAC1	
10	gpio.26	I/O	ADC2_CH9, DAC2	
11	gpio.27	I/O	ADC2_CH7	
12	gpio.14	I/O	ADC2_CH6, HSPI_CLK	Emite señal PWM en el Boot.
13	gpio.12	I/O	ADC2_CH5, HSPI_MISO	Strapping Pin. Falla al bootear si está en HIGH.
14	GND	-		
15	gpio.13	I/O	ADC2_CH4, HSPI_MOSI	
16	gpio.9	X		Conectado a la memoria flash interna
17	gpio.10	X		Conectado a la memoria flash interna
18	gpio.11	X		Conectado a la memoria flash interna
19	5V	-		
20	gpio.6	X		Conectado a la memoria flash interna
21	gpio.7	X		Conectado a la memoria flash interna
22	gpio.8	X		Conectado a la memoria flash interna
23	gpio.15	I/O	ADC2_CH3, HSPI_CS	Strapping Pin. Emite señal PWM en el Boot,
24	gpio.2	I/O	ADC2_CH2	Strapping Pin. Conectado al LED azul del módulo. Debe estar en LOW para ingresar al modo FLASH.
25	gpio.0	I/O	ADC2_CH1	Strapping Pin. Emite señal PWM en el Boot. Debe estar en LOW para ingresar al modo FLASH.
26	gpio.4	I/O	ADC_CH0	Strapping Pin.
27	gpio.16	I/O	UART2_RXD	
28	gpio.17	I/O	UART2_TXD	
29	gpio.5	I/O	VSPI_CS	Strapping Pin. Emite señal PWM en el Boot,
30	gpio.18	I/O	VSPI_CLK	
31	gpio.19	I/O	VSPI_MISO	
32	GND	-		
33	gpio.21	I/O	I2C_SDA	
34	gpio.3	INPUT		HIGH en Boot.
35	gpio.1	OUTPUT		Salida de Debug en Boot.
36	gpio.22	I/O	I2C_SCL	
37	gpio.23	I/O	VSPI_MOSI	
38	GND	-		

Tabla 5. Detalle de pines del MCU ESP32

Diseño del circuito y mapeo de pines:

Para diseñar el circuito se comenzó analizando en una visión general el diagrama de bloques indicado anteriormente; en esta visión global de la máquina queda claro que la misma tiene 4 ejes de movimiento, X, Y, Z y A, cada eje tiene al menos un motor y un final de carrera, (el eje Y posee dos motores en paralelo debido a sus características físicas), también tenemos un motor extra para generar el movimiento oscilatorio, por lo tanto necesitamos **controlar estos motores y sensar los finales de carrera**.

Además, debemos guardar los archivos de corte para ser procesados por la máquina independientemente de una PC, para ello se decidió incorporar una tarjeta de memoria a través de un módulo micro-SD utilizando una **interfaz SPI**.

Por otro lado, se decidió también incorporar una interfaz física a la máquina, la que estará implementada por otro microcontrolador, que se comunicará con el principal a través de una **interfaz UART**. Esto lo analizaremos en profundidad más adelante.

Teniendo en cuenta las interfaces que debemos utilizar, para decidir qué pines utilizaríamos del módulo primero seleccionamos aquellos pines dedicados exclusivamente a las interfaces UART (27 y 28), y a la interfaz VSPI (29, 30, 31 y 37).

Luego vemos qué periféricos necesitan sólo ser leídos, como los finales de carrera, así que para ellos seleccionaremos aquellos pines que sólo pueden ser configurados como entradas (3, 4, 5 y 6).

Ahora nos queda decidir qué pines utilizaremos para los 5 drivers de los motores. Para controlar los drivers de los motores paso a paso, necesitamos al menos 2 pines de salida digital para cada uno, uno para la señal STEP y otro para la señal DIR, que controla la dirección del motor. Para el driver con mosfet, sólo necesitamos 1 salida PWM.

Asignamos esos pines evitando los pines de Strapping que pueden alterar el Boot del MCU, así seleccionamos los pines 7, 8, 9, 10, 11, 12, 15, 33 y 36 para estas señales. Analizaremos esta selección y las características de los drivers más adelante.

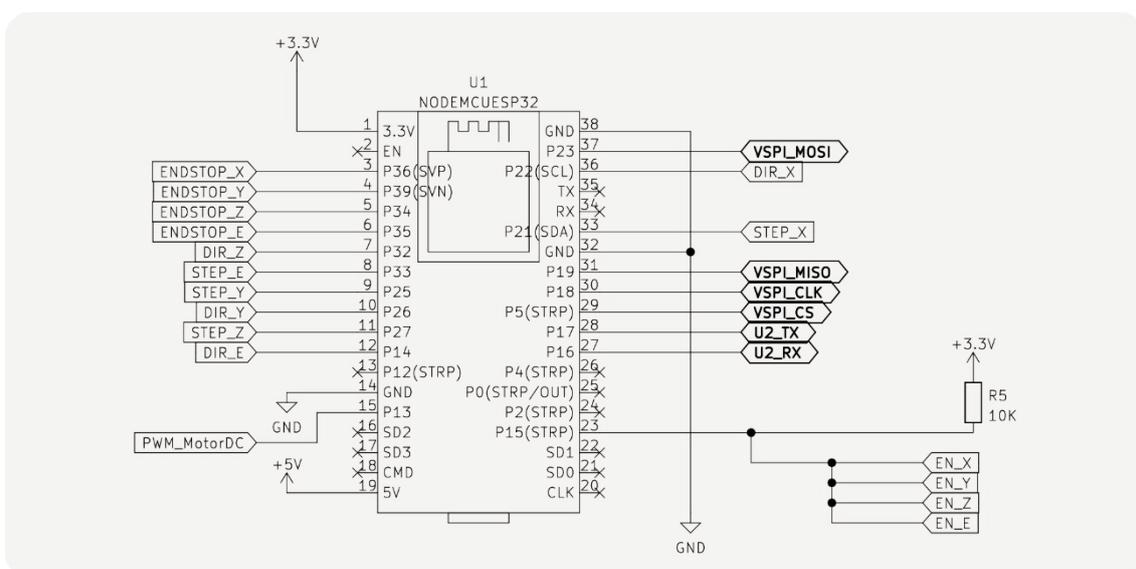


Figura 10. Esquemático del circuito del microcontrolador

Lista de componentes electrónicos del desarrollo final

Descripción	Cantidad
Módulo Nodemcu ESP32	1
Arduino Pro-Micro	1
Módulo Step-Down 2[A] "Mini-360"	1
Módulo microSD	1
Módulo Display Full Graphics	1
Placa perforada 9x15 [cm]	1
Placa perforada 7x9 [cm]	1
Capacitor Electrolítico 47 [uF] 16[V]	6
IRL520N	1
Resistencia 10k 1/4 [W]	1
Resistencia 1k 1/4 [W]	1
Resistencia 2,2k 1/4 [W]	1
Cable alimentación 220[V] 10[A]	1
Conector JST Hembra 4 pin	5
Conector JST Hembra 3 pin	4
Conector JST Macho 4 pin	5
Conector JST Macho 3 pin	4

Tabla 6. Lista de componentes electrónicos del desarrollo final.

Bloque de alimentación:

Para alimentar el sistema completo se utilizó la misma fuente que para el circuito prototipo, una fuente de 12 [V] y 20 [A].

Esta tensión es necesaria para alimentar los motores, pero debe ser adaptada para los elementos lógicos del circuito. Observando la hoja de datos del microcontrolador ESP32, podemos ver sus características eléctricas:

Symbol	Parameter	Min	Typical	Max	Unit
VDD33	Power supply voltage	3.0	3.3	3.6	V
I_{VDD}	Current delivered by external power supply	0.5	-	-	A
T	Operating ambient temperature	-40	-	85	°C

Tabla 7. Características eléctricas ESP32

Vemos que necesita una fuente de energía de mínimo 0.5 [A] y entre 3.0 y 3.6 [V].

Recordando que estamos trabajando con el módulo "Nodemcu ESP32", vemos que este contiene un regulador lineal interno de 3.3[V] cuyo modelo es AMS1117.

Si observamos el esquemático vemos que el integrado CP2012 que nos permite comunicarnos a través de USB con la PC y así programar el módulo, recibe como tensión de entrada el voltaje del puerto USB, y a través de un diodo evita que la tensión presente en el rail de VDD5V alcance el rail VUSB, sin embargo se ha visto que en varios modelos de este módulo ambos railes están conectados, esto supondría que el integrado CP2012 esté alimentado también por el pin de 5[V], y según su hoja datos, la tensión máxima que soporta es de 6[V].

El estado lógico del pin DIR determina la dirección en que se moverá el motor y una señal en el pin STEP indica que el motor debe realizar un paso o una fracción de este de acuerdo a lo configurado con los pines que ajustan el divisor de micro-stepping.

Driver de motor Oscilador de la herramienta:

En el caso del motor oscilador de la herramienta, el mismo es un motor DC, que podemos controlar con un mosfet y una señal PWM que genera el MCU.

Como el MCU que estamos utilizando trabaja a 3.3[V], para no complejizar el circuito de disparo del mosfet se buscó uno del tipo Logic Level. Un modelo que se consiguió fácilmente y a un bajo precio es el integrado IRLR2905, cuyas características eran más que suficientes para la aplicación.

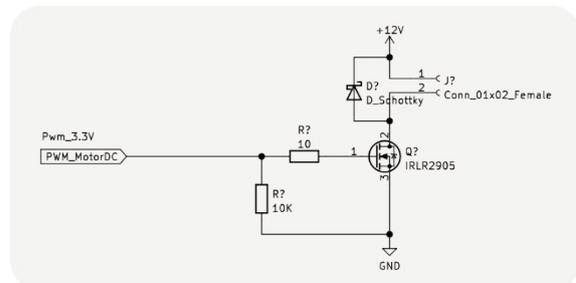


Figura 13. Driver de motor DC con mosfet IRLR2905

Los mosfet Logic Level necesitan una tensión mucho menor que los comunes para entrar la región de saturación, es en dicha región donde debemos trabajar en este caso para que la resistencia del canal del semiconductor sea mínima y no disipe energía en forma de calor.

