

# **Router CNC para placas PCB:**

## **Prácticas Profesionalizantes**

*Grupo 1.*



- **Docentes:**

- Osvaldo P. Ivani
- Sebastián Amago Prato
- Martín A. Ricciardelli
- Juan Martín Hermida

- **Estudiantes:**

- Dante G. Mele Lentile (Project Manager)
- León A. Martin
- Hernán A. Silva
- Nicolás L. Fertonani
- Laureano M. Rivera Pascua

- **Curso:** 4<sup>to</sup> Año Ciclo Superior - Electromecánica

- **Fecha de entrega:** 16/11/2020

# ÍNDICE

<b><u>Resumen</u></b>	<b><u>3</u></b>
<b><u>Introducción</u></b>	<b><u>4</u></b>
<b>Contextualización y Objetivos:</b>	4
<b>Placas de Circuito Impreso o PCB:</b>	5
<b>Exigencias y necesidades de la Empresa:</b>	6
<b><u>Diseño 3D, Mecánico y Estructural</u></b>	<b><u>7</u></b>
<b>Diseños conceptuales:</b>	7
<b>Pre-diseños y configuración:</b>	8
<b>Diseño en detalle:</b>	9
<b><u>Desvíos:</u></b>	9
<b><u>Ensamble Ajuste Fino Total + Electrónica:</u></b>	11
Subconjunto Inferior:	13
Subconjunto Puente:	19
Subconjunto Cabezal:	21
Subconjunto Carcasa:	27
<b><u>Cálculos Estructurales:</u></b>	33
<b>Diagramas de Operaciones y Planos:</b>	35
<b><u>Electrónica, Electricidad y Control</u></b>	<b><u>36</u></b>
<b>Primeras Investigaciones: Lógica y Control.</b>	36
<b>Potencia y Alimentación:</b>	37
<b><u>Alimentación:</u></b>	37
<b><u>Motor DC de Husillo:</u></b>	38
<b><u>Motores Paso a Paso:</u></b>	39
Drivers:	39
<b>Área Electrónica:</b>	40
<b><u>Canalización de Cables:</u></b>	41
<b><u>Placas de Control:</u></b>	42
Circuito Dimmer para Tira LED:	42
Circuito de Seguridad:	44
Circuito Inverter - Parada de Emergencia:	47
Regulador de Velocidad del Motor del Husillo:	48
<b>Botonera:</b>	49
<b><u>Jack y Porta LED:</u></b>	50
<b><u>Pantalla Controladora LCD Full Graphic Smart Controller:</u></b>	50
<b>Softwares y Firmware:</b>	51

<u>FlatCAM:</u>	51
Parámetros a definir para el agujereado de pads:	51
Parámetros a definir para el Enrutamiento de Pistas:	52
Parámetros a definir para el Corte de la Placa de Cobre:	53
Utilización del Software	53
<u>CNC GCode Controller:</u>	60
Utilización del Software:	64
<u>Firmware Marlin:</u>	68
Instrucciones:	68
Configuraciones para nuestra aplicación:	69
LCD (Full Graphic Smart Controller):	71
<u>Comandos G:</u>	73
<u>Comandos M:</u>	75
<b><u>Resumen de Especificaciones</u></b>	<b><u>76</u></b>
<b><u>Conclusiones</u></b>	<b><u>77</u></b>
<b><u>Oportunidades de Mejora</u></b>	<b><u>78</u></b>
<b><u>Bibliografia (Biblioteca Virtual)</u></b>	<b><u>79</u></b>

## **Resumen**

En el presente trabajo se realizó el diseño de una Fresadora o Router de Control Numérico Computarizado, capaz de fabricar diseños de placas de circuito impreso mediante mecanizado por arranque de viruta.

En este informe se plasmará la evolución de este proyecto, desde el inicio, donde se presentó una necesidad a resolver, hasta su conclusión.

Para el diseño tanto mecánico como electrónico y eléctrico, se siguió la metodología generada por Morris Asimow, dividiendo el mismo en una etapa introductoria de “Diseño conceptual”, en donde se plasma el problema a resolver, se investiga su viabilidad y se genera un concepto del producto a diseñar. Una etapa de “Configuración” o “Forma” en donde se define la configuración básica del producto, teniendo en cuenta todos los aspectos que lo condicionan. Una etapa de “Diseño en Detalle” en donde se pulen todos los contenidos generados en las etapas anteriores, se determinan las dimensiones del producto, sus materiales, etc. Y, por último, una etapa de “Planeación de cara a la manufactura” en donde se determinan las máquinas a la hora de generar el producto, se analizan los costos del mismo y se produce toda la documentación de respaldo con el objetivo de generar un correcto entregable.

# Introducción

## **1. Contextualización y Objetivos:**

En el marco de la materia de Prácticas Profesionalizantes en el Instituto Industrial Luis A. Huergo, el nueve de marzo del año en curso, se planteó el objetivo de generar un proyecto que solucione una determinada necesidad de una empresa. Con este objetivo en marcha, contactamos con la empresa Ingeniería Caamaño S.A. (I.C.S.A), enfocada a los servicios de ingeniería orientados a los sistemas de transportes verticales. Su necesidad, producir las placas de circuito impreso (PCB) necesarias para los displays de los ascensores. Nuestra solución, un equipo con movimiento en tres ejes, capaz de mecanizar por arranque de viruta las pistas de cobre, los agujeros y el corte de dichas placas.

Con esta solución, el tiempo de producción de las placas PCB disminuye notablemente, así como su costo. Al mismo tiempo, se vuelve innecesaria la utilización de uno de los métodos más comunes a la hora de realizar placas PCB, el de generar las pistas mediante el uso de químicos o “ácidos de grabado”. Evitando posibles efectos adversos potenciales para la salud y el medio ambiente.

Teniendo en cuenta los beneficios que otorga esta solución, nos enfocamos en diseñar un Router CNC para PCB's. El diseño de una máquina puede definirse como un proceso iterativo en donde se deben tomar decisiones sin la información completa, para luego volver al punto de inicio y realizar los ajustes necesarios.

Al día de la fecha hemos contactado con más de 70 empresas para que nos asesoren y/o coticen todos los elementos que componen nuestro diseño, hemos realizado 100 planos tanto mecánicos como eléctricos y electrónicos, 65 diagramas de operaciones, nuestro diseño en 3D posee más de 1000 elementos, incluyendo piezas estructurales fundamentales hasta toda la bulonería a utilizar. Dando como resultado más de 5000 horas de trabajo<sup>1</sup> durante la pandemia de COVID-19 del presente año.

Este documento se encontrará en una carpeta compartida de Google Drive, con el objetivo de generar una *carpeta de proyecto*. Dentro de la misma se podrán encontrar todos los documentos que componen nuestro proyecto.

---

<sup>1</sup>

Ver Anexo B: *Planilla de Horas*. Para observar el detalle anual de control de horas , dirigirse a la Hoja de Cálculo de Google “Planilla de Horas - Grupo 1”

## 2. Placas de Circuito Impreso o PCB:

Una placa de circuito impreso es una placa que tiene como objetivo permitir el montaje de distintos componentes electrónicos y, mediante el mismo, realizar conexiones entre ellos. Es decir, constituye tanto un soporte físico de los componentes como una situación de conexión de los mismos.

Está compuesta por un material de baja conductividad eléctrica y posee caminos de un material conductor. Usualmente se pueden encontrar en fibra de vidrio (puesto que otorgan ciertas ventajas y un costo no tan alto) pero también existen placas que poseen Pertinax (tienen un costo menor). Existen multitudes de configuraciones de cara a la estructura de la placa, pueden ser de una sola capa, de doble capa, multi-capa, flexibles, rígidas, de alta frecuencia y muchas variantes más.

Como se mencionó, existen distintos métodos para fabricar los caminos o buses de material conductor, entre ellos, el método químico. Una alternativa a este último es el fresado. Aquí lo que se hace es remover el cobre de las placas dejando únicamente los caminos conductores (sobre los cuales se montan los componentes). Esta última solución es la que nuestra máquina satisface.

### 3. Exigencias y necesidades de la Empresa:

A continuación se plasmarán las necesidades planteadas por parte la Empresa que es, al fin y al cabo, el usuario final de nuestro proyecto. Teniendo claras las necesidades y las exigencias del cliente pudimos enfocar nuestro diseño para cumplirlas de la forma más óptima y conveniente posible.

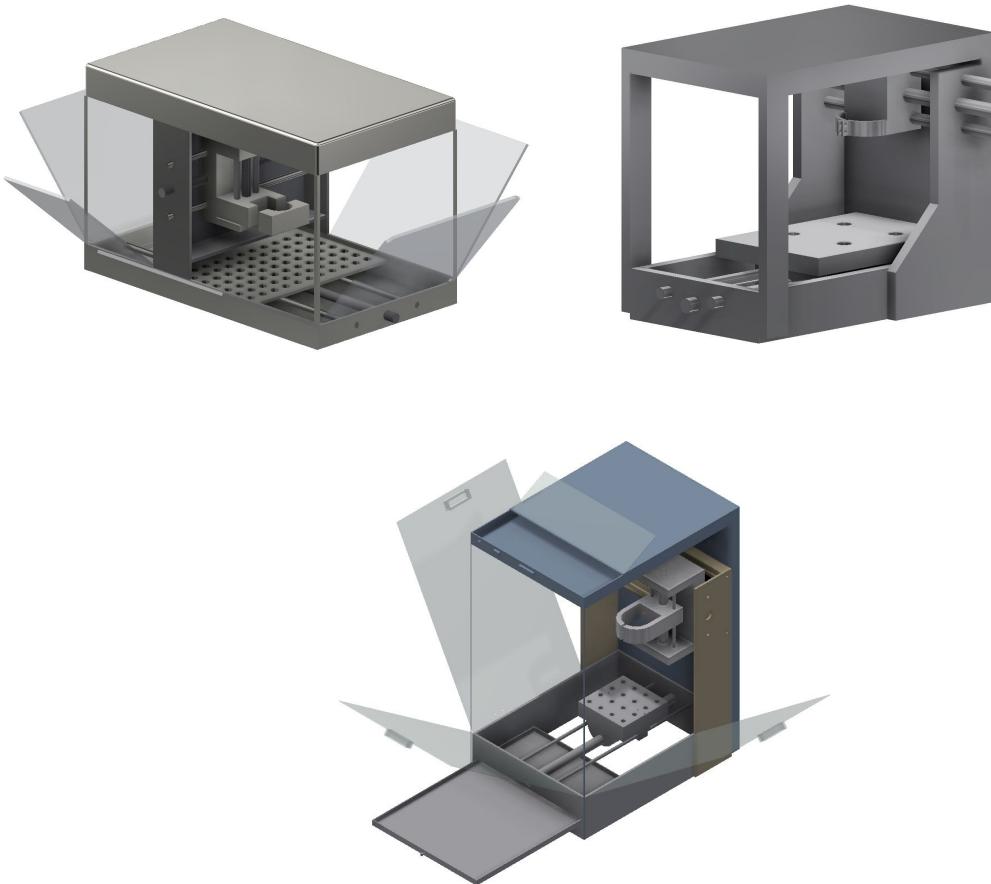
#### 3.1. Exigencias y necesidades a satisfacer:

- Fresar las pistas conductoras;
- Realizar los agujeros para los componentes electrónicos;
- Realizar un precorte de la placa;
- Resolución mínima de 1 mm;
- Posibilidad de utilizar una herramienta capaz de realizar los fresados y agujereados en las placas;
- Posibilidad de mecanizar, como mínimo, placas de 100 mm x 150 mm. Puesto que serán las de mayor producción;
- Área de corte de 200 mm x 200 mm x 50 mm;
- Movimiento independiente en los 3 ejes;
- Recorrido mínimo de eje X de 200 mm;
- Recorrido mínimo de eje Y de 200 mm;
- Recorrido mínimo de eje Z de 50 mm;
- Utilización de Arduino Mega 2560 y Ramps 1.4 para el control;
- Aislamiento de los componentes electrónicos del área de trabajo mediante una carcasa;
- Garantizar la seguridad del operario mediante un sistema de seguridad;
- Conexión mediante USB para control y envío de coordenadas;
- Capacidad de variar la velocidad del husillo;
- Seleccionar los programas a utilizar tanto para el control de la máquina como para el diseño de las PCB's a fabricar.

# Diseño 3D, Mecánico y Estructural

## 1. Diseños conceptuales:

En esta primera etapa, luego de conocer los objetivos de nuestro proyecto y las exigencias por parte de la empresa, nos enfocamos en realizar una investigación con el objetivo de conocer cómo es un Router CNC para placas PCB, que tipos existen, donde se pueden encontrar en el mercado, etc. Mediante esta investigación hemos realizado algunos diseños conceptuales en donde planteamos a grandes rasgos la configuración mecánica de nuestro diseño. Es entonces donde definimos la cantidad de subconjuntos, qué método utilizaremos para la transmisión de movimientos, etc.