El circuito utilizado es el indicado en la *Figura 13*, el mismo consiste en el elemento de conmutación que es el mosfet y se añaden dos resistencias y un diodo de protección externo que lo protege de tensiones inversas que se generan debido a la naturaleza inductiva del motor DC, que muchas veces no es necesario ya que se incorpora dentro del mosfet. La resistencia de 10K funciona como resistencia de pull-down para apagar el mosfet en ciertos casos en que podría permanecer encendido y la resistencia de 10 ohm en serie con la compuerta limita las pequeñas corrientes que se generan para cargar la misma en la conmutación, protegiendo la salida del MCU.

Finales de carrera:

Generalmente la máquina desconoce la posición relativa del cabezal al sistema de coordenadas definido por la estructura física, esto es porque no dispone de sensores que den información espacial al firmware.

Hay múltiples soluciones a este problema utilizando distintos tipos de sensores que dan mayor o menor información espacial, y de esto depende su complejidad y costo de aplicación.

Para este proyecto se decidió utilizar finales de carrera o endstops, que son llaves tipo switch que pueden colocarse en el origen y/o final de los ejes de movimiento; la máquina busca la activación de estas llaves para detectar estos puntos y definir el origen o cero de cada eje de movimiento. Esta operación se realiza cada vez que la máquina se enciende, previo a cada corte, o en el caso que se mueva el cabezal manualmente.

Si se colocan dos finales de carrera por eje, tanto al principio como al final, se aporta más seguridad a la máquina ya que evita que intente moverse accidentalmente por fuera de

los límites físicos dañando el sistema mecánico en consecuencia y también aporta información sobre la extensión total de cada eje, que de utilizarse un solo final de carrera debe indicarse al firmware manualmente.

Interfaces SPI y UART

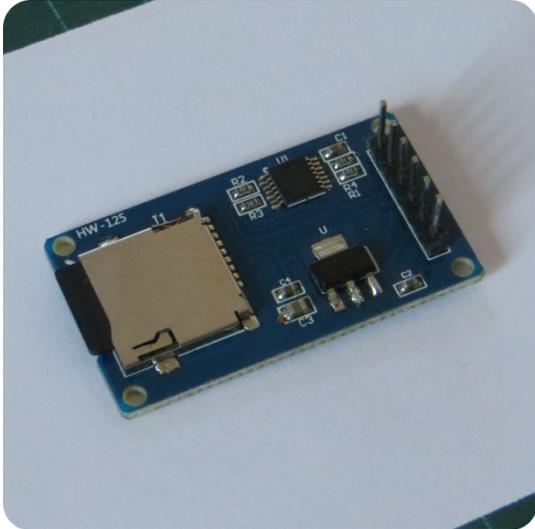


Figura 14. Lector de tarjetas microSD

El módulo utilizado para la lectura de la tarjeta de memoria por SPI, se conoce comercialmente como “MicroSD Card Adapter”.

Este módulo contiene un socket para la tarjeta micro sd, un integrado conversor de niveles y un regulador lineal de 3.3[V] idéntico al que se encuentra en el módulo del microcontrolador. Cabe destacar que como la lógica del MCU que se está utilizando funciona a 3.3[V], no sería necesario un conversor de nivel ni el regulador, sin embargo, se decidió utilizar este módulo por su amplia disponibilidad, ya que es muy utilizado en conjunto con los MCU's de la serie Arduino cuya lógica funciona a 5[V].

En el caso del puerto UART para establecer la comunicación bidireccional con la interfaz física, se añadió un conector de 4 pines (GND, 5[V], TX y RX) que se utiliza para conectar la misma al ESP32 y alimentarla con la misma fuente de 5[V].

2.3.3 Desarrollo de la interfaz UART física:

Tal como fue explicitado anteriormente, se decidió añadir a la máquina una interfaz física. Esto supone muchas ventajas en relación con el sistema prototipo, y también le añade algunas mejoras al sistema que podemos listar:

- Permite observar el estado de la máquina en todo momento, incluso emitir un sonido o un mensaje de alarma en caso de percibir algún error de funcionamiento.
- Permite controlar la posición de la máquina sin entrar a la interfaz web, dando mayor comodidad al usuario.
- Habilita al usuario la posibilidad de realizar procesos sin intervención de una PC, por ejemplo, repetir trabajos previos guardados en la memoria.

La utilización específica del puerto UART y no otros protocolos de comunicación como SPI o I2C para la comunicación, se justifica analizando el firmware Fluid NC y observando que el mismo permite la declaración de un puerto extra para enviar o recibir comandos desde su configuración. Por lo tanto, para añadir esta funcionalidad no fue necesario modificar el firmware sino sólo configurarlo adecuadamente.

La elección del microcontrolador para la interfaz fue en primer lugar debido a la disponibilidad del Arduino Pro-Micro, sin embargo, para esta aplicación en particular podrían utilizarse módulos más económicos con similares características.

Un módulo similar basado en el MCU Atmega328 es el “Arduino Pro Mini”, el mismo viene en versiones de 5 y 3.3[V], la única desventaja es que no posee un conversor usb-serie incorporado, que para el caso de una aplicación final es ideal, pero como se debían realizar pruebas y reprogramaciones del prototipo utilizando el puerto, no se optó por este MCU.

El Arduino Pro-Micro trabaja con una lógica de 5[V], por lo tanto, para comunicarse a través del puerto UART con el ESP32 es necesario convertir los niveles a 3.3[V], esto puede hacerse con un conversor de nivel, pero también existe una solución más práctica.

Debido a que el protocolo utilizado por UART utiliza dos pines, uno para transmitir y otro para recibir, se puede resolver el problema con un simple divisor resistivo en el pin de transmisión del Arduino, que disminuya la tensión y la adapte a los niveles de 3.3[V] del ESP32.

Calculamos el valor del divisor resistivo para

este caso, así: $\frac{v_2}{v_1} = \frac{3.3[V]}{5[V]} = 0.66$

La ecuación del divisor resistivo es la

siguiente: $\frac{v_2}{v_1} = \frac{R_2}{R_1 + R_2}$

Si fijamos R1 en 1[K], podemos hallar el

valor de R2 como: $\frac{(0.66 * R_1)}{(1 - 0.66)} = R_2 = 1.94 \text{ } \Omega$

Elegimos una resistencia comercial cercana al valor hallado que sea menor que el mismo para no superar el máximo de 3.3[V], de esta manera asignamos a R2 un valor de 1.8 Ω .

Para el caso del pin receptor del Arduino, este recibe la señal de 3.3[V] del ESP32 directamente, y puede deducir correctamente los niveles de tensión, ya que si observamos la hoja de datos del MCU Atmega32U4 (que posee internamente el módulo Pro-Micro), en el apartado de características eléctricas, encontramos que el valor mínimo de tensión que puede ser leído como un estado “ALTO” es de $0.2 * VCC + 0.9$ [V], en nuestro caso VCC es 5[V], por lo tanto este valor es de 1.9[V], es decir que las señales de 3.3[V] son entendidas como un estado ALTO y la comunicación se produce sin problemas.

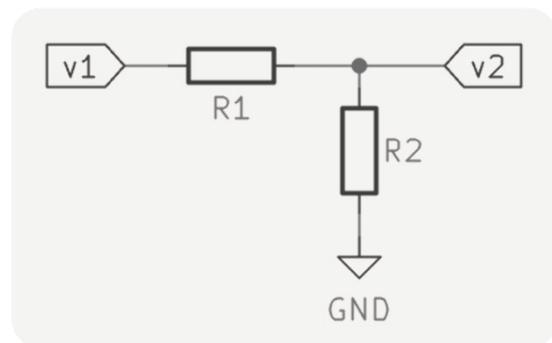


Figura 15. Diagrama divisor resistivo

Circuito esquemático:

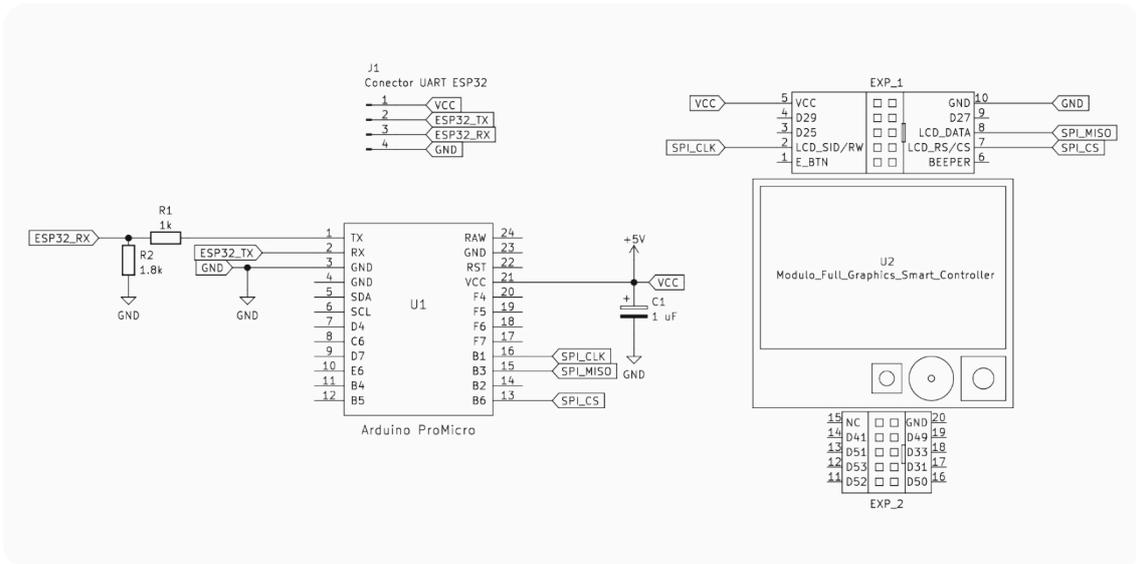


Figura 16. Circuito esquemático de la interfaz física

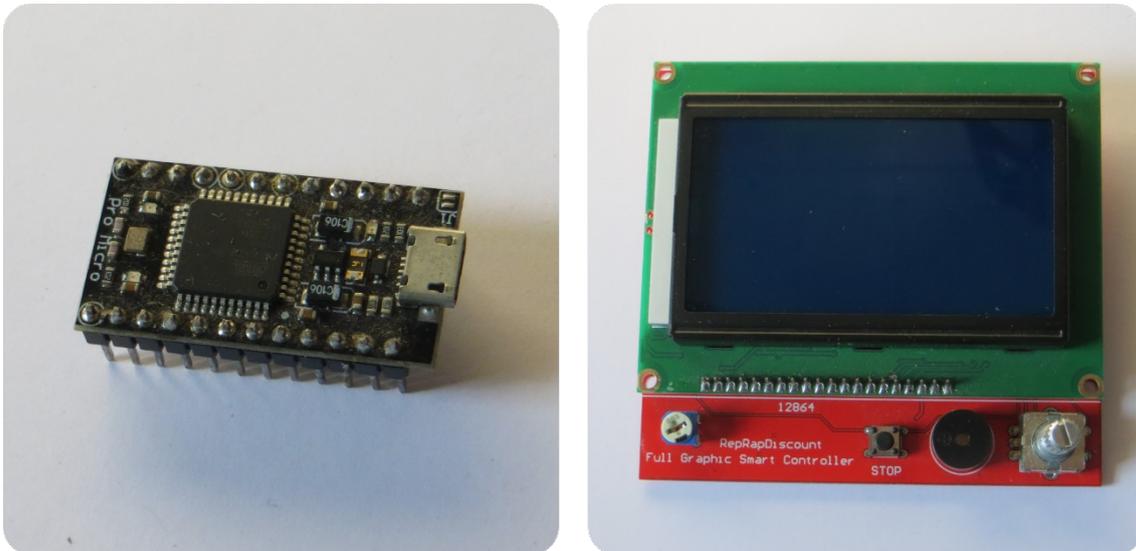


Figura 17. Arduino Pro Micro y Display Full-Graphics utilizados

2.4 Desarrollo del firmware

En este apartado se detalla por un lado la aplicación y configuración de un firmware Open-Source llamado “FluidNC” utilizado en el microcontrolador principal de la máquina y el desarrollo del firmware para la Interfaz UART.

2.4.1 Fluid NC

Fluid NC [8] es un firmware para máquinas CNC optimizado para el ESP32, es una nueva generación de un firmware anterior llamado GRBL_ESP32. El mismo incluye una interfaz de usuario Web y una gran flexibilidad para operar con una variedad de tipos de máquinas diferentes. Esto incluye la habilidad de controlar máquinas con múltiples tipos de herramientas, por ejemplo: láseres, herramientas rotativas, intercambiadores de herramientas, etc.

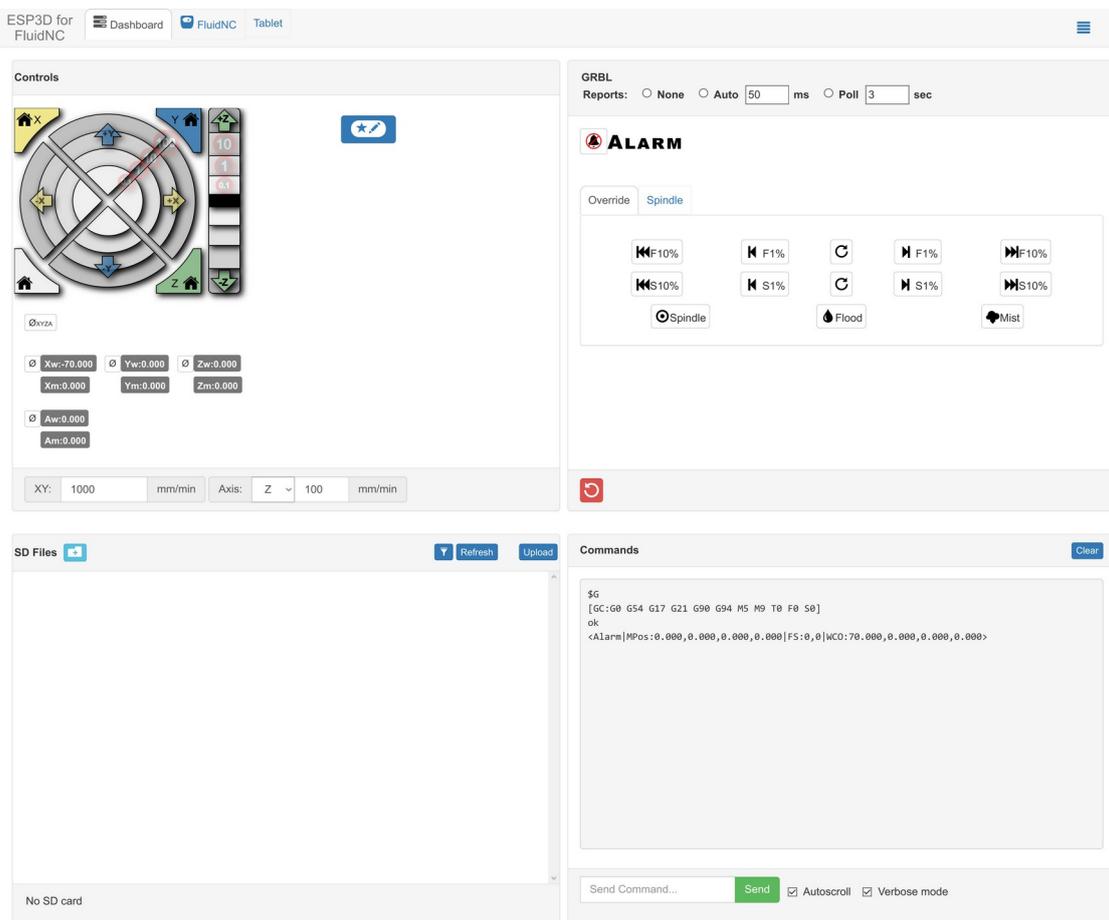


Figura 18. Captura de pantalla de la Interfaz Web de Fluid NC.

Configuración de Fluid NC:

El firmware Fluid NC fue diseñado para facilitar la tarea de configuración para diversas máquinas con distinta cantidad de ejes, tipos de drivers y periféricos, y nos da la posibilidad de descargar el firmware compilado y realizar la configuración a través de un archivo de texto con extensión *.yaml, que puede ser cargado desde la misma interfaz web del firmware, posterior a grabar el mismo en el microcontrolador.

Esta opción no sólo evita el paso de compilar el firmware completo, sino también, la posibilidad de realizar ajustes finos mientras la máquina esté en funcionamiento o también añadir ejes, herramientas con distintas características o sensores extra, sin necesidad de volver a compilar y grabar el firmware.

A continuación, se detallarán los campos que fueron configurados en el archivo *.yaml para adaptar el firmware a la configuración especial de la máquina.

Configuración general

El archivo *.yaml se estructura utilizando tabulaciones para definir áreas de configuración, hay opciones que se pueden configurar en la base del archivo, y otras que pertenecen a distintas áreas.

Dentro de la configuración general podemos asignar un nombre y un tipo de placa que estarán expuestos en la interfaz.

```
name: "Cortadora de carton ESP32"  
board: "ESP32 Nodemcu"
```

Figura 19. Configuración general

Configuración de la interfaz SPI para la tarjeta de memoria

```
spi:  
  miso_pin: gpio.19  
  mosi_pin: gpio.23  
  sck_pin: gpio.18  
  
sdcard:  
  cs_pin: gpio.5  
  card_detect_pin: NO_PIN  
  frequency_hz: 800000
```

Figura 20. Configuración SPI y micro SD

Para habilitar la tarjeta de memoria a través de SPI, definimos los pines utilizados por el protocolo y luego definimos el uso de una tarjeta de memoria con "sdcard:", dentro de esta área de configuración definiremos el pin de "Chip Select" del SPI, el pin de detección de tarjeta SD presente si lo hubiera y la frecuencia de trabajo de la tarjeta de memoria.

Configuración de motores

Para configurar los motores utilizados primero deberemos definir el tipo de control utilizado para el manejo de los motores PAP. Bajo la etiqueta “stepping:” podemos definir el motor que usa fluid NC para el control, en el caso de nuestros drivers la mejor elección es RMT.

RMT es un periférico interno del que dispone el MCU habitualmente utilizado como transceptor infrarrojo, que fluid NC utiliza para generar una serie de pulsos sincronizados en las salidas de control de los motores. Los ajustes de este apartado se setearon a los valores por defecto indicados en la página de fluid NC.