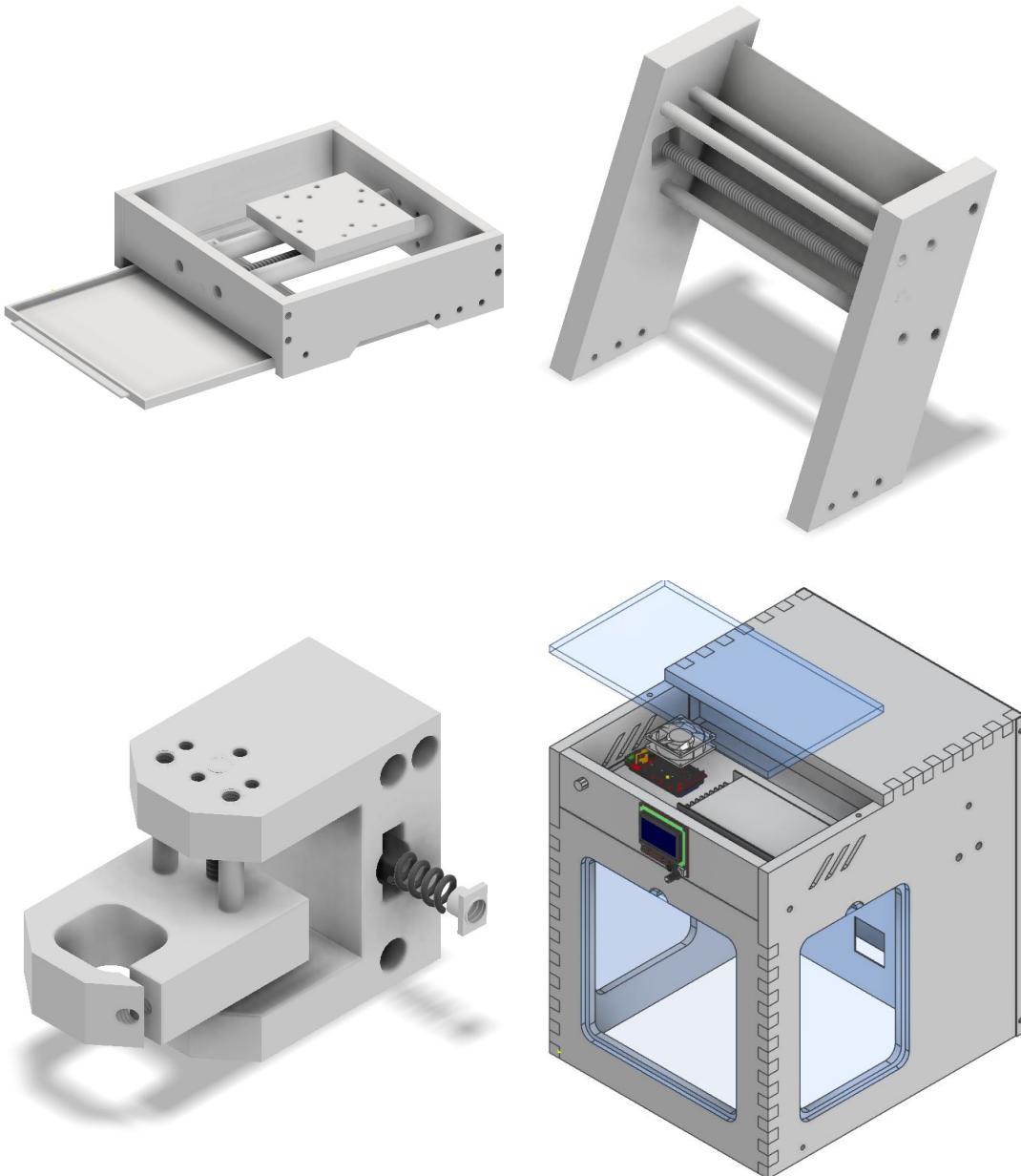
*Diseños conceptuales realizados en el mes de marzo mediante Autodesk Inventor*

Se puede apreciar cómo en esta instancia hemos definido la existencia de una “Mesa”, protagonizando el movimiento en el eje Y (de adelante hacia atrás y viceversa), la existencia de un “Subconjunto Cabezal”, protagonizando el movimiento en el eje X (de izquierda a derecha y viceversa) y la existencia de un “Soporte de Motor”, protagonizando el movimiento en el eje Z (de arriba hacia abajo y viceversa). Al mismo tiempo, definimos la existencia del Subconjunto Carcasa, con el objetivo de crear una máquina más segura para el operario (gracias a las Puertas de acrílico) y de proteger los elementos electrónicos junto otros subconjuntos.

## 2. Pre-diseños y configuración:

En esta segunda etapa, luego de adquirir los conocimientos resultado de las investigaciones y de conocer la configuración básica de nuestra máquina, decidimos realizar un estudio en profundidad de sus características constructivas y darle forma a este estudio mediante la creación de Pre-diseños.

Es aquí donde incorporamos nuevas ideas como el sistema anti-backlash. Se realizaron análisis preliminares de esfuerzos mecánicos con el objetivo de definir las características de las piezas sometidas a los mismos y sus materiales, se tuvo en cuenta el proceso de fabricación de las piezas y se definió y diseñó de forma conceptual cada una de las piezas que integran los 4 subconjuntos.



*Pre-diseños realizados en el mes de abril a través de Autodesk Inventor.*

### 3. Diseño en detalle:

En esta etapa se utilizó como insumo los Pre-diseños realizados anteriormente y propusimos como objetivo generar el diseño mecánico final de nuestra máquina. Siguiendo con nuestra metodología de diseño, este diseño final lo dividimos en 4 instancias. Un primer ensamble, denominado Ensamble 1. Un segundo ensamble, denominado Ensamble 2, un tercer ensamble, denominado Ensamble Ajuste Fino y, por último, un ensamble denominado Ensamble Ajuste Fino Total. Este último se realizó con el objetivo de ultimar detalles mecánicos y realizar un diseño de calidad.

En paralelo a estas actividades decidimos empezar a comunicarnos con proveedores y cotizar aquellas piezas que ya poseían medidas, definir la disponibilidad de las mismas, sus medidas comerciales, etc. Al mismo tiempo, realizamos diagramas de operaciones preliminares, con el objetivo de empezar a definir los procesos de mecanizado.

Durante la realización de los Ensamblés 1 y 2, que tenían como objetivo definir las dimensiones finales de todas las piezas de la máquina (a excepción del Subconjunto Carcasa) y seleccionar la bulonería a utilizar, nos encontramos con ciertos errores que cometimos durante la realización de los Pre-diseños. La gran mayoría de estos desvíos estaban relacionados con los procesos de fabricación y los costos de las piezas, por lo que fueron surgiendo a medida que contactamos con proveedores y realizamos los diagramas de operaciones preliminares. En el siguiente apartado se mencionan los desvíos sustanciales ocurridos en aquel entonces.

Durante la realización del Ensamble Ajuste Fino, donde colocamos el Subconjunto Carcasa sobre el Ensamble 2, también surgieron desvíos. Estos estaban relacionados con las Puertas y con la colocación y fabricación de la parte posterior del Subconjunto, específicamente con una pieza, a día de hoy eliminada, denominada “Chapon Trasero”.

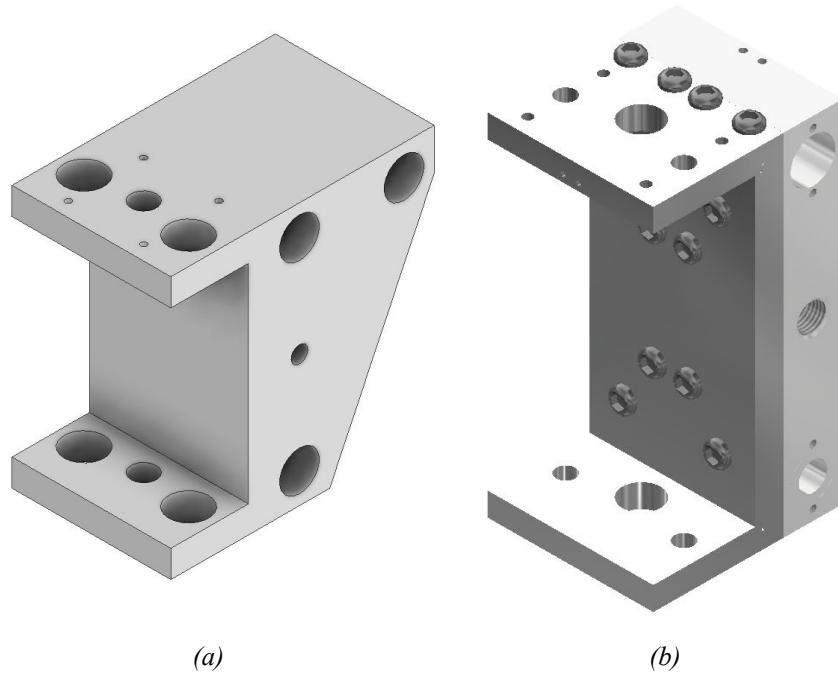
#### 3.1. Desvíos:

Los desvíos son aquellos cambios necesarios que van surgiendo a medida que avanza el diseño, el cual es un proceso en constante transformación donde, como mencionamos, se deben tomar decisiones sin la información completa, para luego volver al punto de inicio y realizar los ajustes necesarios.

- Mesa: Al realizar el Ensamble 1, diseñamos una Mesa de Acero con un espesor aproximado de  $\frac{3}{4}$  de pulgada (19,05 mm). Al avanzar con los diseños notamos que, con esas medidas, las Guías de Mesa debían ser innecesariamente grandes para soportar el peso de la Mesa, aumentando costos no solo en las Guías, sino también en las piezas que se relacionan con ellas.
  
- Vínculo Cabezal: En su momento, existió una pieza denominada “Vínculo Cabezal” que tenía como objetivo protagonizar el

movimiento en el eje X. y ser la estructura del Subconjunto Cabezal. Esta iba a ser una sola pieza de Acero o Aluminio y en ella, realizaríramos los agujeros para las Guías y el Tornillo de Vínculo Cabezal, el alojamiento para su Tuerca y otros mecanizados necesarios para las piezas que se relacionarían con el mismo. A la hora de determinar costos y el proceso de fabricación notamos que esta propuesta era completamente inviable y decidimos modificar el “Vínculo Cabezal” por 4 piezas: Techo de Vínculo Cabezal, Piso de Vínculo Cabezal, Vertical de Vínculo Cabezal y Bloque de Vínculo Cabezal. Esta modificación trajo consigo la eliminación de la tercera Guía de Vínculo Cabezal puesto que su aporte estructural era prácticamente nulo.

Esta modificación otorgó grandes beneficios tanto en los aspectos económicos, como en los aspectos relacionados con la resistencia mecánica y los mecanizados.



Uno de los primeros diseños del Vínculo Cabezal (a) y su modificación (b)

- Puertas en el Subconjunto Carcasa: En un principio las 4 Puertas de este Subconjunto se soportarían mediante un sistema de encastre e imanes con el objetivo teórico de facilitar la apertura de las mismas. La incorporación de estos imanes y su funcionamiento lo definimos como poco confiable y, por ello, optamos por utilizar un sistema de bisagras en 3 de las 4 Puertas. La Puerta restante (Puerta Lateral Izquierda) complicaba el funcionamiento óptimo del sistema de bisagras y, por lo tanto, quedaría fija.

En este punto ya resolvimos las complicaciones dadas por el sistema de sujeción mediante imanes pero no definimos un sistema que permita un cierre adecuado de las Puertas. Con el tiempo dimos con cerraduras denominadas Cerraduras de Leva, estas constituyen a día de hoy la solución más adecuada a esta problemática, asegurando un cierre completamente confiable.

- Cambio de Rodamientos: Terminando el Ensamble 2, reemplazamos los Rodamientos Lineales por Bujes de Resina Acetal con el objetivo de disminuir costos y facilitar el cambio de los mismos provocado por el desgaste. Con este mismo objetivo creamos las Tapas para Bujes, las cuales evitan que los Bujes se desplacen y salgan de su alojamiento. En caso de necesitar cambiar los Bujes, se deben quitar las Tapas, colocar el Buje para Guía nuevo y colocar, nuevamente, las Tapas en su posición.
- Chapon Trasero: En el Pre-diseño planteamos la existencia de un Chapón Trasero. Una chapa de aluminio de grandes dimensiones que se colocaría en la parte posterior de la máquina. A la hora de diseñar esta pieza nos encontramos con grandes problemas relacionados con la resistencia mecánica de la misma, su sujeción era relativamente complicada y su fabricación innecesariamente compleja. Luego de varios diseños, optamos por crear una pieza denominada “Culata Carcasa” que ocupa el mismo espacio y posee el mismo objetivo

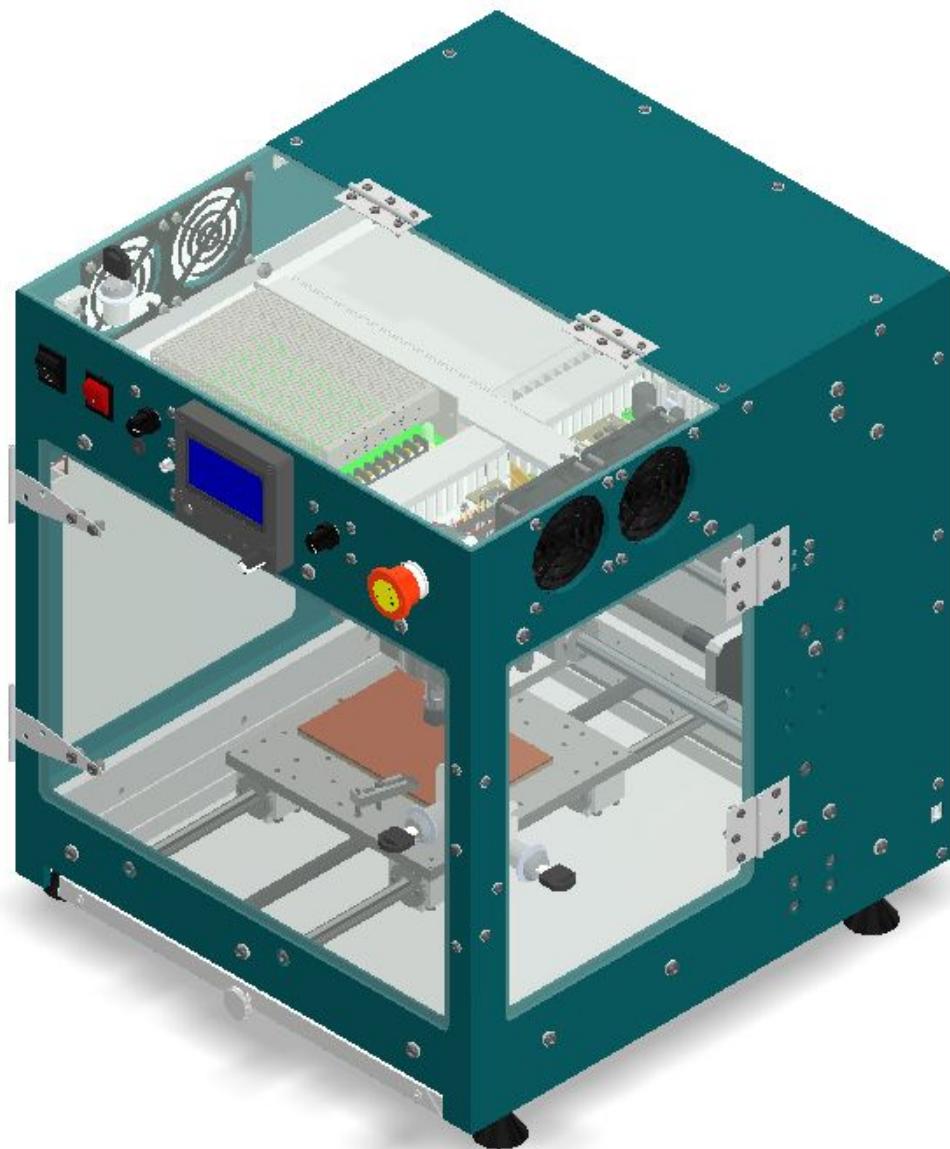
### 3.2. Ensamble Ajuste Fino Total + Electrónica:

Luego de enfrentar y resolver los desvíos antes plasmados y otras dificultades propias de la realización del proyecto, realizamos el Ensamble Ajuste Fino Total. Como mencionamos, este ensamble tenía como objetivo ultimar todos los detalles mecánicos de la máquina y, de ese modo, darnos la posibilidad de comenzar a producir los documentos necesarios para la cuarta y última etapa de nuestra metodología de diseño, es decir, “Planeación de cara a la manufactura”. Es así como comenzamos con los Planos Mecánicos 2D, evidentemente, tomando como referencia el Manual de Normas IRAM de Dibujo Tecnológico, edición 2009.