Para configurar la salida PWM para el control del motor del oscilador, el firmware nos da la posibilidad de declarar salidas personalizadas, esto es, declarar salidas PWM que luego pueden ser controladas por comandos GRBL, así definimos una salida en el pin que establecimos para el motor DC. También definimos la frecuencia de trabajo del generador PWM.

```
stepping:
  engine: RMT
  idle_ms: 250
  pulse_us: 4
  dir_delay_us: 0
  disable_delay_us: 0
  segments: 12
```

Figura 21. Configuración de tipo de control de motores

```
user_outputs:
  analog0_pin: gpio.13
  analog0_hz: 120
```

Figura 22. Configuración de driver pwm

```
axes:
  x:
    steps_per_mm: 800.000
    max_rate_mm_per_min: 1000.000
    acceleration_mm_per_sec2: 25.000
    max_travel_mm: 400.000
    soft_limits: false
  motor0:
    stepstick:
      step_pin: gpio.22
      direction_pin: gpio.21
  homing:
    cycle: -1
```

Figura 23. Configuración de ejes de motores PAP

Finalmente configuramos cada eje, en la Figura 23 se muestra la configuración para el eje X, donde vemos que se puede configurar la cantidad de pasos por milímetro, la aceleración máxima en $[\text{mm}/\text{s}^2]$, la longitud máxima del eje en $[\text{mm}]$, y luego el tipo de driver (stepstick en nuestro caso) y los pines establecidos para el control del driver del motor. Los tipos de driver admitidos pueden verse en la página web de Fluid NC.

Configuración de interfaz UART

Finalmente configuramos la interfaz UART que usa el sistema para comunicarse con la interfaz física.

Definimos los pines de tx, rx, “baud rate” y modo del uart2, cabe aclarar que el uart1 es el que utiliza el MCU para comunicarse con el conversor Serie/USB. También definimos un nuevo canal de comunicación en el UART número 2.

```
uart2:
  txd_pin: gpio.17
  rxd_pin: gpio.16
  baud: 38400
  mode: 8N1
```

```
uart_channel1:
  uart_num: 2
```

Figura 24. Configuración de interfaz UART

Los canales de comunicación que utiliza el sistema de Fluid NC para recibir comandos y enviar mensajes pueden ser variados, por ejemplo: telnet, websockets, usb.

2.4.2 Firmware de Interfaz UART

Para desarrollar la interfaz en primera instancia observamos el hardware con el que vamos a trabajar, así tenemos, una pantalla full-graphics conectada a través de SPI, un encoder rotativo, un buzzer, y un botón de stop, además de un puerto UART para la comunicación con el microcontrolador.

En esta instancia se decidió utilizar sólo la pantalla para simplificar la operación, es decir que los elementos de entrada (encoder y botón) y el buzzer serán incorporados a futuro.

Para la conexión por UART, como el microcontrolador posee un solo puerto UART físico, y el mismo lo utilizamos para debuggear el firmware, se decidió generar un UART virtual, a través del uso de la librería “SoftwareSerial”.

Una vez establecidos los métodos de lectura del puerto y escritura en pantalla, procedemos a analizar los mensajes que nos puede enviar el firmware Fluid NC a través de este puerto para decidir de qué manera debemos interpretarlos y mostrarlos en la pantalla.

Así investigando este firmware, y analizando la lista de comandos que pueden enviarse, nos encontramos con el comando “?”, este comando genera un string con la información posicional actual de la máquina y el estado de esta.

Un mensaje de información típico de la máquina es el siguiente:

```
<Run|MPos:0.000,0.000,0.000,0.000|FS:1000,0>
```

Vemos que es una cadena de caracteres con una estructura fija delimitada por los caracteres “<” y “>”, al inicio figura un mensaje que refleja el estado de la máquina (Por ejemplo: Run, Idle o Alarm) y luego lo sigue separado por el carácter “|” cada parámetro posicional luego del texto “MPos:” separados por comas, hacia el final lo que figura como “FS:” resulta ser la velocidad de movimiento actual de la máquina.

Por lo tanto para obtener los parámetros sólo debemos tomar el string recibido y procesarlo teniendo en cuenta su estructura, para guardar los datos que nos interesan del mismo que deseamos mostrar en la pantalla como: posición en X, Y, X y A, y el estado de la máquina.

Esto lo realizaremos diseñando una clase llamada “Procesador_Mensaje” con la siguiente estructura:

```

1  #ifndef PROCESADOR_MENSAJE_H
2  #define PROCESADOR_MENSAJE_H
3
4  #include <Arduino.h>
5
6  class Procesador_Mensaje{
7      //Parámetros de la máquina:
8      private:
9          float x,y,z;
10         String estado;
11
12     public:
13         Procesador_Mensaje();
14         int procesar(String linea);
15
16         float get_X();
17         float get_Y();
18         float get_Z();
19         String get_Estado();
20     };
21
22 #endif
23

```

Figura 25. Definición de la clase "Procesador_mensaje"

La clase como se ve en la imagen posee 4 variables tipo *float* privadas, que almacenan los valores leídos de cada eje (X, Y, Z y A) y una variable de tipo *string* que guarda el estado de la máquina.

Posee 7 métodos, de los cuales uno es el constructor de la clase, 5 son getters con los que recuperamos el valor de los parámetros privados y un método llamado "procesar" que toma como parámetro el *string* que será procesado.

Este método es el encargado de hacer el proceso de observar el *string* completo que, a partir de los caracteres de delimitación y teniendo en cuenta la estructura antes mencionada, convierte los valores indicados por caracteres en *floats* que son guardados en los parámetros de la clase.

2.5 Desarrollo del software

Para generar los diseños se decidió utilizar el software Inkscape [9], el cuál es un programa de diseño por vectores, con la ventaja de ser libre y Open-Source.

Este software incorpora un plugin denominado "gcodetools" [10] que toma un modelo bidimensional y con unas pocas configuraciones entrega un archivo de tipo *.gcode apto para distintos tipos de máquinas, especialmente grabadores láser.

La forma en que presenta la información este plugin es incompatible con el firmware de la máquina que utiliza el sistema GRBL, por lo tanto se escribió un software de post-procesado, que admite un archivo de texto con extensión *.gcode y entrega el mismo archivo adaptado al sistema GRBL de la máquina.

Para ello se estudió la estructura de algunos archivos generados por gcodetools contrastando cada línea con el glosario de comandos indicado por el sistema GRBL, y así se definió de qué manera deberían ser presentados los comandos para que el firmware de la máquina pueda interpretarlos correctamente, de este modo, utilizando funciones de manipulación de la librería String, este programa toma línea por línea del archivo de origen, extrayendo los valores de los parámetros de movimiento de cada eje y luego los escribe en el archivo de salida en el formato correcto, esta forma de manipular la información permite también hacer operaciones de escala o conversión sobre los parámetros, que pueden ser útiles en algunas aplicaciones.

En resumen, el flujo de trabajo para generar un diseño de corte para la máquina se puede sintetizar en tres pasos:

- 1) Se diseña el modelo de corte en el software Inkscape, obteniendo un archivo *.SVG.

- 2) Se realiza la transformación del diseño en instrucciones para la máquina utilizando el plugin Gcodetools, obteniendo el archivo *.GCODE.
- 3) Se realiza el post-procesado del archivo con el programa desarrollado en Processing [11] y se obtiene finalmente un archivo *.GCODE adaptado al firmware de la máquina que puede ser transferido a través de la interfaz web de esta a su tarjeta de memoria.

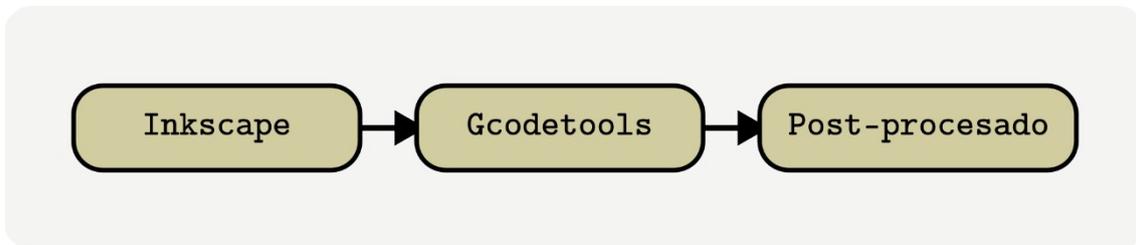


Figura 26. Cadena de software para generar un archivo de corte

Capítulo 3: Resultados

3.1 Resultados generales por sistema

En este capítulo se observan los resultados obtenidos separando cada sistema que compone la máquina y estudiándolos por separado, así se desarrolla el análisis del sistema mecánico, electrónico, del firmware/software y finalmente se estudia el desempeño general de la máquina en el corte de distintos materiales.

Resultados del sistema mecánico:

El formato elegido para la transferencia de movimiento, en el que el carro posee el motor dentro de él facilita la modificación de la longitud de los ejes, pensando en la personalización de la máquina, utilizando los mismos modelos de piezas pueden crearse máquinas con dimensiones diferentes, sólo eligiendo ejes y correas apropiados.

En relación con el eje X, se observó que posee baja resistencia a la flexión, debido a que el eje resultó ser muy largo para las varillas de 8 [mm] de diámetro que no están soportadas como en el caso del eje Y, en su uso normal, el eje se flexionaba cuando se probó por primera vez, por esto se añadió un eje auxiliar de mayor sección, mejorando esta característica.

Los motores utilizados en la máquina mostraron un buen desempeño y ejercen la fuerza necesaria para mover los ejes sin perder pasos en las pruebas que se realizaron, y con gran repetibilidad incluso al no contar con un sistema de retroalimentación.

Las piezas estructurales impresas en 3D han soportado las fuerzas ejercidas durante el movimiento y corte de la máquina sin deformarse ni romperse, de todas maneras, si se piensa en generar un producto industrial se deberían fabricar estas piezas de materiales más resistentes.

En relación con la herramienta de corte, el desempeño ha sido bueno para materiales finos, como el cartón de 1 y 3 [mm], sin embargo, con el material más grueso, el cartón corrugado doble de 6 [mm] de espesor, ha mostrado cierta dificultad para cortarlo y se vio que el motor oscilador no dispone de la fuerza o la velocidad suficiente para mantener la cuchilla oscilando cuando se intentó cortar de una sola pasada el material grueso. Para solucionar este problema, se optó por modificar el código de corte, y realizar múltiples pasadas con menor profundidad de corte y una menor velocidad, gracias a esto se logra cortar sin problemas el cartón doble, con la desventaja de que se necesita más tiempo de corte.

Se notó que algunas piezas de la herramienta de corte sufren un gran desgaste debido al rozamiento, es el caso de la pieza central del bloque oscilador, y la pieza de transferencia de movimiento giratorio del rotador; estas piezas deberían ser fabricadas en un material con menos desgaste y posiblemente mecanizadas en vez de impresas en 3D, con esto lograríamos un menor costo de mantenimiento de la máquina.

Por otro lado, se observó que la cuchilla elegida se desafilaba muy rápidamente, por lo que podrían probarse otro tipo de cuchillas de corte que no se desgasten tan fácilmente, y den mayor cantidad de tiempo de corte antes de reponerlas.

Resultados del sistema electrónico:

El circuito final si bien es bastante sencillo, cumple con los objetivos propuestos ampliamente; aún así pensando en una producción industrial de la máquina sería necesario fabricar un PCB donde se incorpore directamente el módulo ESP32 sin la placa de desarrollo, diseñando un sistema de alimentación más adecuado. De la misma manera se pueden incorporar directamente los componentes del módulo utilizado para la tarjeta microSD, integrando todo en un solo PCB.

La conectividad inalámbrica del microcontrolador da la posibilidad que se puedan incorporar interfaces de control y monitoreo inalámbricas, en un futuro, ampliando la facilidad y practicidad de uso de esta.

La separación de la interfaz física es interesante, ya que da la posibilidad de comercializar la máquina optando por una versión sin interfaz; además, esta característica modular, permite la mejora continua de la interfaz externa o la creación de diferentes interfaces por separado de la máquina de corte, permitiendo incluso su utilización en otras máquinas de fabricación automática.

Resultados del sistema de software y firmware:

El firmware FluidNC es muy fácil de configurar, y está muy bien documentado, lo que facilita el proceso de adaptación a proyectos particulares y máquinas con diversas morfologías.

La interfaz web que provee funciona muy bien y es muy práctica para pasar los diseños realizados en una PC que no necesita estar en el mismo espacio físico que la máquina, lo cual es muy común en algunas empresas donde el equipo de diseño se encuentra fuera del taller de fabricación; incluso esto da la posibilidad de bajar archivos a la máquina remotamente a través de la web, dando más flexibilidad al equipo de diseño.

Por otro lado, la interfaz física es ideal para ajustar la posición de cabezal previo a realizar el corte, pudiendo aprovechar más el material y facilitando el uso de la máquina al estar haciendo varios cortes en serie. Además, permite ver fácilmente el estado de la máquina e incluso mostrar errores cuando se presentan.

En relación con el software, la utilización de tres herramientas diferentes de software hace un poco compleja la operación de trasladar los diseños realizados a la máquina de corte, además que la configuración en el plugin *gcodetools*, no es muy práctica y puede ser difícil de realizar para el usuario. Por esto sería necesario integrar los tres softwares en una sola herramienta, o bien desarrollar un software que permita tomar los archivos de diseño en crudo, y realizar el proceso de análisis y generación del archivo *gcode* configurando apropiadamente el mismo, con una interfaz visual que permita previsualizar el archivo y obtener ciertos parámetros, como por ejemplo, el

tiempo de corte; es de esta manera que trabajan los software de impresoras 3D para “cortar” los modelos y obtener el archivo que será leído por la máquina.

3.2 Desempeño en el corte de distintos grosores de cartón

En las siguientes imágenes puede verse el corte de distintas figuras geométricas en distintos grosores de cartón. Los cortes tienen una gran precisión y un muy buen acabado de los bordes.

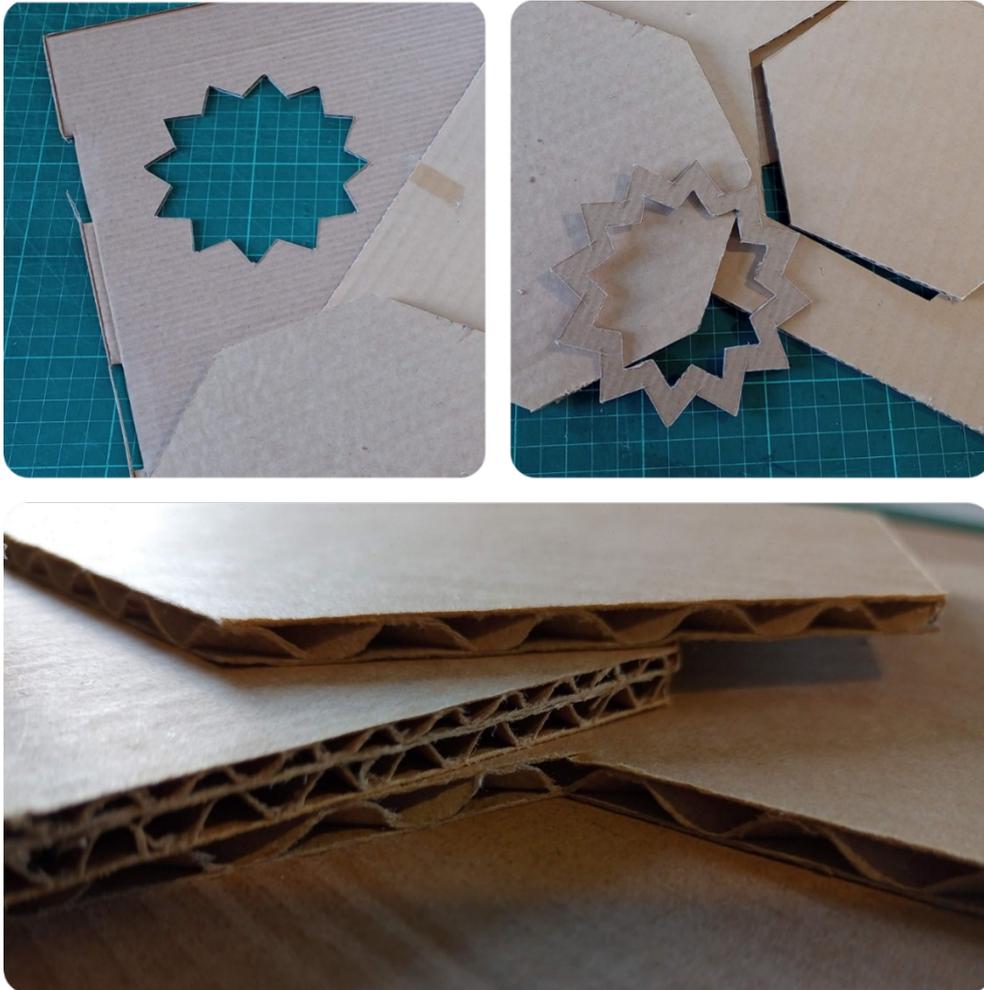


Figura 27. Pruebas de corte

Si comparamos con una máquina de corte láser, vemos que una desventaja del láser es la terminación del borde, que aparece quemado de color negro y a veces puede dañar también la superficie, en contraste con el método de corte utilizado en este proyecto con el que se obtiene un corte limpio.



Figura 28. Detalle de bordes cortados con la máquina (izquierda) y con corte láser (derecha)

En algunos casos, particularmente en los bordes angulosos y cuando se cortan huecos, se observa que la cuchilla se pasa levemente del borde y deja una marca, esto se da por la morfología de la cuchilla que realiza un corte más ancho de acuerdo con la profundidad a la que penetra, lo que puede solucionarse ajustando los parámetros de corte, haciendo que el avance de la cuchilla sea compensado en función de la profundidad para que no llegue a cortar el borde.



Figura 29. Detalle de borde cortado por la cuchilla

3.3 Características finales del producto

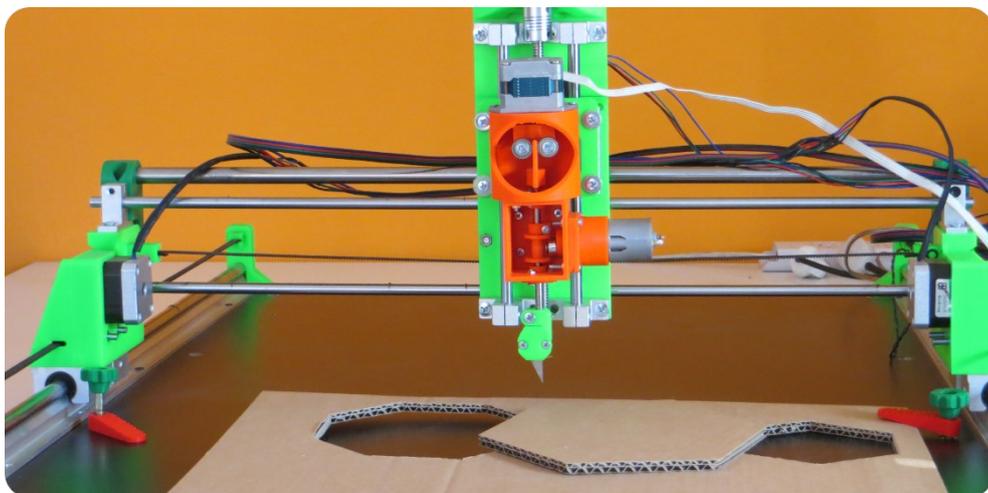


Figura 30. Fotografía de la máquina de corte finalizada

- Dimensiones: 90 x 60 x 35 [cm].
- Peso: 16 [kg]

- Materiales compatibles: cartón de hasta 6 [mm] (corrugado), papeles, cartulinas, papel Kraft de hasta 460 [gr].
- Consumo: máximo de 240 [W].
- Precio de venta: \$232.000

Capítulo 4: Análisis de Costos

En este capítulo analizaremos los costos que se dieron durante la ejecución de este proyecto y las vías de financiación de cada parte.