En paralelo a estas actividades puramente mecánicas, decidimos realizar el último ensamble de nuestra máquina. Como todos los detalles mecánicos ya estaban terminados, en este ensamble colocamos todos los componentes electrónicos de la misma y, como consecuencia, obtuvimos un diseño lo más detallado posible. De esta manera comenzamos con el “Ensamble Ajuste Fino Total + Electrónica”, el cual se puede observar en el Anexo C: *PDF 3D - Ensamble Ajuste Fino Total + Electrónica*.

Para una mayor organización y teniendo en cuenta que este último ensamble contiene más de 1000 elementos, decidimos realizar la planilla “Datos - Ensamble Ajuste Fino Total + Electrónica”. En ella se resumen absolutamente todos los detalles de este ensamble, desde los nombres de las piezas y sus códigos hasta el objetivo de todos y cada uno de los tornillos, arandelas y tuercas. Para su lectura ver Anexo A: *Datos - Ensamble Ajuste Fino Total + Electrónica*.

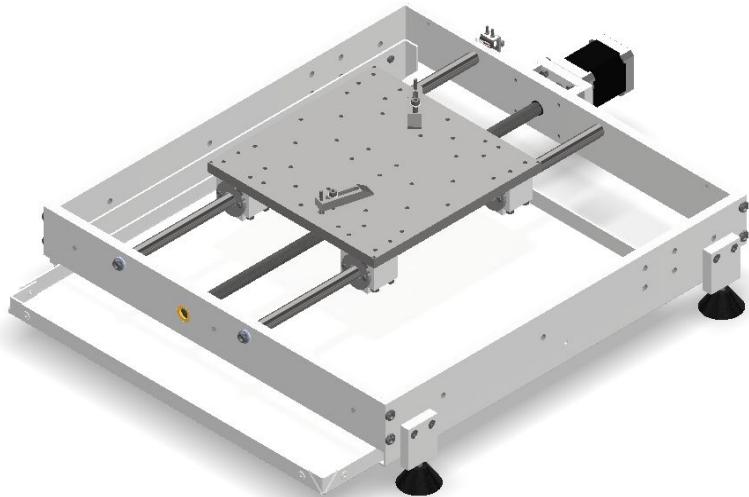
A continuación mostraremos en profundidad cada subconjunto que compone nuestra máquina, por supuesto, tomando como referencia el “Ensamble Ajuste Fino Total + Electrónica”.



*Ensamble Ajuste Fino Total + Electrónica realizado en Autodesk Inventor*

### 3.2.1. Subconjunto Inferior<sup>2</sup>:

El diseño del presente Subconjunto es el siguiente:



*Subconjunto Inferior realizado en Autodesk Inventor*

Este Subconjunto está compuesto por la siguientes piezas:

Cantidad	Nombre de Pieza	Cantidad	Nombre de Pieza
1	Acople Rígido - Eje Y	1	Manija para Bandeja
4	Buje para Guía	1	Mesa
1	Bandeja	1	Motor Paso a Paso - Eje Y
1	Buje para Tornillo - Eje Y	2	Pared Espaciador - Sub. Inferior
2	Chaponete	4	Patas Regulables
1	Culata	1	Resorte para Tuerca de Mesa
4	Esquina de Bandeja	4	Soporte de Buje para Guía
1	Frente	4	Soporte de Patas Regulables
1	Guía de Bandeja Izquierda	1	Soporte de Tuerca de Mesa
1	Guía de Bandeja Derecha	8	Tapa para Buje
2	Guía de Mesa	1	Techo Espaciador - Sub. Inferior
1	Lateral Izquierdo	1	Tornillo de Mesa
1	Lateral Derecho	1	Tuerca de Mesa

*Datos extraídos de Anexo A: Datos - Ensamble Ajuste Fino Total + Electrónica.*

<sup>2</sup>

Para observar con mayor detalle este subconjunto, ver Plano “Subconjunto Inferior - Explosión” y “Subconjunto Inferior - Ensamble”. GI-A y GI-AA, respectivamente.

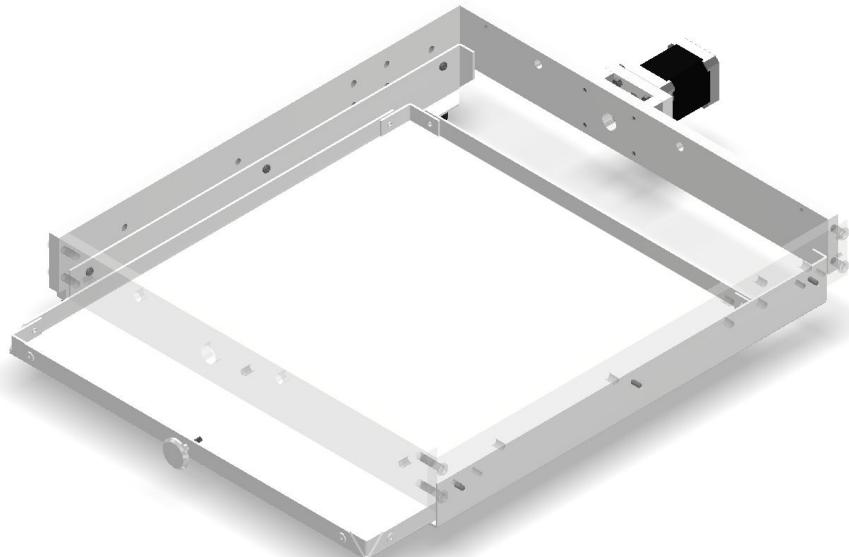
Para su explicación, dividiremos este Subconjunto en las siguientes partes:

- Bandeja y piezas relacionadas;
- Estructura y Motor Paso a Paso - Eje Y;
- Mesa y piezas relacionadas.

#### - **Bandeja y piezas relacionadas:**

La Bandeja tiene como objetivo alojar la viruta que se generará durante el mecanizado. En su frente posee una Manija para Bandeja para facilitar su extracción. En sus esquinas, se encuentran las Esquinas de Bandeja, estas 4 piezas se encargan de mantener la forma de la Bandeja y evitar que sus pliegues se abran.

Para permitir el movimiento de la Bandeja, se encuentra la Guía de Bandeja Derecha y la Guía de Bandeja Izquierda, sujetadas a los Laterales de la máquina. Al igual que la Bandeja, estas piezas poseen cortes a 45° para facilitar su plegado.



*Bandeja y piezas relacionadas - Subconjunto Inferior*

#### - **Estructura y Motor Paso a Paso - Eje Y:**

La estructura de este Subconjunto está compuesta por el Frente, la Culata y los Laterales (Izquierdo y Derecho). Sujetos a estos últimos, se encuentran los Soportes para Patas Regulables que, como su nombre lo indica, contienen las Patas Regulables. Al mismo tiempo, están provistos de agujeros para sujetar las Columnas y los Laterales Carcasa.

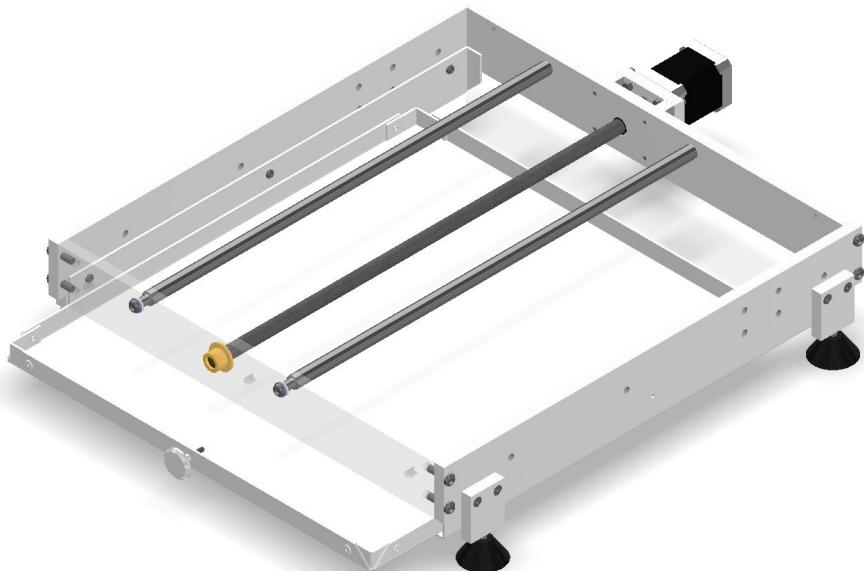
El Lateral Derecho particularmente posee pequeños agujeros roscados en uno de sus cantos cortos con el objetivo de sujetar la placa USB-B Hembra.

La Culata posee agujeros para unir tanto las Paredes y el Techo Espaciador como el Chapon.

Por otro lado, como parte de la estructura, se encuentra el Tornillo de Mesa, con métrica de MF12x1,5. Uno de sus extremos se adosa al Eje del Motor Paso a Paso mediante el Acople Rígido - Eje Y y el otro se coloca en el Buje para Tornillo - Eje Y. Este último se inserta a presión sobre el Frente mediante un Sistema de Ajuste Fino de Deslizamiento (tolerancias H7; j6).

Finalmente en la estructura encontramos las Guías de Mesa, estas tienen el objetivo de darle mayor estabilidad al movimiento de la Mesa. Se vinculan a los Bujes para Guias mediante un Sistema de Ajuste Fino de Juego Libre muy Justo (tolerancias H6; f7). En sus extremos poseen pequeños rebajes para facilitar su sujeción al Frente y a la Culata y para aportar mayor estabilidad estructural.

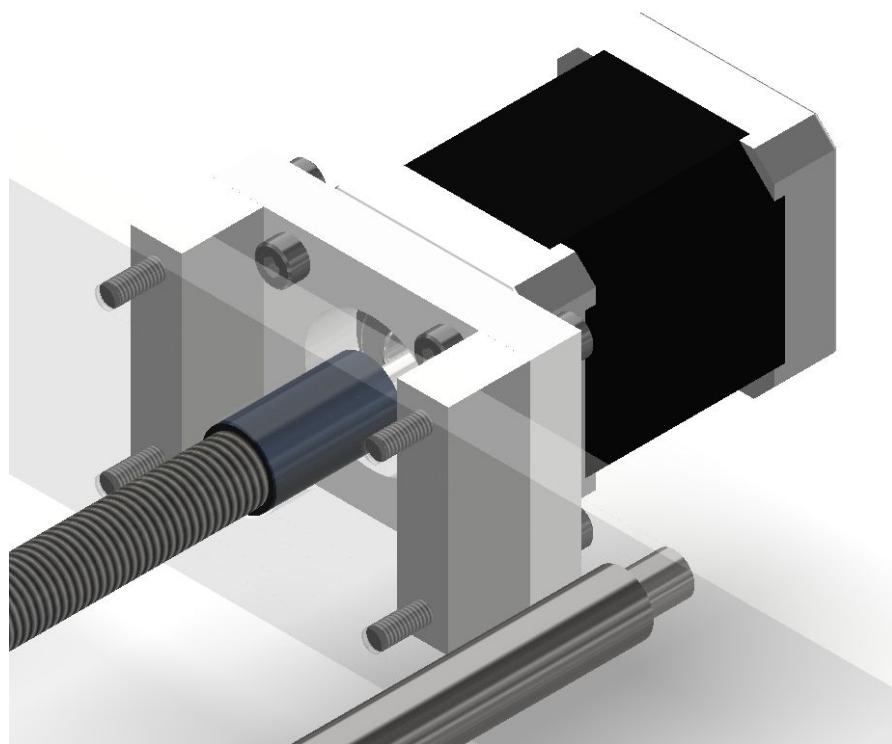
El Motor Paso a Paso - Eje Y, es un NEMA 17 de Alto Torque<sup>3</sup>. El mismo se sujetta mediante tornillos al Techo Espaciador y, este último, a la Culata mediante las Paredes Espaciador. Estas 3 piezas tienen el objetivo de generar el espacio suficiente para la colocación del Acople Rígido - Eje Y que, como mencionamos, vincula el eje del Motor Paso a Paso con el Tornillo de Mesa.