Dado que el desarrollo se dio en un lapso grande y teniendo en cuenta la inflación que sufre año a año la moneda nacional, se tomaron en cuenta los datos históricos del IPC (Índice de precios) detallados en la página web del INDEC (Instituto Nacional de Estadística y Censos); con estos datos se calculó la inflación acumulada para actualizar los costos de cada componente de la máquina según su fecha de compra (febrero de 2022) a la fecha de escritura de este proyecto.

En la siguiente tabla se desglosan los datos históricos desde el año 2022 y posteriormente el índice de inflación acumulada obtenido.

Porcentaje de inflación mensual			
Mes	2022	2023	2024
Enero	3,9	6	20,6
Febrero	4,7	6,6	-
Marzo	6,7	7,7	-
Abril	6,1	8,4	-
Mayo	5,1	7,8	-
Junio	5,3	6	-
Julio	7,4	6,3	-
Agosto	7	12,4	-
Septiembre	6,2	12,8	-
Octubre	6,4	8,3	-
Noviembre	4,9	12,8	-
Diciembre	5,1	25,5	-

Tabla 8. Inflación porcentual por mes.

Índice de Inflación acumulada desde 02/2022			
Mes	2022	2023	2024
Enero	-	1,9906543	7,05439749
Febrero	1,047	2,12203748	
Marzo	1,117149	2,28543437	
Abril	1,18529509	2,47741085	
Mayo	1,24574514	2,6706489	
Junio	1,31176963	2,83088783	
Julio	1,40884058	3,00923377	
Agosto	1,50745942	3,38237875	
Septiembre	1,60092191	3,81532323	
Octubre	1,70338091	4,13199506	
Noviembre	1,78684658	4,66089043	
Diciembre	1,87797575	5,84941749	

Tabla 9. Índice de inflación acumulada desde febrero de 2022.

El índice de inflación acumulada obtenido desde febrero de 2022 hasta febrero de 2024 es de **7.05**.

4.1 Costos de materiales y mano de obra

Costos de materiales:

De acuerdo con las tablas de materiales antes observadas en el desarrollo del proyecto, ahora ampliamos estas con el costo de cada componente actualizado según el índice de inflación calculado, así obtenemos las tablas de componentes electrónicos y componentes mecánicos, de la misma forma se toman las tablas de piezas impresas en 3D y se calcula su costo.

Lista de componentes estructurales, mecánicos y electromecánicos			
Descripción	Cantidad	Costo Unitario	Total
Base de madera melamina 90 [cm] x 60 [cm] x 18 [mm]	1	\$ 42.617,25	\$ 42.617,25
Barra Soportada Templada p/ sbr12 x 900 [mm]	2	\$ 49.614,66	\$ 99.229,31
Soporte SK8	8	\$ 2.140,10	\$ 17.120,78
Barra Templada y Cromada Ø8 x 600 [mm]	2	\$ 7.941,12	\$ 15.882,24
Barra Templada y Cromada Ø8 x 200 [mm]	2	\$ 3.164,04	\$ 6.328,08
Rodamiento SCS8LUU	4	\$ 7.532,71	\$ 30.130,85
Rodamiento SBR12UU	4	\$ 6.659,29	\$ 26.637,16
Tuerca THSL Ø8 [mm] Paso 2 [mm] Antibacklash	1	\$ 5.694,29	\$ 5.694,29
Varilla THSL Ø8 [mm] Paso 2 [mm] x 200 [mm]	1	\$ 8.000,62	\$ 8.000,62
Motor NEMA 17 0.9°	5	\$ 13.387,95	\$ 66.939,75
Motor DC RS-795	1	\$ 24.322,50	\$ 24.322,50
Correa GT2	3	\$ 3.990,00	\$ 11.970,00
Rodamiento 608	1	\$ 1.500,00	\$ 1.500,00
Rodamiento 625	3	\$ 1.540,00	\$ 4.620,00
Acople Flexible 5x8 mm	1	\$ 4.860,00	\$ 4.860,00
Polea GT2 Lisa 10mm x 5mm	6	\$ 3.500,00	\$ 21.000,00
Tuerca manguito para madera M6 x10u.	1	\$ 3.500,00	\$ 3.500,00
Repuesto para cutter bisturí x10u.	1	\$ 4.800,00	\$ 4.800,00
TOTAL			\$ 395.152,84

Tabla 10. Costos de componentes mecánicos

Lista de componentes electrónicos del prototipo			
Descripción	Cantidad	Costo Unitario	Total
Arduino MEGA	1	\$ 16.771,95	\$ 16.771,95
Drivers DRV8825	5	\$ 1.959,90	\$ 9.799,50
Módulo RAMPS 1.6	1	\$ 7.021,80	\$ 7.021,80
Fuente 12 [V] 20 [A]	1	\$ 34.899,12	\$ 34.899,12
Cable 4 polos x 24 AWG x10m.	1	\$ 6.220,00	\$ 6.220,00
TOTAL			\$ 74.712,37

Tabla 11. Costos de componentes electrónicos del prototipo

Lista de componentes electrónicos del desarrollo final

Descripción	Cantidad	Costo Unitario	Total
Módulo Nodemcu ESP32	1	\$ 10.885,00	\$ 10.885,00
Arduino Pro-Micro	1	\$ 16.000,00	\$ 16.000,00
Módulo Step-Down 2[A] "Mini-360"	1	\$ 2.450,00	\$ 2.450,00
Módulo microSD	1	\$ 3.600,00	\$ 3.600,00
Módulo Display Full Graphics	1	\$ 28.759,00	\$ 28.759,00
Placa perforada 9x15 [cm]	1	\$ 2.451,00	\$ 2.451,00
Placa perforada 7x9 [cm]	1	\$ 1.508,00	\$ 1.508,00
Capacitor Electrolítico 47 [uF] 16[V]	6	\$ 98,00	\$ 588,00
IRL520N	1	\$ 975,00	\$ 975,00
Resistencia 10k 1/4 [W]	1	\$ 50,00	\$ 50,00
Resistencia 1k 1/4 [W]	1	\$ 50,00	\$ 50,00
Resistencia 2,2k 1/4 [W]	1	\$ 50,00	\$ 50,00
Cable alimentación 220[V] 10[A]	1	\$ 2.500,00	\$ 2.500,00
Conector JST Hembra 4 pin	5	\$ 480,00	\$ 2.400,00
Conector JST Hembra 3 pin	4	\$ 380,00	\$ 1.520,00
Conector JST Macho 4 pin	5	\$ 480,00	\$ 2.400,00
Conector JST Macho 3 pin	4	\$ 380,00	\$ 1.520,00
TOTAL			\$ 77.706,00

Tabla 12. Costos de componentes electrónicos del desarrollo final

Piezas impresas en 3D:

A estos costos debemos añadir los costos de fabricación de las piezas impresas en 3D, para ello en función del tiempo de impresión y la cantidad de material se evalúa el costo individual que luego se suma al final.

El costo de material se evalúa a partir del costo actual en el mercado del material utilizado, y en función del peso calculamos su valor. Actualmente (febrero de 2024) el valor por [Kg] de filamento PETG oscila entre \$16700 y \$20000, tomamos un valor medio como referencia de \$18350.

La función para calcular el costo por peso (C_p) se establece como:

$$C_p = \text{Peso de pieza} \frac{[\text{gr}] * \$ 18350}{1000}$$

Ahora observando el parámetro del tiempo de impresión, debemos tener en cuenta para estos dos factores el consumo energético de la máquina y el desgaste. Para esto vamos a tomar de referencia una impresora 3D de gama media no industrial, como por ejemplo el modelo Ender-5 Plus de la marca Creality, esta máquina tiene un costo actual de \$1000000, y su consumo energético máximo es de 550[W].

Según la página web de la empresa de energía local (EPE), el costo actual por kWh oscila entre \$14.11 y \$24.48, obteniendo un valor medio de \$19.3.

Para calcular el desgaste de la máquina tomamos una vida útil estimada de 10000 horas, teniendo en cuenta su costo de compra, obtenemos un valor de amortización de \$100 por hora.

Así la función para calcular el costo por tiempo de trabajo de la impresora es:

$$Ch = \text{Tiempo de impresión [min]} * (1 [h] / 60 [min]) * \left(\$ 100 + \$ 19.3 * \left(\frac{550}{1000} \right) \right)$$

Volviendo a la Tabla 2 y 3, podemos ampliar estas obteniendo el costo de impresión de cada pieza:

Resumen de costo de piezas impresas en 3D				
Nombre	t [min]	p [gr]	Cantidad	Costo
Carro Eje Y	254	71	2	\$ 3.542,24
Soporte de tensores Eje Y	63	20	4	\$ 1.932,58
Separadores Eje Y	4	1	2	\$ 51,45
Tensores móviles de correas	12	3	6	\$ 463,04
Soporte Eje Auxiliar del Eje Y	54	17	2	\$ 823,01
Pieza Central del carro del Eje X	177	55	1	\$ 1.335,56
Soporte del motor del Eje X	51	14	1	\$ 350,92
Separador de poleas del Eje X	2	1	2	\$ 44,07
Soporte Eje Auxiliar del Eje X	74	23	1	\$ 558,48
Soporte del motor del eje Z	171	55	1	\$ 1.324,50
Acople del soporte del motor del eje Z al Eje X	16	5	1	\$ 121,25
Carro Eje Z	79	23	1	\$ 567,69
Soporte inferior varilla roscada del Eje Z	16	5	1	\$ 121,25
Pieza de acople de herramienta de corte al Eje Z	55	16	1	\$ 395,00
Pata de sujeción de material	13	4	8	\$ 778,93
Perilla de sujeción de material	9	3	8	\$ 573,14
Cuerpo del sistema rotador	125	37	1	\$ 909,40
Pieza central del rotador	16	4	1	\$ 102,90
Pieza móvil del rotador	12	2	1	\$ 58,82
Cuerpo central de la herramienta	86	22	1	\$ 562,25
Buje	7	2	2	\$ 99,21
Oscilador	13	2	1	\$ 60,67
Soporte del Motor Oscilador	27	8	1	\$ 196,58
Separador del Motor Oscilador	9	2	1	\$ 53,29
Excéntrico del Motor Oscilador	8	2	1	\$ 51,45
Pieza de sostén de la cuchilla de corte	17	4	1	\$ 104,74
			TOTAL	\$ 15.182,41

Costos de Mano de obra:

El costo de mano de obra se puede desglosar en varios puntos indicando todas las actividades que fueron desarrolladas para la ejecución del proyecto y el tiempo destinado a cada una.