*Estructura - Subconjunto Inferior*

<sup>3</sup>

Para más información dirigirse al apartado “Motores Paso a Paso” situado en “Electrónica, Electricidad y Control”.



*Motor Paso a Paso - Eje Y - Subconjunto Inferior*

- **Mesas y piezas relacionadas:**

La “Mesa” es de 200 mm por 200 mm. Se encuentra provista de gran cantidad de agujeros roscados para permitir la sujeción tanto de los “Chaponetes” como de la placa a mecanizar. En la primer figura se muestra una placa PCB de 10 cm x 15 cm y, debajo de ella, una Placa de Mecanizado (Madera de Sacrificio) de dimensiones similares.

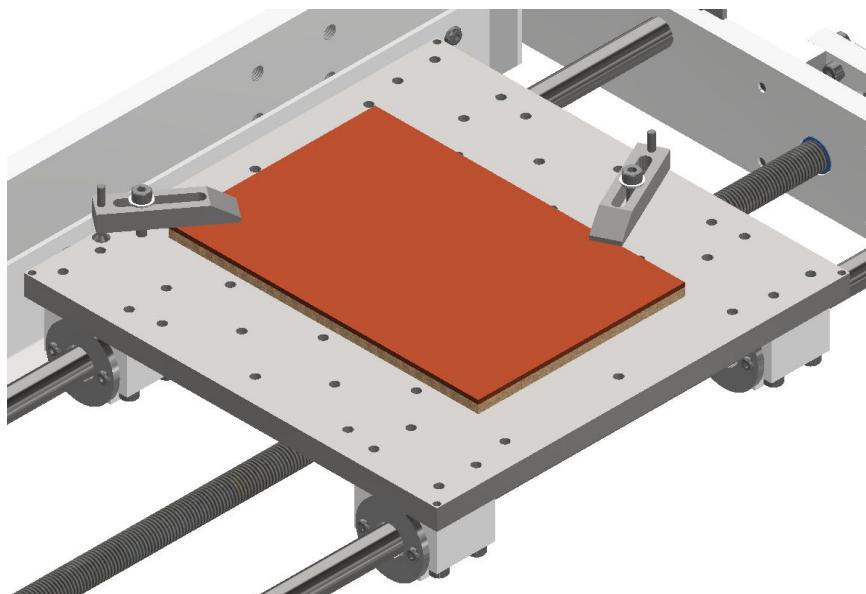
Los “Chaponetes” poseen una ranura por la cual se podrá desplazar un tornillo que tiene como función sujetarlo a la Mesa. A su vez, poseen un agujero roscado en su parte posterior donde se colocará un tornillo cabeza fresada que funcionará como apoyo del Chaponete. Ambos tornillos pueden intercambiarse en función de las necesidades a la hora del mecanizado.

En la parte inferior de la Mesa se encuentran los Soportes de Bujes para Guías, las Tapas para Bujes, los Bujes para Guías, el Soporte de Tuerca de Mesa y la Tuerca de Mesa.

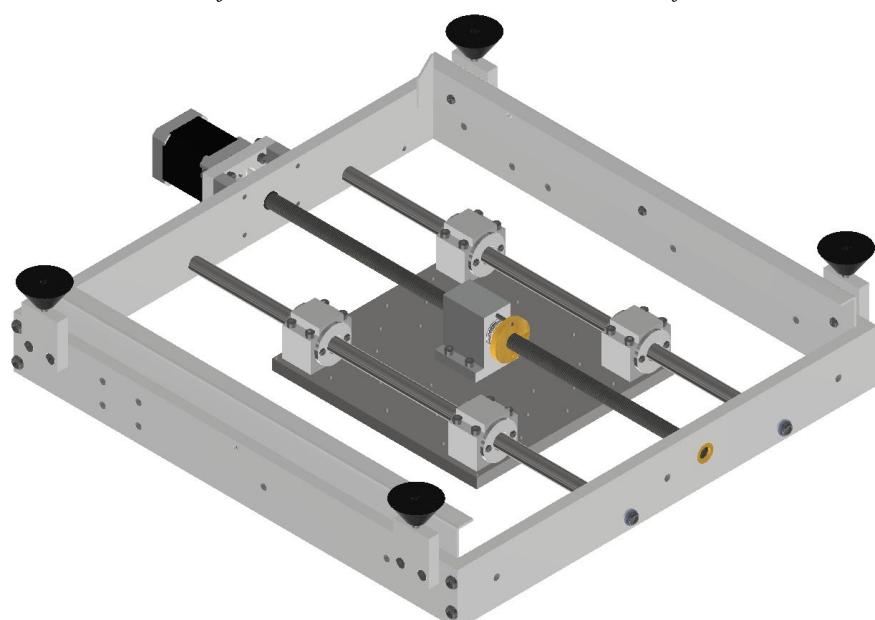
Los Soportes de Bujes para Guías tienen la función de alojar los Bujes para Guías de Resina Acetal. Estos Bujes se colocan mediante un sistema de Ajuste Corriente Deslizante (H8; h9) y, para evitar su desplazamiento, se utilizan las Tapas para Bujes. Dichas Tapas se sujetarán mediante tornillos a los Soportes de Bujes para Guías. Este sistema de Buje-Tapa tiene como objetivo facilitar el reemplazo de los Bujes a la hora de realizar un mantenimiento.

La “Tuerca de Mesa”, lógicamente, posee rosca interna MF12x1,5. En su cabeza posee agujeros no roscados para alojar los tornillos gusano que se encuentran en el “Soporte de Tuerca de Mesa”, estos tornillos tienen la finalidad de evitar que la Tuerca rote sobre su eje.

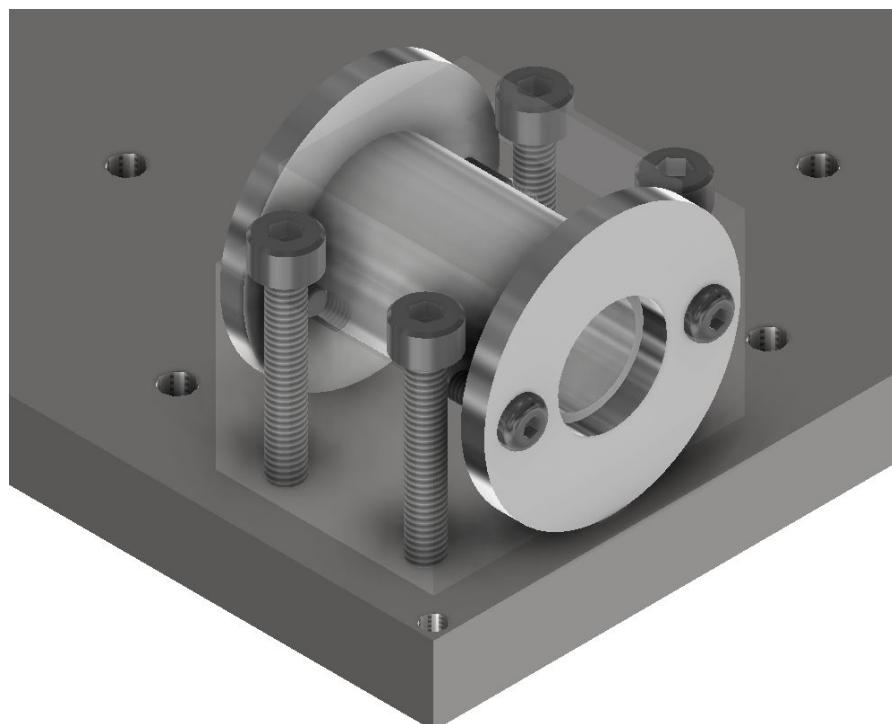
El Soporte de Tuerca de Mesa posee un agujero pasante roscado para que el Tornillo de Mesa pase por él y, en el mismo eje, posee otro agujero ciego de mayor diámetro. Este último tiene la finalidad de alojar la Tuerca de Mesa y el Resorte para Tuerca de Mesa que evita el juego axial (este sistema “Resorte-Tuerca” se denomina, Sistema Anti-Backlash).



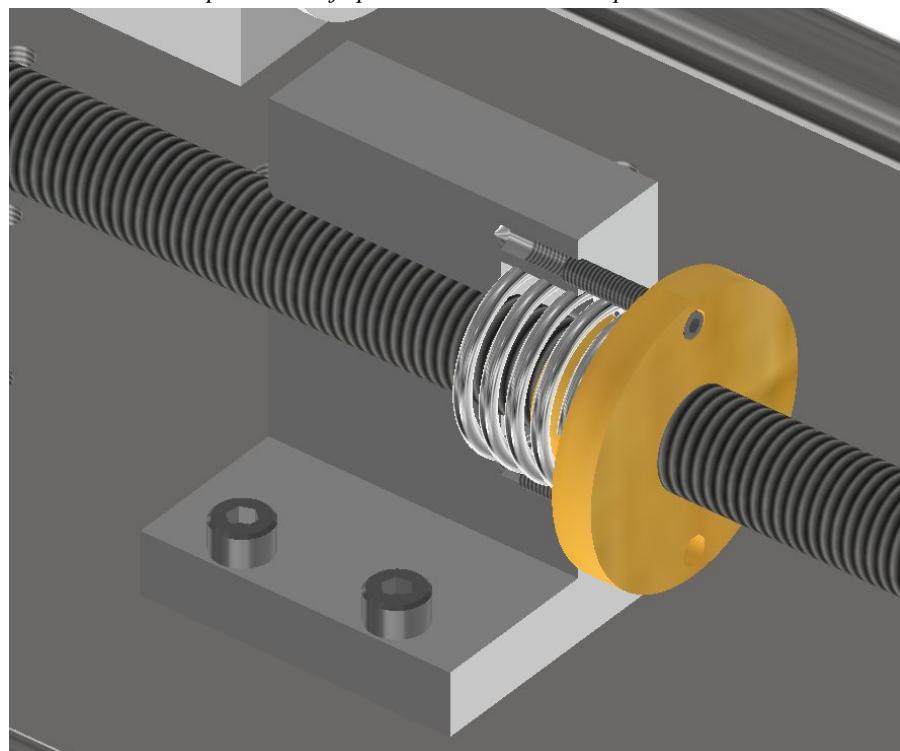
*Mesa junto con Chaponetes, sujetando una placa de cobre virgen de 10 cm x 15 cm.  
Debajo de ella se encuentra la Madera de Sacrificio*



*Vista Inferior de Mesa y piezas relacionadas - Subconjunto Inferior.*



*Soporte de Buje para Guía, Buje para Guía y Tapas Para Bujes. Nótese que el Soporte de Buje para Guía ha sido transparentado.*



*Corte del Soporte de Tuerca de Mesa. Aquí se ve claramente el sistema Anti-Backlash.*

### 3.2.2. Subconjunto Puente<sup>4</sup>:

El diseño del presente Subconjunto es el siguiente:



*Subconjunto Puente realizado en Autodesk Inventor*

Este Subconjunto está compuesto por la siguientes piezas:

Cantidad	Nombre de Pieza	Cantidad	Nombre de Pieza
1	Acople Rígido - Eje X	2	Guías de Vínculo Cabeza
1	Buje para Tornillo - Eje Y	1	Motor Paso a Paso - Eje X
1	Columna Derecha	1	Soporte de Final de Carrera - Eje X
1	Columna Izquierda	1	Tornillo de Vínculo Cabezal
1	Final de Carrera - Eje X	1	Travesaño Estructural

*Datos extraídos de Anexo A: Datos - Ensamble Ajuste Fino Total + Electrónica.*

En el presente Subconjunto contamos con dos Columnas, Derecha e Izquierda, ambas tienen como objetivo soportar todas las piezas encargadas de la transmisión y el movimiento (Guías, Tornillos, Motor Paso a Paso, etc.). Al mismo tiempo, cada una está provista de agujeros roscados que se utilizan para sujetar ambos Laterales Carcasa.

<sup>4</sup>

Para observar con mayor detalle este subconjunto, ver Plano “Subconjunto Puente - Explosión” y “Subconjunto Puente - Ensamble”. G1-B y G1-BB, respectivamente.

Entre las Columnas se encuentra el Travesaño Estructural, esta pieza tiene como objetivo aportar mayor rigidez al Subconjunto en particular y a la máquina en general.

El Motor Paso a Paso - Eje X (también NEMA 17 de Alto Torque) se sujetta desde su parte posterior, notable diferencia con respecto a los otros Subconjuntos. Al igual que en el Subconjunto anterior, el eje del Motor se vincula al Tornillo de Vínculo Cabezal mediante el Acople Rígido - Eje X.

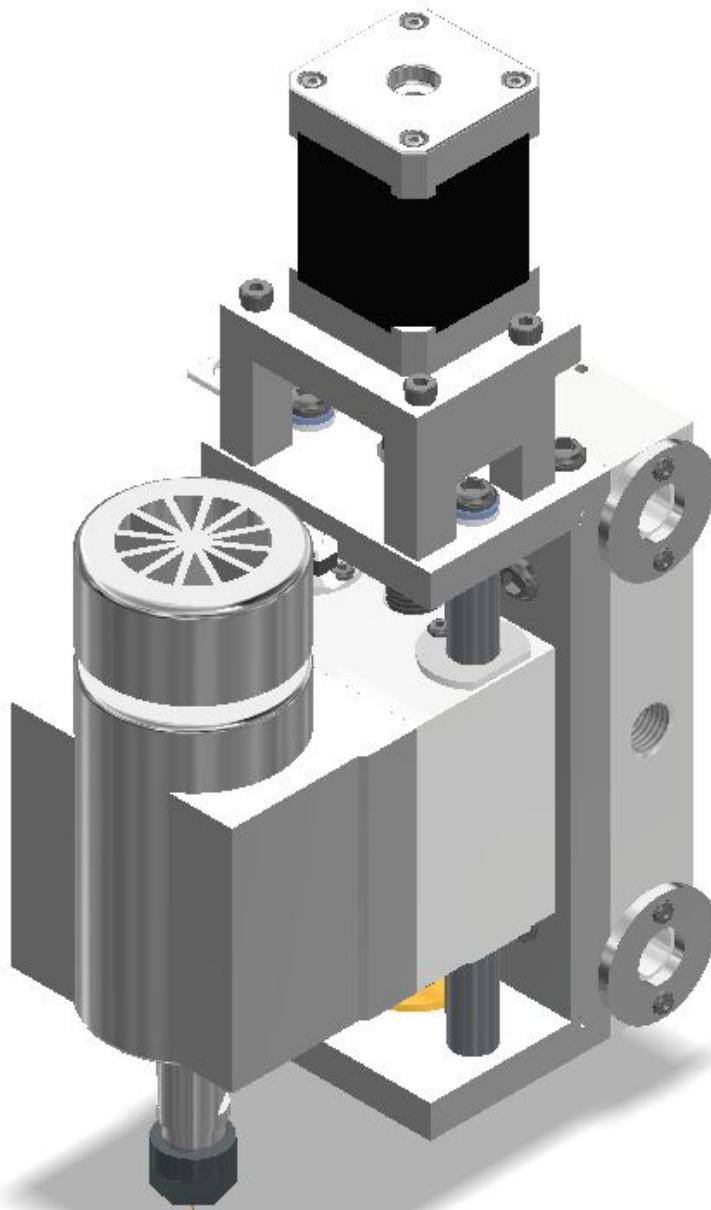
El Tornillo de Vínculo Cabezal también es MF12x1,5. En sus extremos posee rebajes que permiten la unión al Acople Rígido - Eje X y al Buje para Tornillo - Eje X. Este último se inserta a presión sobre la Columna Izquierda mediante un Sistema de Ajuste Fino de Deslizamiento (tolerancias H7; j6)

Las Guías de Vínculo Cabezal tienen el mismo objetivo que todas las Guías de máquina, es decir, facilitar el movimiento de su carro, en este caso, del Subconjunto Cabezal. Al igual que en el Subconjunto anterior, en sus extremos poseen rebajes con el objetivo de permitir su sujeción y de aumentar la estabilidad estructural.

Finalmente, existe un Soporte de Final de Carrera - Eje X. Como su nombre lo indica, este sujetta el Final de Carrera del Eje X cuyo objetivo es limitar el movimiento en dicho eje, protagonizado por el Subconjunto Cabezal

### 3.2.3. Subconjunto Cabezal<sup>5</sup>:

El diseño del presente Subconjunto es el siguiente:



*Subconjunto Cabezal realizado en Autodesk Inventor*

---

<sup>5</sup>

*Para observar con mayor detalle este subconjunto, ver Plano “Subconjunto Cabezal - Explosión” y “Subconjunto Cabezal - Ensamble”. G1-C y G1-CC, respectivamente.*

Este subconjunto está compuesto por las siguientes piezas:

Cantidad	Nombre de Pieza	Cantidad	Nombre de Pieza
1	Acople de Soporte de Motor	1	Piso de Vínculo Cabezal
1	Acople Rígido - Eje Y	1	Resorte para Tuerca de Vínculo Cabezal
1	Bloque de Vínculo Cabezal	2	Separador de Bujes
4	Buje de Soporte de Motor	1	Motor DC
4	Buje para Guía	1	Soporte de Motor
1	Buje para Tornillo - Eje Z	4	Tapa para Buje
1	Final de Carrera - Eje Z	1	Techo de Vínculo Cabezal
2	Guía de Soporte de Motor	1	Techo Espaciador
1	Guía para Cable de Motor	1	Tornillo de Soporte de Motor
1	Motor Paso a Paso - Eje Z	1	Tuerca de Vínculo Cabezal
1	Pared Espaciador Izquierda	1	Vertical de Vínculo Cabezal
1	Pared Espaciador Derecha		

Datos extraídos de Anexo A: Datos - Ensamble Ajuste Fino Total + Electrónica.

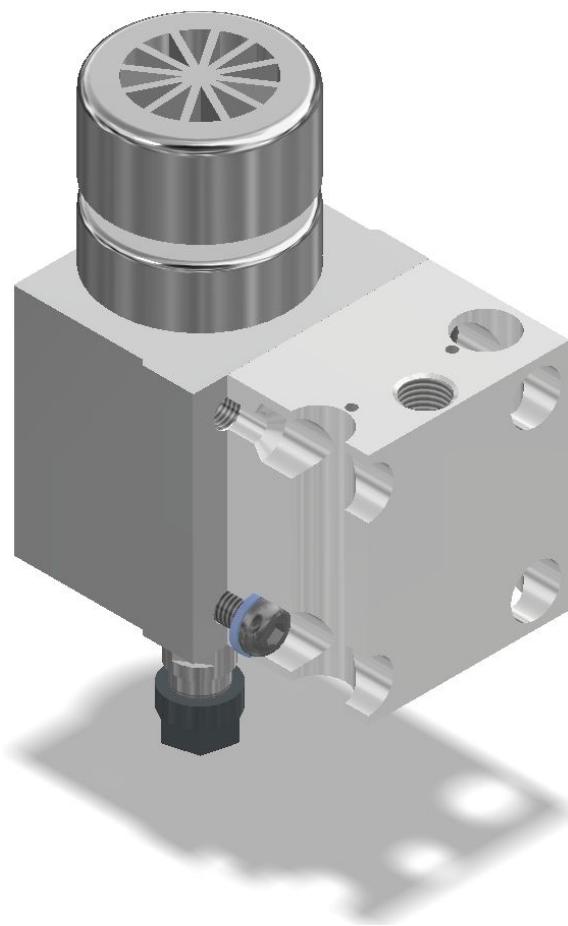
Para su explicación, lo dividiremos en las siguientes partes:

- Motor DC y piezas relacionadas;
- Estructura;
- Motor Paso a Paso - Eje Z y piezas relacionadas.

#### - Motor DC y piezas relacionadas:

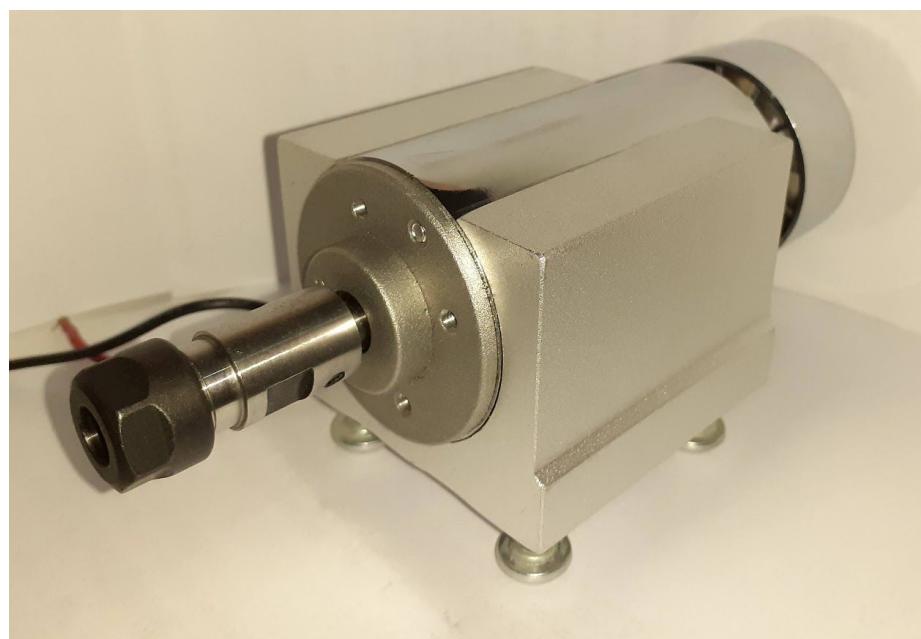
El Acople de Soporte de Motor tiene el objetivo de permitir el movimiento en el eje Z protagonizado por el Motor DC y el Soporte de Motor. La unión con este último se realiza mediante tornillos ubicados en los agujeros internos del Acople de Soporte de Motor. La presencia de las arandelas grower resulta fundamental, pues tienen como objetivo asegurar una correcta sujeción del Soporte de Motor.

En la siguiente imagen se observa la unión mencionada.



*Acople de Soporte de Motor en unión con Soporte de Motor - Subconjunto Cabeza.*

Es importante destacar que el Motor DC se encuentra colocado a presión en su Soporte, este tipo de sujeción asegura una eliminación prácticamente total de las vibraciones a la hora del mecanizado.



Al mismo tiempo, el Acople de Soporte de Motor posee agujeros dentro de los cuales se colocarán los Bujes de Soporte de Motor mediante un Sistema de Ajuste Corriente de Juego Libre (Tolerancias H8; f8). Teniendo en cuenta las características de este sistema de tolerancias, se colocan unos tornillos que tienen como objetivo evitar que el Buje rote sobre su eje y/o se deslice. Al mismo tiempo, el sistema utilizado para que las Guías de Soporte de Motor se deslicen sobre los Bujes es un Sistema de Ajuste Fino de Juego Libre muy Justo (tolerancias H7; f7)

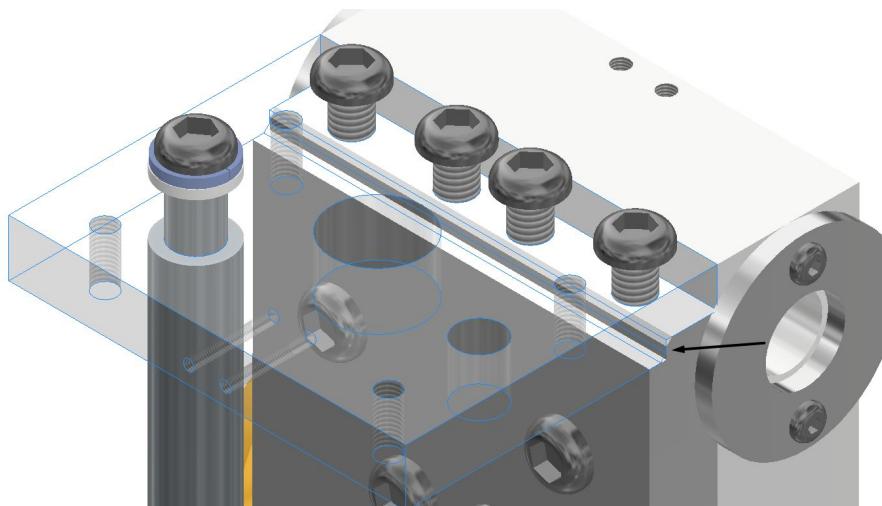
- **Estructura:**

Como se mencionó en el apartado “Desvíos”, la estructura del presente Subconjunto está compuesta por las siguientes piezas:

- Piso de Vínculo Cabezal;
- Techo de Vínculo Cabezal;
- Vertical de Vínculo Cabezal;
- Bloque de Vínculo Cabezal

El Techo, el Piso y la Vertical de Vínculo Cabezal tienen la finalidad de darle rigidez al Subconjunto, soportar las Guías de Soporte de Motor, el Tornillo de Soporte de Motor, el Motor Paso a Paso - Eje Z y demás piezas que lo integran. Estas tres piezas se vinculan mediante tornillos y, para asegurar un correcto contacto y encastre, poseen rebajes con chaflanes a 45° como se puede apreciar en la siguiente imagen.

El Final de Carrera - Eje Z se sujetta al Techo de Vínculo Cabezal y tiene como objetivo limitar la carrera del Soporte de Motor junto con el Motor DC

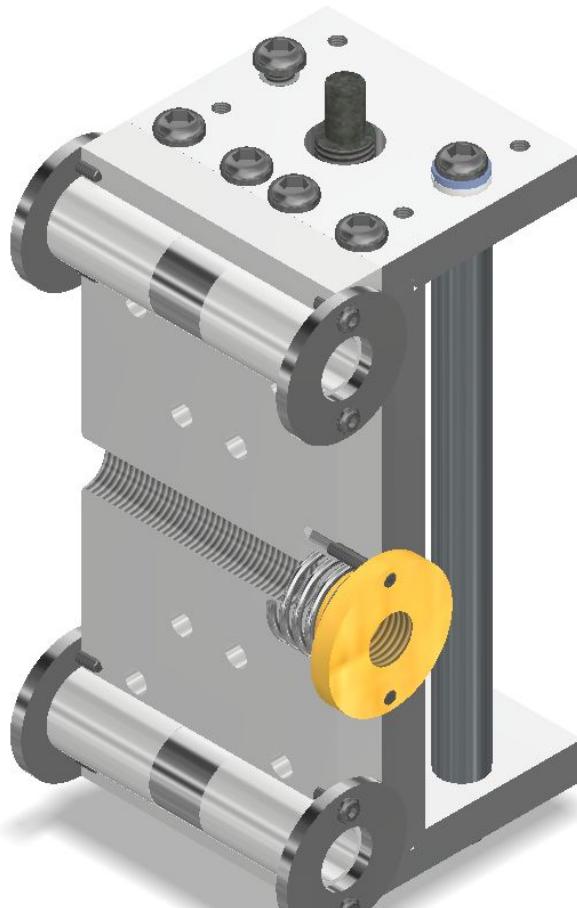


*Unión entre Techo y Vertical de Vínculo Cabezal. Nótese los rebajes y chaflanes presentes en ambas piezas.*

El Bloque de Vínculo Cabezal se encuentra en la parte posterior del Subconjunto y tiene la función de permitir el movimiento del mismo. En su interior se alojan los Bujes para Guías y, entre ellos, un Separador de Bujes que, como su nombre indica, tiene la finalidad de separar ambos Bujes para facilitar su remoción en caso de realizar un reemplazo o mantenimiento. Como estos Bujes para Guía poseen el mismo sistema de tolerancias que aquellos presentes en el Subconjunto Inferior, se utiliza nuevamente el sistema “Buje-Tapa”. Estas Tapas se sujetan mediante tornillos al Bloque de Vínculo Cabezal.