Para el cálculo de los honorarios por el trabajo realizado se toma como referencia una unidad arancelaria establecida por el CIEER [12] (Colegio de Ingenieros Especialistas de Entre Ríos) llamada "Ingenio" [13], el cual se ajusta a través de resoluciones que

emite el CIEER; de acuerdo con la última resolución [14] (febrero 2024), se fijó el valor de la unidad en **\$5800**.

El valor de esta unidad se corresponde con los honorarios mínimos que pueden establecerse por ingenieros especialistas, y para cada tipo de trabajo o para determinados proyectos y de acuerdo con la antigüedad se establecen múltiplos de esta unidad para valorar los trabajos específicos. En este caso para este proyecto tomaremos el valor de la unidad para establecer el valor de la hora de trabajo en \$5800, y así calcular los costos totales por la mano de obra debidos al presente proyecto.

Costos mano de obra		
Detalle actividad	Tiempo [h]	Valor total
Diseño 3D integral de la estructura de la máquina y de piezas impresas en 3D	120	\$ 696.000,00
Diseño 3D de la herramienta de corte y piezas constitutivas	30	\$ 174.000,00
Diseño del sistema electrónico	45	\$ 261.000,00
Armado de la máquina	15	\$ 87.000,00
Prototipado del circuito electrónico, armado y conexionado de cables	25	\$ 145.000,00
Configuración y ajuste del firmware FluidNC	12	\$ 69.600,00
Programación del software de post-procesado	30	\$ 174.000,00
Gestión de compra de componentes y materiales y de recursos de financiación	12	\$ 69.600,00
Pruebas y estudio del desempeño de la máquina	15	\$ 87.000,00
TOTAL		\$ 1.763.200,00

Tabla 14. Costos de mano de obra

Resumen:

En resumen, obtenemos los siguientes datos sobre los costos totales del proyecto:

Resumen de Costos Totales	
COSTO TOTAL Materiales mecánicos	\$ 395.152,84
COSTO TOTAL Materiales electrónicos	\$ 152.418,37
COSTO TOTAL Piezas impresas en 3D	\$ 15.182,41
COSTO TOTAL Mano de Obra	\$ 1.763.200,00
TOTAL	\$ 2.325.953,62

Tabla 15. Resumen de costos totales del proyecto.

Cálculo del costo de fabricación por unidad y precio del producto:

Anteriormente calculamos los costos totales del proyecto, ahora si tomamos sólo lo referente a la fabricación de la máquina deducimos los costos totales de fabricación del producto de la siguiente manera:

Costos totales por unidad	
MATERIALES	
COSTO TOTAL Materiales mecánicos	\$ 395.152,84

COSTO TOTAL Materiales electrónicos finales	\$ 118.825,12
COSTO TOTAL Piezas impresas en 3D	\$ 15.182,41
MANO DE OBRA	
COSTO Armado de máquina	\$ 87.000,00
TOTAL	\$ 616.160,37

Tabla 16. Costos totales por unidad

El costo total de la máquina por unidad es de **\$616.160,37**.

Para definir el precio de venta del producto observamos los precios actuales de productos competitivos y fijamos el precio para que sea levemente menor que productos con similares características y luego calculamos el margen de ganancia.

Comparación con otros productos similares:

Para comparar nuestro producto, elegimos máquinas con características que permitan fabricar los mismos productos que se pueden crear con la máquina, y así encontramos en el mercado: Máquinas de corte láser, Routers CNC industriales y máquinas de menor porte de grabado y corte por láser de diodo.

Confeccionamos una lista indicando los precios de cada producto, ordenándolos de menor a mayor precio:

Productos con características similares	
Detalle	Precio de venta
Ortur S2 10W, diodo láser, de bajo porte	\$ 1.644.900,00
Atomstack A10 Pro 50w, diodo láser, de bajo porte	\$ 1.650.900,00
Odyssey 30x40[cm], láser CO2 de 50W, industrial	\$ 3.780.000,00
Máquina de corte láser K6040, láser de CO2 de 60W, industrial	\$ 4.708.000,00
Router CNC industrial Starcam 913 de 1600W	\$ 4.803.210,00

Tabla 17. Productos con características similares.

Observando la tabla, podemos fijar una ganancia del 100% y ubicar el producto con un precio competitivo de: **\$1.232.000**

4.2 Gestión de financiamiento Proyecto “Encartonate”

Para financiar el proyecto se buscaron líneas de subsidio en el ámbito local, de esta manera, se aplicó a una convocatoria de la provincia de Santa Fe llamada “Industrias Creativas: Concurso Anual”, con un proyecto denominado “Encartonate” cuyo objetivo principal fue financiar la construcción de la máquina con el objetivo de sumarle valor agregado a los productos generados por el emprendimiento Kanchay a través de la incorporación de procesos de fabricación digital y la producción de diseños más complejos y atractivos.

El proyecto “Encartonate” fue beneficiario de dicho subsidio, se ejecutó entre los meses de enero y julio de 2022, con un monto total de \$200.000, este monto fue distribuido en cuatro categorías: Máquina (\$90.000), Diseño (\$60.000), Insumos (\$20.000) y Publicidad (\$30.000).

Con el monto indicado para la categoría Máquina se logró financiar la primera etapa de desarrollo y fabricación del prototipo de la máquina. Los montos indicados para “Diseño”, “Insumos” y “Publicidad” se destinaron de acuerdo con la planificación del proyecto para una investigación y desarrollo de diseños paramétricos, para la compra de insumos para el emprendimiento y para el diseño de un catálogo para la marca respectivamente.

Dentro de los productos nuevos que se logró producir con el proyecto fueron calados para tapas de cuadernos con distintas formas geométricas, soportes y luminarias con diseños novedosos y más precisos; una serie de fotografías de estos se puede ver a continuación.



Figura 31. Productos realizados por el emprendimiento Kanchay para el proyecto "Encartonate"

4.3 Amortización de la inversión

Si bien no es la intención del proyecto comercializar la máquina, podemos diseñar un plan de comercialización para amortizar la inversión de este y poder construir y vender unidades del producto, para ello debemos en primer lugar cubrir los gastos iniciales del proyecto y el costo neto de fabricación por unidad.

Si observamos los costos totales según la Tabla 15, vemos que la suma es de: \$ 2.325.953,62, a este monto debemos descontarle el monto establecido por el proyecto “Encartonate” con el que se financió parte de estos costos iniciales, el monto fue de \$90.000, que, actualizado según el índice de inflación antes hallado, nos da un total de: \$634.500. De esta manera el monto restante es: **\$1.691.453,62**.

Con el precio fijado en la sección 4.2, vemos que la ganancia por producto vendido es de \$616.000, por lo que para recuperar la inversión deberemos fabricar y vender 3 unidades, obteniendo una ganancia neta de: \$156.546,38.

Estrategias de financiación alternativas:

Al inicio del proyecto se pensó en la financiación de este a través de la solicitud de subsidios a organismos públicos y privados, que muchas veces ofrecen este tipo de posibilidades, principalmente para proyectos vinculados a la ciencia y la tecnología, por lo tanto, para la mejora y avance de este pueden seguir gestionándose este tipo de beneficios.

Otra alternativa que se piensa para financiar el restante es publicar el proyecto creando una página web y abriendo el desarrollo de este a la comunidad, buscando donaciones para quienes deseen aportar al proyecto.

Ámbitos de comercialización:

Se puede comercializar la máquina en distintos ámbitos:

- a) Como una herramienta de trabajo para pequeñas pymes o emprendedores.
- b) Como un instrumento educativo para universidades y escuelas técnicas, que pueda ser fabricada por las mismas instituciones a través de un instructivo y luego modificada por los estudiantes tanto las características del firmware como del software para adaptarla a sus necesidades.
- c) Como una herramienta creativa para espacios donde se desarrollen actividades artísticas más cercanas al arte plástico.

Capítulo 5: Discusión y Conclusión.

5.1 Análisis de resultados obtenidos

Observando el desarrollo integral de este proyecto, podemos notar que en la fabricación de maquinarias y sistemas de automatización en general, existe una relación intrínseca entre las áreas de la mecánica, la electrónica y la informática, para lo que fue necesario mucha investigación y estudio, principalmente en lo relacionado a la mecánica, ya que la formación recibida en la facultad en temáticas relacionadas a electrónica y de programación me ha permitido abordar este desarrollo sin inconvenientes.

Evaluando el trabajo realizado y haciendo una autocrítica, en este tipo de proyectos en los que entran en juego tantas áreas diferentes del conocimiento, es de gran importancia contar con un equipo de trabajo que incorpore profesionales de las distintas áreas nombradas anteriormente que se dediquen específicamente a cada parte del proyecto, de esta manera se hubiera logrado alcanzar un mejor producto en mucho menor tiempo de el que requirió la conclusión de este proyecto.