Al mismo tiempo, el Bloque posee un alojamiento para el Resorte para Tuerca de Vínculo Cabezal (con mismas características que el resorte presente en el Subconjunto Inferior) y para la Tuerca de Vínculo Cabezal y el respectivo agujero roscado para el Tornillo de Vínculo Cabezal (presente en el Subconjunto Puente).

En la siguiente imagen se pueden apreciar las piezas mencionadas.



*Corte en la parte posterior del Bloque de Vínculo Cabezal - Subconjunto Cabezal.*

El Tornillo de Soporte de Motor también es MF12x1,5 y, nuevamente, uno de sus extremos está unido al Acople Rígido - Eje Z y el otro al Buje para Tornillo - Eje Z colocado en el Piso de Vínculo Cabezal con el mismo sistema de tolerancias que los Bujes para Tornillos presentes en los anteriores Subconjuntos.

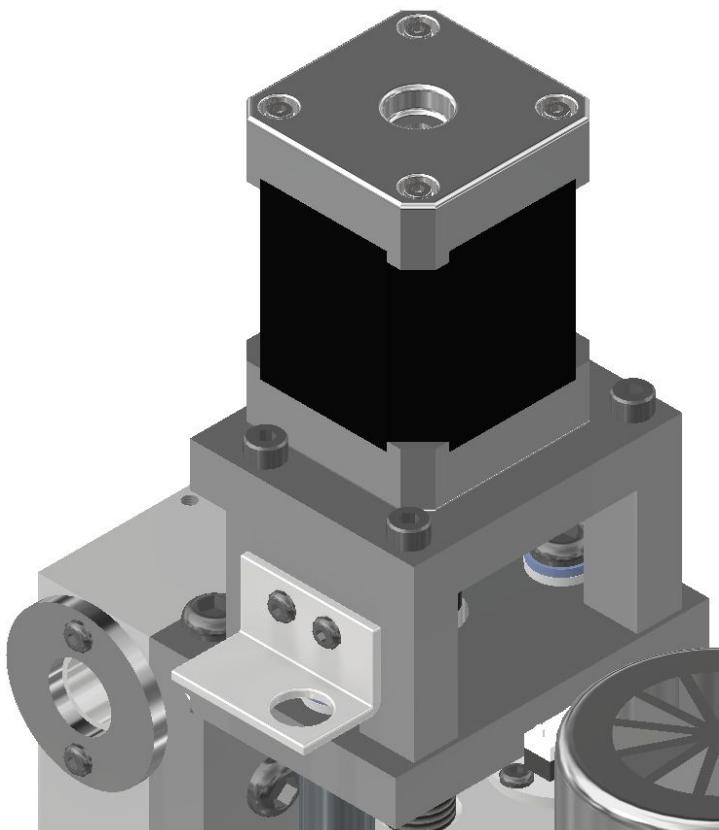
Las Guías de Soporte Motor también poseen unos pequeños rebajes en sus extremos y se sujetan tanto al Piso de Vínculo Cabezal como al Techo de Vínculo Cabezal mediante tornillos.

- **Motor Paso a Paso - Eje Z y piezas relacionadas:**

Al igual que en Subconjunto Inferior, el Subconjunto Cabezal posee 3 piezas (Paredes Espaciador y Techo Espaciador) que separan al Motor Paso a Paso (NEMA 17 de Alto Torque) de la estructura del Subconjunto.

Las Paredes se colocan sobre el Techo de Vínculo Cabezal y, sobre ellas, se sitúa el Techo Espaciador. Lógicamente, el Motor Paso a Paso se coloca mediante tornillos sobre este último.

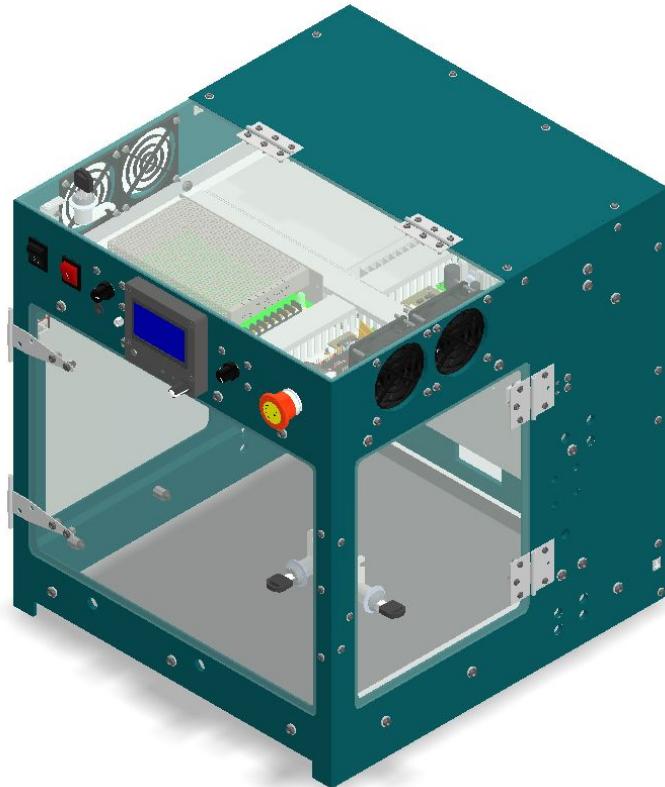
Sujeto a la Pared Espaciador Izquierda se encuentra la Guía para Cable de Motor. Esta tiene como objetivo direccionar los cables del Motor DC hacia la Cadena Portacables, evitando posibles daños en los mismos.



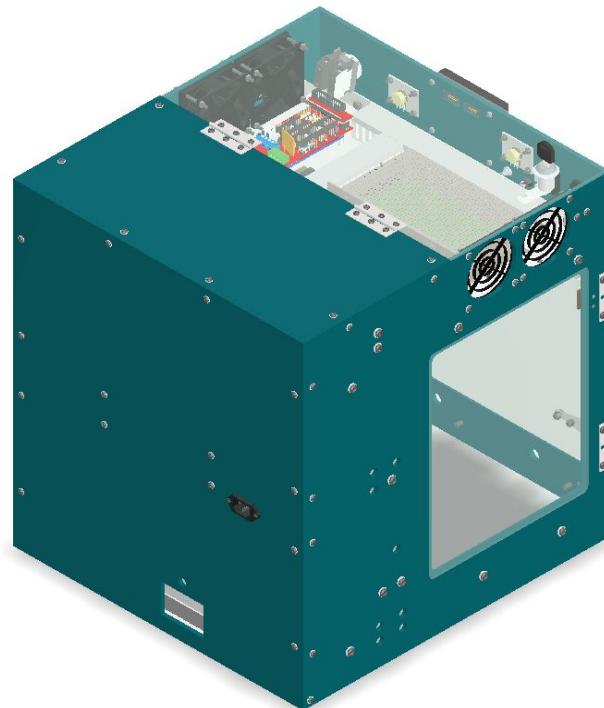
*Motor Paso a Paso - Eje Z y piezas relacionadas - Subconjunto Cabezal.*

### 3.2.4. Subconjunto Carcasa<sup>6</sup>:

El diseño del presente Subconjunto es el siguiente:



*Vista delantera del Subconjunto Carcasa realizado en Autodesk Inventor*



*Vista posterior del Subconjunto Carcasa realizado en Autodesk Inventor.*

---

<sup>6</sup>

Para observar con mayor detalle este subconjunto, ver Plano “Subconjunto Carcasa - Explosión” y “Subconjunto Carcasa - Ensamble”. G1-D y G1-DD, respectivamente.

Este subconjunto está compuesto por las siguientes piezas:

Cantidad	Nombre de Pieza	Cantidad	Nombre de Pieza
2	Acople L para Soporte LED	1	Lateral Izquierdo Carcasa
2	Bisagra tipo Libro Chica	1	Puerta de Inspección
2	Bisagra tipo Libro Grande	1	Puerta Frontal
2	Bisagra tipo T	1	Puerta Lateral Derecha
3	Cerradura de Leva - Ø:18; L:25 mm	1	Puerta Lateral Izquierda
2	Chapa para Potenciómetro	2	Separador Lateral
1	Chapa de Refrigeración	1	Soporte para Final de Carrera - Eje Y
1	Chapa Base	1	Soporte L Frontal
1	Chapón	7	Soporte L
1	Cielo Raso	1	Soporte LED
1	Culata Carcasa	1	Soporte para Culata Carcasa Izq.
6	Distanciador para Laterales	1	Soporte para Culata Carcasa Der.
1	Final de Carrera - Eje Y	1	Traba para Cerradura
1	Frente Carcasa	1	Tope para Cerradura Frontal
1	Cadena Portacables	1	Tope para Cerradura Lateral
1	Lateral Derecho Carcasa	1	Techo Carcasa

Datos extraídos de Anexo A: Datos - Ensamble Ajuste Fino Total + Electrónica.

Para su explicación, dividiremos el Subconjunto en las siguientes partes:

- Estructura Externa;
- Estructura Interna.

#### - Estructura Externa:

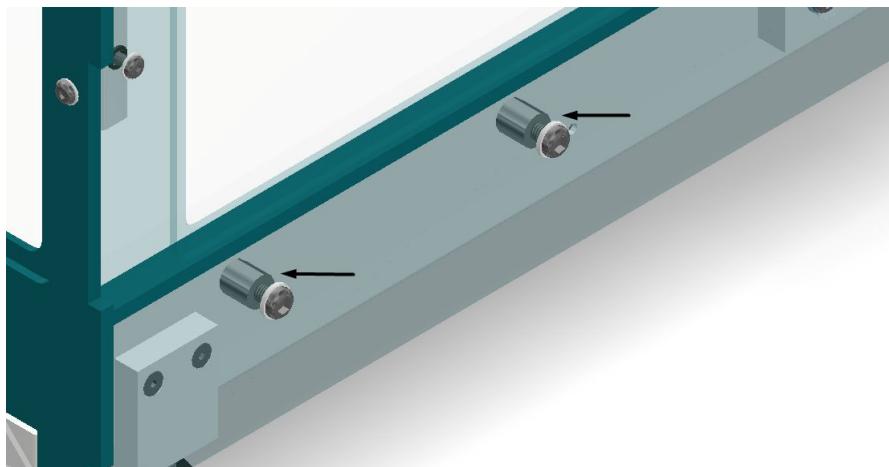
La estructura externa del Subconjunto está compuesta por el Frente Carcasa, Lateral Derecho e Izquierdo Carcasa, Culata Carcasa, Techo Carcasa, Separador Lateral, Chapa Base y las 4 Puertas (Frontal, Lateral Izquierda, Lateral Derecha y de Inspección).

El Frente, la Culata, los Laterales y el Techo Carcasa se unen entre sí utilizando ranuras de encastre y, en ciertos puntos, mediante

los Soportes L y el Soporte L Frontal. La función de estos últimos es la de aportar rigidez estructural.

Entre los Laterales Carcasa y los Laterales de la máquina, se encuentran los Separadores Laterales. Su función, meramente estética, es la de ocultar el espacio existente entre las piezas mencionadas.

Al mismo tiempo, el Frente y los Laterales Carcasa se sujetan a la estructura interna de la máquina y, entre ellos, se encuentran los Distanciadores para Laterales. En cambio, la Culata Carcasa se sujeta al Travesaño Estructural mediante los Soportes para Culata Carcasa Izquierdo y Derecho.



*Distanciadores para Laterales - Subconjunto Carcasa*

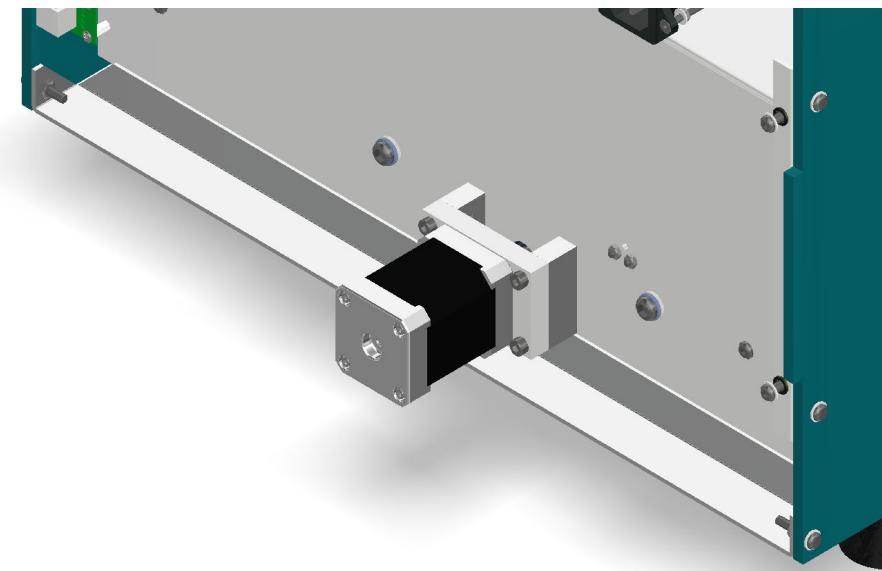
La Puerta Lateral Izquierda se encuentra pegada al Lateral Izquierdo Carcasa. La Puerta Lateral Derecha se abre gracias a dos Bisagras tipo Libro Grandes de 180°. La Puerta Frontal se abre gracias a dos Bisagras tipo T de 270°. Por último, la Puerta de Inspección se abre mediante dos Bisagras tipo Libro Chicas de 180°. Clavados y pegados en las Puertas Frontal y Lateral Derecha y, haciendo tope con sus respectivas Bisagras superiores, se encuentran dos imanes de neodimio con el objetivo de controlar el Circuito de Seguridad<sup>7</sup>

Todas las Puertas, a excepción de la Lateral Izquierda, se cierran mediante Cerraduras de Leva de un cuarto de vuelta. El Tope para Cerradura Frontal, el Tope para Cerradura Lateral y la Traba para Cerradura tiene el objetivo permitir el correcto cierre de las mismas.

La Chapa Base se encuentra en la parte inferior de la máquina, sujeta a los Laterales Carcasa. Si bien en un principio se diseñó con fines puramente estéticos, evita la entrada de polvo.

<sup>7</sup>

Para más información dirigirse al apartado “Circuito de Seguridad” situado en “Placas de Control”.



*Chapa Base - Subconjunto Carcasa.*

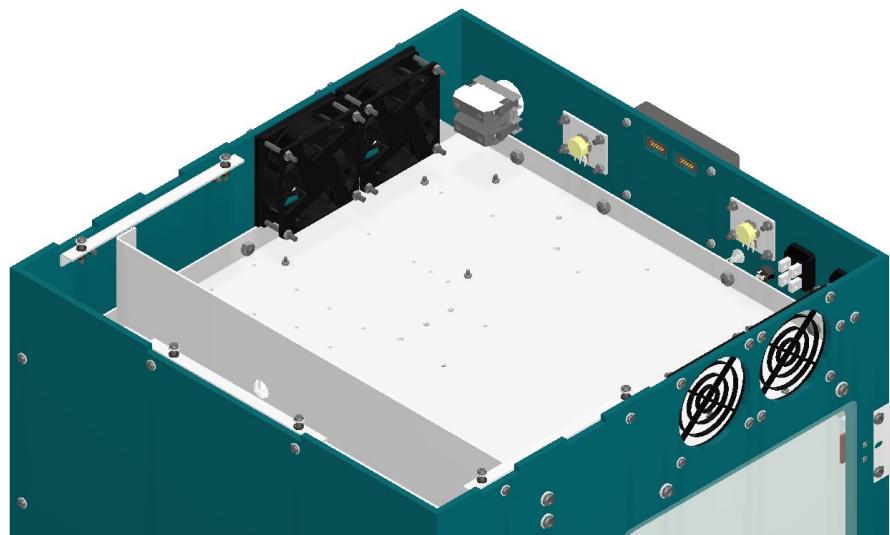
- **Estructura Interna:**

La estructura interna del Subconjunto está compuesta por el Cielo Raso, las Chapas para Potenciómetros, la Chapa de Refrigeración, el Chapon, el Soporte para Final de Carrera - Eje Y, su respectivo Final de Carrera, el Soporte LED y la Cadena Portacables.

El Cielo Raso se sujet a los Laterales y al Frente Carcasa mediante tornillos y es la base del área en donde se encuentran la gran mayoría de los componentes electrónicos de la máquina.

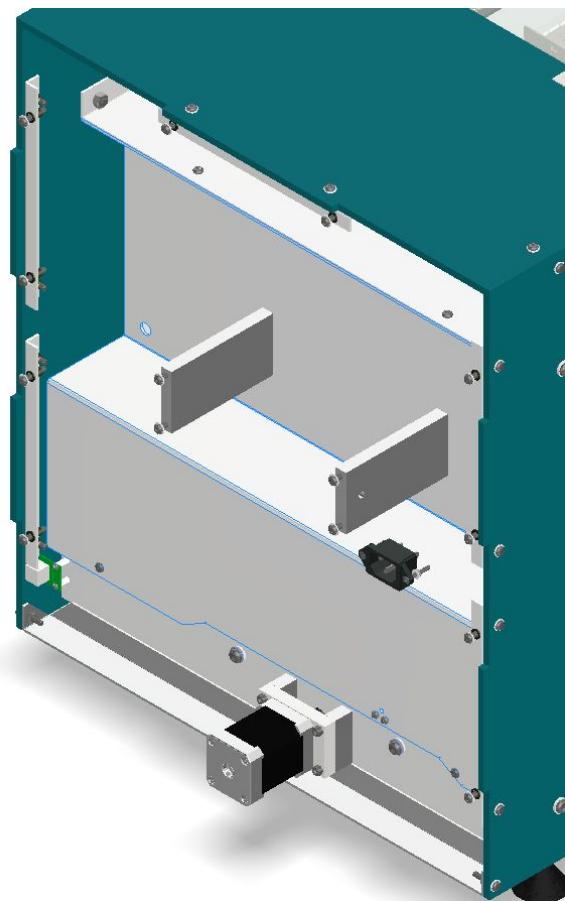
Sobre el Frente Carcasa podemos encontrar las Chapas para Potenciómetros cuyo objetivo es sujetar correctamente los potenciómetros de la botonera

Delimitando el área destinada a los componentes electrónicos podemos observar la Chapa de Refrigeración. Esta pieza se encuentra sujet a los Laterales Carcasa y tiene el objetivo de generar un flujo de aire adecuado para asegurar una correcta refrigeración de los componentes electrónicos.



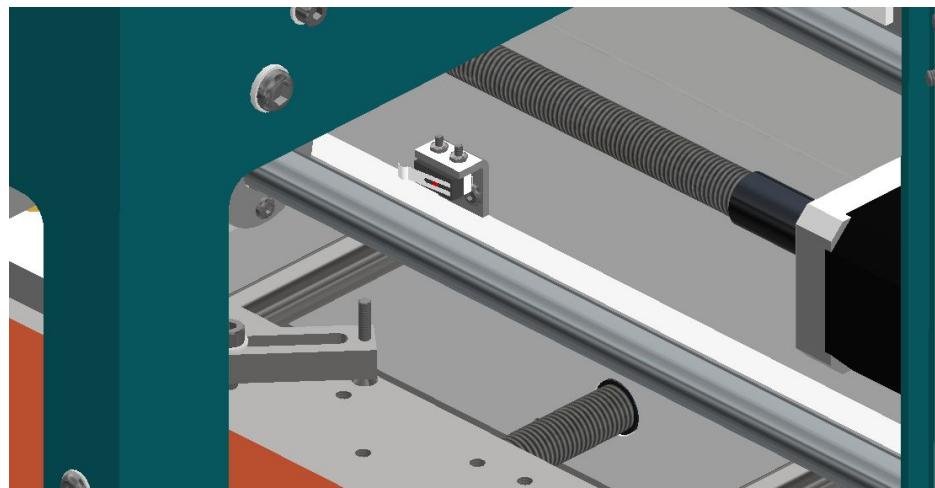
*Cielo raso, Chapas para Potenciómetros y Chapa de Refrigeración - Subconjunto Carcasa.*

El Chapon se sujeta mediante tornillos al Cielo raso y a la Culata y tiene el objetivo de generar un espacio para alojar cables como de Motores Paso a Paso, de Finales de Carrera, de alimentación, etc. Sujetos a esta pieza también se encuentran los antes mencionados Soportes para Culata Carcasa.



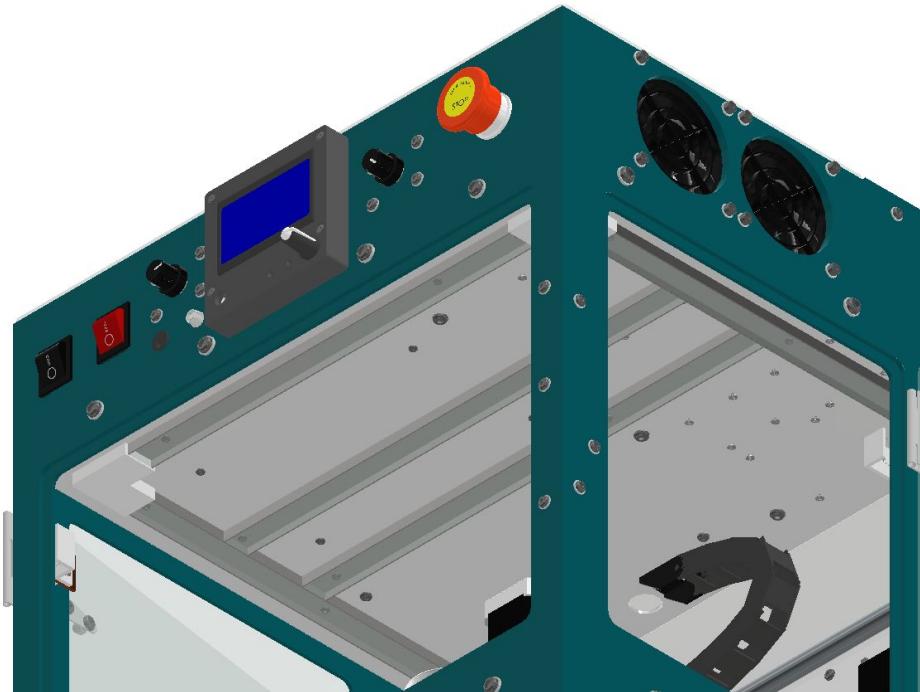
*Chapon destacado en azul junto con los Soportes para Culata Carcasa.*

En la cara interna del Chapon se encuentra fijado mediante tornillos el Soporte para Final de Carrera - Eje Y junto con su respectivo Final de Carrera.



*Soporte de Final de Carrera - Eje Y junto con su Final de Carrera colocados en la cara interna del Chapon.*

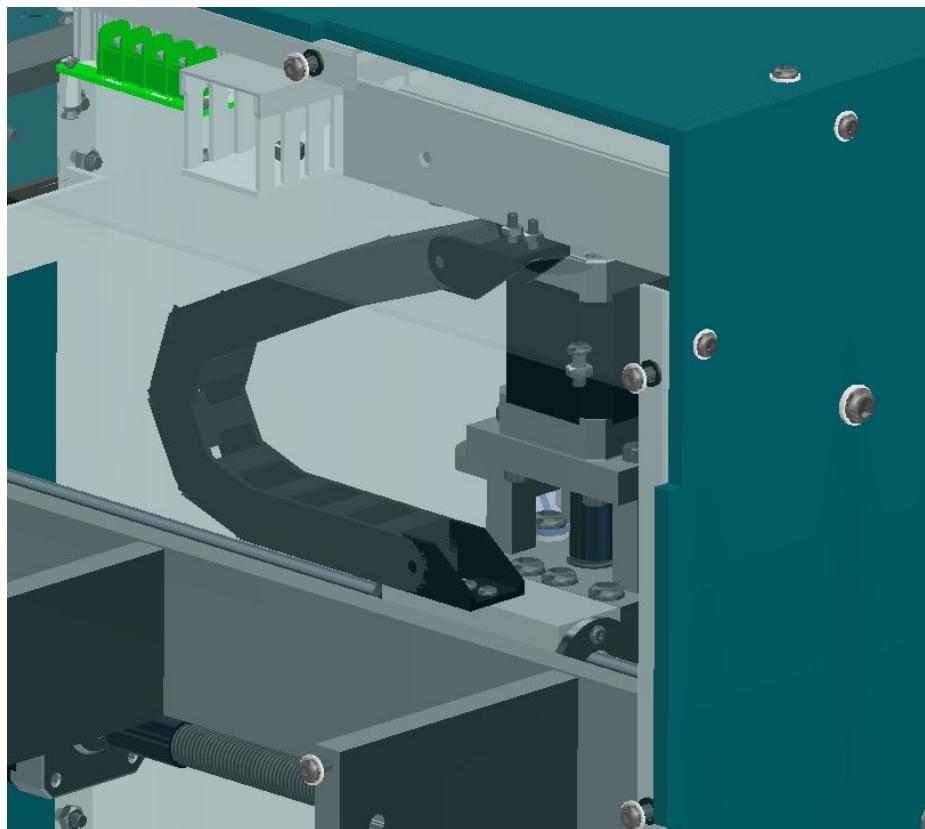
El Soporte LED tiene la finalidad de alojar la tira LED que iluminará el área de mecanizado de la máquina. Se sujeta mediante tornillos al Cielo raso y, para mejorar la terminación de sus esquinas, posee Acoples L en las mismas.



*Soporte LED junto con sus Acoples L - Subconjunto Carcasa.*

La Cadena Portacable tiene el objetivo de guiar y contener los cables del Motor DC, del Motor Paso a Paso - Eje Z y del Final de

Carrera - Eje Z hacia el Área Electrónica. Uno de sus extremos se encuentra sujeto al Cielo raso y el otro al Bloque de Vínculo Cabezal.

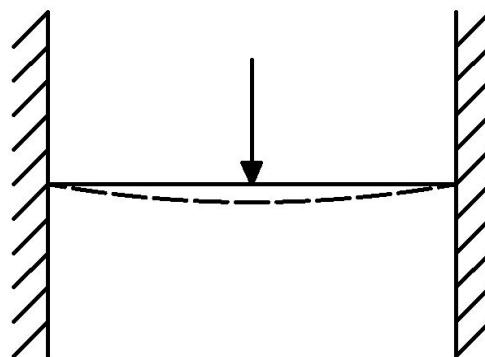


*Cadena Portacables - Subconjunto Carcasa.*

### 3.3. Cálculos Estructurales:

A medida que fuimos diseñando nuestra máquina, hemos realizado distintos cálculos relacionados con los esfuerzos mecánicos a los que están sometidos ciertas piezas críticas de la misma. A continuación, se plasmarán brevemente los cálculos de las Guías de Mesa y de las Guías de Vínculo Cabezal

Para estos cálculos, se considera que las guías se encuentran en un sistema de “Viga Dblemente Empotrada”, esquematizado en la siguiente figura:



*Esquema de Viga Dblemente Empotrada.*

Es importante destacar que, en pos de sobredimensionar los esfuerzos, se tomó la carga como puntal y, al mismo tiempo, se supuso que solo será soportada por una sola Guía.

- **Guías de Mesa:**

**Datos:**

- Masa: 3 kg (Extraída de Autodesk Inventor)
- Coeficiente de Seguridad: 2
- Largo: 43,4 cm
- Material: Acero SAE 1040
- Tensión de Rotura: 5600  $\frac{Kg}{cm^2}$
- Módulo de Elasticidad: 2100000  $\frac{Kg}{cm^2}$
- Flecha admisible: 0,01 cm

$$F = \frac{P \times C_s. \times (l)^3}{192 \times E \times J} \Rightarrow 0.01\text{cm} = \frac{3\text{ Kg} \times 2 \times (43,4\text{ cm})^3}{192 \times 2100000 \frac{Kg}{cm^2} \times \frac{\pi \times (d)^4}{64}} \Rightarrow \frac{\pi \times (d)^4}{64} = \frac{3\text{ Kg} \times 2 \times (43,4\text{ cm})^3}{0.01\text{cm} \times 192 \times 2100000 \frac{Kg}{cm^2}}$$

$$d = \sqrt[4]{\frac{3\text{ Kg} \times 2 \times (43,4\text{ cm})^3 \times 64}{\pi \times 0.01\text{cm} \times 192 \times 2100000 \frac{Kg}{cm^2}}} = 1,25\text{ cm}$$

El diámetro de las dos Guías de Mesa es de 12,7 mm.