Aún así se ha logrado cumplir ampliamente con los objetivos propuestos, logrando solucionar los problemas hallados en el proceso, creando una máquina que permite el corte de cartón y papel de distintos grosores de manera precisa, con una terminación comparable con aquella producida por las máquinas de corte láser; el producto es de bajo costo y porte, características que lo hacen ideal para su utilización en talleres con producciones pequeñas y con poco espacio disponible, abriendo otras posibilidades de creación y diseño de objetos en papel y cartón a partir de la incorporación del diseño digital en el proceso productivo, lo que permite la manufactura de una gran variedad de productos altamente personalizados, añadiéndoles un gran valor agregado.

De la misma manera todos los conocimientos adquiridos son muy valiosos para mi desarrollo profesional y complementan en gran manera la formación brindada por la universidad en todos estos años de mi carrera.

5.2 Trabajos futuros

Intercambio de cabezal: La máquina fue pensada de tal manera que puede incorporarse a futuro un sistema multi-cabezal que añada funciones extra como grabado y corte de placas de circuito, grabado y corte por abrasión de materiales como madera y plástico blando, mecanizado láser, plotter de dibujo sobre papeles, impresión 3D, etc.

Mejoras en el sistema mecánico: El sistema mecánico de la máquina puede robustecerse para que permita trabajar con materiales más duros y también se pueden modificar las dimensiones de esta sin modificar la mayoría de las piezas constitutivas, adaptándola a diversos escenarios y necesidades de los usuarios.

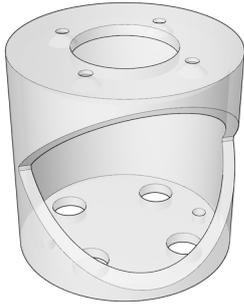
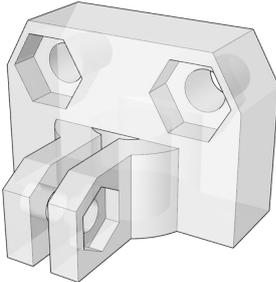
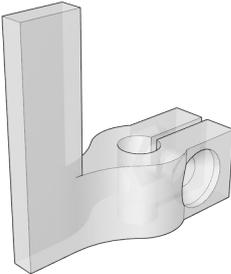
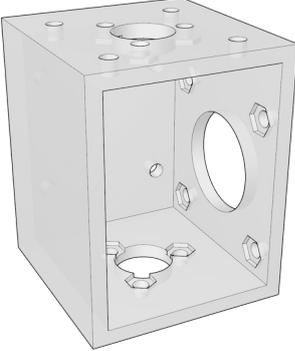
Desarrollo de la interfaz física: La interfaz física con que se puede observar el estado de la máquina y controlarla manualmente, se pensó de tal manera en que se puede desarrollar independientemente de la máquina de corte y conformarse en un proyecto en sí mismo, de tal manera que pueda utilizarse como una interfaz universal para distintas máquinas del mismo tipo que permitan una conexión por UART, añadiendo más funciones y configuraciones que mejoren su adaptabilidad.

Capítulo 6: Literatura Citada.

- [1] - Página de Instagram del emprendimiento “Kanchay”.
<https://www.instagram.com/kanchay.sf/>
- [2] - Web del software FreeCAD. <https://www.freecad.org>
- [3] - Referencia utilizada para el diseño de la herramienta de corte.
<https://www.thingiverse.com/thing:1436870>
- [4] - Web del software y firmware Repetier. <https://www.repetier.com>
- [5] - Esquemáticos y detalle del producto RAMPS 1.6. https://reprap.org/wiki/RAMPS_1.6
- [6] - Hoja de datos del módulo NodeMCU ESP32.
https://docs.ai-thinker.com/_media/esp32/docs/nodemcu-32s_product_specification.pdf
- [7] - Hoja de datos de ESP32-WROOM32.
https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.pdf
- [8] - Página web del proyecto FluidNC. <http://wiki.fluidnc.com>
- [9] - Página web del software Inkscape. <https://inkscape.org>
- [10] - Página de GitHub del plugin *gcodetools*. <https://github.com/cnc-club/gcodetools>
- [11] - Página web del software Processing. <https://processing.org>
- [12] - Página web del Colegio de Ingenieros Especialistas de Entre Ríos.
<https://cieer.org.ar/release/index.php/institucionalhor/legislacion/resoluciones>
- [13] - Ley que define la unidad arancelaria “Ingenio”.
<https://www.hcder.gov.ar/archivosDownload/textos/E21366-28102020-2.pdf>
- [14] - Resolución que define el valor del Ingenio para febrero de 2024.
<https://cieer.org.ar/release/images/documentos/resoluciones/1997%20NUEVO%20VALOR%20INGENIO.pdf>

Anexo

Anexo 1: Detalle de las piezas diseñadas para la herramienta de corte

Lista de piezas para impresión 3D de Herramienta de corte		
Nombre	Render	Cantidad
Cuerpo del sistema rotador		1
Pieza central del rotador		1
Pieza móvil del rotador		1
Cuerpo central de la herramienta		1

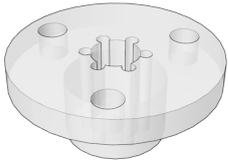
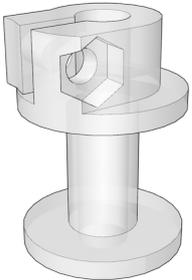
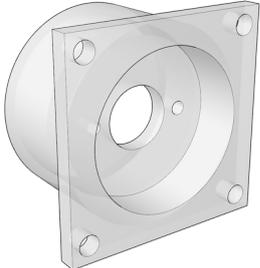
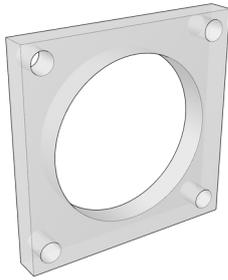
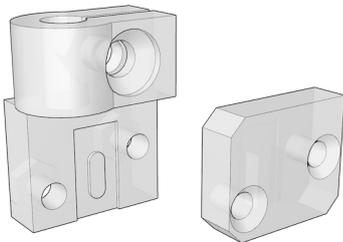
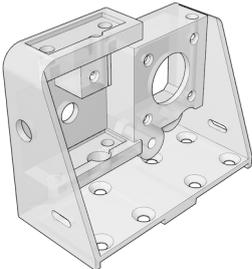
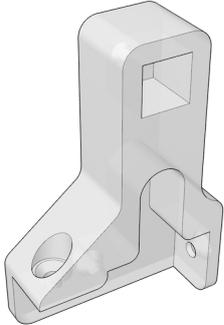
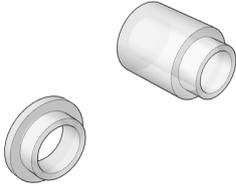
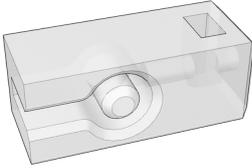
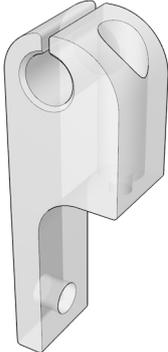
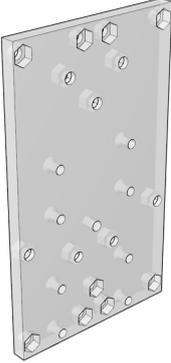
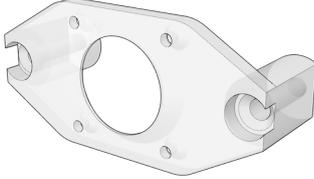
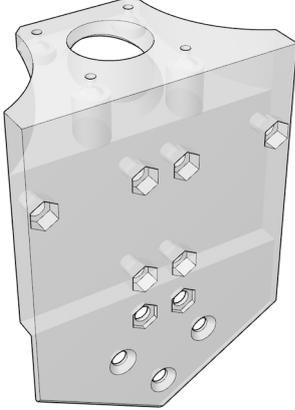
Buje		2
Oscilador		1
Soporte del Motor Oscilador		1
Separador del Motor Oscilador		1
Excéntrico del Motor Oscilador		1
Pieza de sostén de la cuchilla de corte		1

Tabla 18. Lista de piezas para impresión 3D de Herramienta de corte.

Anexo 2: Detalle de las piezas diseñadas para la estructura mecánica de la máquina

Lista de piezas para impresión 3D de la estructura de la máquina		
Nombre	Render	Cantidad
Carro Eje Y		2
Soporte de tensores Eje Y		4
Separadores Eje Y		2
Tensores móviles de correas		6
Soporte Eje Auxiliar del Eje Y		2

Pieza Central del carro del Eje X		1
Soporte del motor del Eje X		1
Separador de poleas del Eje X		2
Soporte Eje Auxiliar del Eje X		1
Soporte del motor del eje Z		1

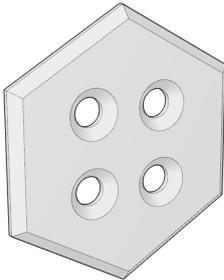
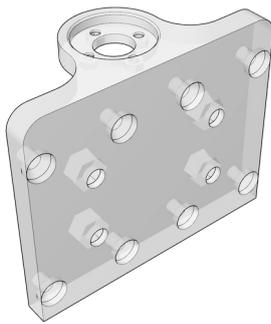
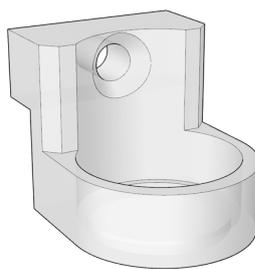
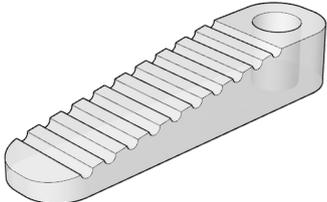
Acople del soporte del motor del eje Z al Eje X		1
Carro Eje Z		1
Soporte inferior varilla roscada del Eje Z		1
Pieza de acople de herramienta de corte al Eje Z		1
Pata de sujeción de material		8
Perilla de sujeción de material		8

Tabla 19. Lista de piezas para impresión 3D de la estructura de la máquina.