- **Guías de Vínculo Cabezal:**

**Datos:**

- Masa: 4 kg (Extraída de Autodesk Inventor)
- Coeficiente de Seguridad: 2
- Largo: 43,4 cm
- Material: Acero SAE 1040
- Tensión de Rotura mínima: 5600  $\frac{Kg}{cm^2}$
- Módulo de Elasticidad: 2100000  $\frac{Kg}{cm^2}$
- Flecha admisible: 0,01 cm

$$F = \frac{P \times C_s. \times (l)^3}{192 \times E \times J} \Rightarrow 0.01\text{cm} = \frac{4\text{ Kg} \times 2 \times (43,4\text{ cm})^3}{192 \times 2100000 \frac{Kg}{cm^2} \times \frac{\pi \times (d)^4}{64}} \Rightarrow \frac{\pi \times (d)^4}{64} = \frac{4\text{ Kg} \times 2 \times (43,4\text{ cm})^3}{0.01\text{cm} \times 192 \times 2100000 \frac{Kg}{cm^2}}$$

$$d = \sqrt[4]{\frac{4\text{ Kg} \times 2 \times (43,4\text{ cm})^3 \times 64}{\pi \times 0.01\text{cm} \times 192 \times 2100000 \frac{Kg}{cm^2}}} = 1,35\text{ cm}$$

El diámetro de la dos Guía de Vínculo Cabezal es de 12,7 mm.

Con el objetivo disminuir los costos y facilitar las compras, las Guías de Soporte de Motor (que están sujetadas a esfuerzos mucho menores) también son de Ø12,7 mm.

#### 4. Diagramas de Operaciones y Planos:

Para la cuarta etapa de nuestra metodología de diseño fue necesario producir los Planos Mecánicos, teniendo en cuenta, como mencionamos, el Manual de Normas IRAM de Dibujo Tecnológico, edición 2009. Al mismo tiempo, realizamos Diagramas de Operaciones cuyo objetivo es describir la secuencia de acciones, mecanizados, operaciones y herramientas necesarias para la manufactura de las piezas. La unión de los Diagramas de Operaciones y los Planos Mecánicos otorga todos los datos necesarios para fabricar la totalidad de la máquina. Es por esto que su interpretación técnica resulta fundamental de cara a la manufactura. Es importante destacar que los Diagramas de Operaciones se realizaron en función de las máquinas y equipos disponibles en el taller del Instituto, es por esto que realizamos una planilla en donde se indica el porcentaje de utilización de cada máquina. Para su lectura, ver Anexo A: *Datos - Ensamble Ajuste Fino Total + Electrónica; Hoja: Planos y Máquinas a Utilizar.*

Nuestro diseño posee más de 90 planos mecánicos y 65 Diagramas de Operaciones. Para su consulta dirigirse al documento “Planos Mecánicos - G1” y “Diagramas de Operaciones - G1” respectivamente.

Con el objetivo aumentar la eficiencia y la organización a la hora de entregar tanto los planos mecánicos como los diagramas de operaciones, realizamos múltiples planillas que indican las distintas fechas de entrega de los mismos, sus códigos, sus revisiones, etc. Para su consulta, ver Anexo E: *Entregas - Ensamble Ajuste FIno Total + Electrónica.*

# Electrónica, Electricidad y Control

## **1. Primeras Investigaciones: Lógica y Control.**

En paralelo a las actividades relacionadas con el Diseño Mecánico, una vez definidas las exigencias por parte de la empresa, realizamos una investigación con el objetivo de determinar qué placa controladora utilizaremos. Nuestras opciones eran dos: Arduino o Raspberry Pi. Si bien la segunda opción tiene grandes ventajas sobre la primera pues posee procesadores con más alto nivel ARM y memorias caché más grandes, la implementación de Arduino en este tipo de proyectos es completamente viable y recomendable. Esta última información sumada a los conocimientos que ya teníamos en cuanto a la programación en Arduino nos sirvió para determinar la utilización del mismo.

Una vez seleccionada la placa controladora, nos enfocamos en determinar el tipo de Shield o Módulo a utilizar para controlar los Drivers de los Motores Paso a Paso y la máquina en general. Para la plataforma Arduino existen dos módulos: CNC Shield y RAMPS.

Si bien el CNC Shield se utiliza específicamente para nuestra aplicación, es decir, máquinas destinadas a realizar mecanizados, implica la selección de una placa controladora Arduino UNO. En cambio, la utilización RAMPS, implica la implementación de Arduino Mega 2560, versión de Arduino mucho más avanzada en comparación con Arduino UNO. Esta mayor capacidad del Arduino Mega otorga ciertas ventajas como la posibilidad de utilizar una pantalla controladora. Al mismo tiempo, RAMPS tiene la gran desventaja de enfocarse únicamente en el control de impresoras 3D, por lo que existen variables como extrusores, camas calientes, condiciones de temperatura y otras cuestiones no relacionadas con nuestra aplicación.

Además, la selección del CNC Shield implica la utilización del firmware GRBL. La selección de RAMPS, del firmware Marlin. Nuevamente, aquí existen diferencias en cuanto a la aplicación para la cual fueron concebidos. GRBL se enfoca en máquinas de mecanizado y Marlin en impresoras 3D.

Finalmente, por sus características y las recomendaciones por parte de Ingeniería Caamaño S.A., seleccionamos el Módulo RAMPS 1.4 junto con su Firmware Marlin.

## 2. Potencia y Alimentación:

En este apartado nos centraremos en la descripción de todos los componentes eléctricos, electrónicos y/o electromecánicos que, en conjunto, hacen a la potencia de la máquina.

### 2.1. Alimentación:

Durante la primeras investigaciones, a la hora de determinar la alimentación de la máquina, seleccionamos una fuente simple 12V con bornes de salida. Supusimos que con solo esta tensión y una adecuada corriente en función de la potencia requerida, la alimentación de la máquina estaba definida.

A la hora de seleccionar y comprar el Motor DC del Husillo<sup>8</sup> nos enfrentamos a un inconveniente pues el mismo posee una potencia máxima de 300W a 48V. De este modo, nos vimos obligados a descartar su alimentación mediante la salida D9 de RAMPS puesto que esos valores de potencia generaríaían la destrucción de la placa. Teniendo en cuenta esto, barajamos la posibilidad de utilizar un Boost Step-Up junto con el módulo MKS MOS. De esta manera, el Boost se encargaría de elevar la tensión de 12 a 48V requiriendo, a cambio, grandes valores de corriente y el MKS MOS se encargaría de manejar estos valores de corriente utilizando D9 únicamente como control PWM, evitando la destrucción del RAMPS.

Aunque esta solución era viable, los valores de consumo de corriente eran sumamente altos, alcanzando los 40A aproximadamente. Consultando con el equipo docente, llegamos a la conclusión que la mejor solución es utilizar dos fuentes por separado, una para control del Motor DC y otra para las placas controladoras, generando un sistema más eficiente. Es por esto que nuestra máquina posee dos fuentes de alimentación. Una de 48V; 7,5A destinada al control de Motor del Husillo mediante un Dimmer y otra de 12V; 12,5A para las placas de control y la tira LED.

Para poder conectar nuestra máquina a la red, existe una Ficha Interlock ubicada en la Culata Carcasa. El contacto a Tierra de todas las partes conductoras se realiza mediante un tornillo ubicado en el Soporte para Culata Carcasa Izquierdo, evitando posibles accidentes ocasionados por un eventual falso contacto. Es importante mencionar que las masas de todas las fuentes de corriente continua también se conectan a Tierra con el fin de disminuir ruidos e interferencias eléctricas y electromagnéticas.

---

<sup>8</sup>

Para más información dirigirse al siguiente apartado “Motor DC de Husillo”.

### 2.3. Motor DC de Husillo:

A la hora de mecanizar placas de cobre comúnmente existen dos opciones. Utilizar un Motor Dremel o un Motor de corriente continua de alta potencia. Consultando diferentes bibliografías notamos que la sujeción de los motores Dremel es el talón de aquiles de muchos Routers CNC. Con el objetivo de evitar este inconveniente, nos inclinamos por utilizar un Motor DC.

Para determinar la potencia y velocidad del motor seleccionado no es correcto basarse únicamente en los cálculos teóricos de velocidad de giro (RPM) pues los mismos dan como resultado un valor innecesariamente alto y técnicamente inalcanzable. Es por esto que consultamos con distintas empresas dedicadas al rubro, entre ellas, PúlsAr I+D. Estas consultas nos permitieron concluir que con velocidades cercanas a los 8000 rpm los resultados son sumamente aceptables e, incluso, competitivos en el mercado.

De este modo seleccionamos un Motor DC capaz de alcanzar velocidades de entre 6000 y 12000 rpm, con una tensión máxima de 48V y una potencia de 300W. El mismo posee 13 boquillas intercambiables de múltiples diámetros para sujetar distintas herramientas de corte.

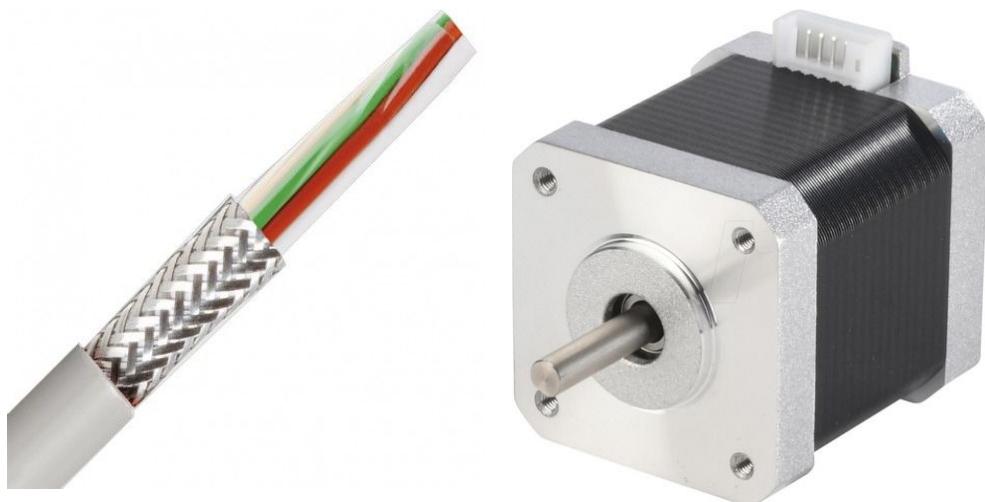


Motor DC de Husillo, 48V; 300W.

## 2.4. Motores Paso a Paso:

Para este tipo de proyectos es muy usual la utilización de Motores Paso a Paso de nomenclatura NEMA 17. Luego de comunicarnos nuevamente con PúlsAr y de realizar una investigación teniendo en cuenta las experiencias de proyectos similares al nuestro, seleccionamos el Motor Paso a Paso SC42STH60-1684, el cual posee un *Holding Torque* de 7,1 kg cm.

Con el objetivo de disminuir las interferencias electromagnéticas comúnmente denominadas “EMI” (*Electromagnetic Interference*) y, como consecuencia, perder la menor cantidad de pasos y aumentar la vida útil de los motores, hemos optado por utilizar cables apantallados multifilares. Estos poseen una jaula o malla protectora de metal a su alrededor y, para eliminar tanto los campos eléctricos como electromagnéticos, es necesario conectar ambos extremos de la misma a masa.



*Cable apantallado multifilar.*

*Motor Paso a Paso NEMA 17.*

### 2.4.1. Drivers:

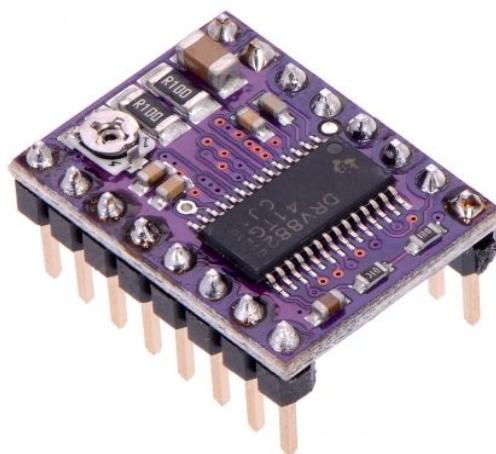
Comúnmente, en conjunto con RAMPS, se pueden utilizar dos tipos de Drivers para controlar los Motores Paso a Paso, A4988 y DRV8825, ambos del fabricante Pololu. Sus principales diferencias radican en la corriente máxima por fase continuos que son capaces de entregar a su Motor y en la cantidad de micropasos que son capaces de manejar.

El A4988 puede entregar 1A máximo por fase y ofrece un máximo de 16 micropasos. En cambio, el DRV8825 es capaz de entregar 1,5A por fase y posee un cantidad máxima de 32 micropasos.

Teniendo en cuenta los valores expuestos, optamos por utilizar los Drivers DRV8825. Los valores de corriente que maneja junto con su excelente cantidad de micropasos y la acción conjunta con los cables apantallados permiten: generar movimientos libres de

vibraciones, disminuir al mínimo la cantidad de interferencias y aumentar al máximo la precisión a la hora del mecanizado.

A fin de evitar posibles calentamientos en estos Drivers, colocaremos en cada uno de ellos disipadores sobre su microcontrolador.



DRV8825 - Drivers.

### 3. Área Electrónica<sup>9</sup>:

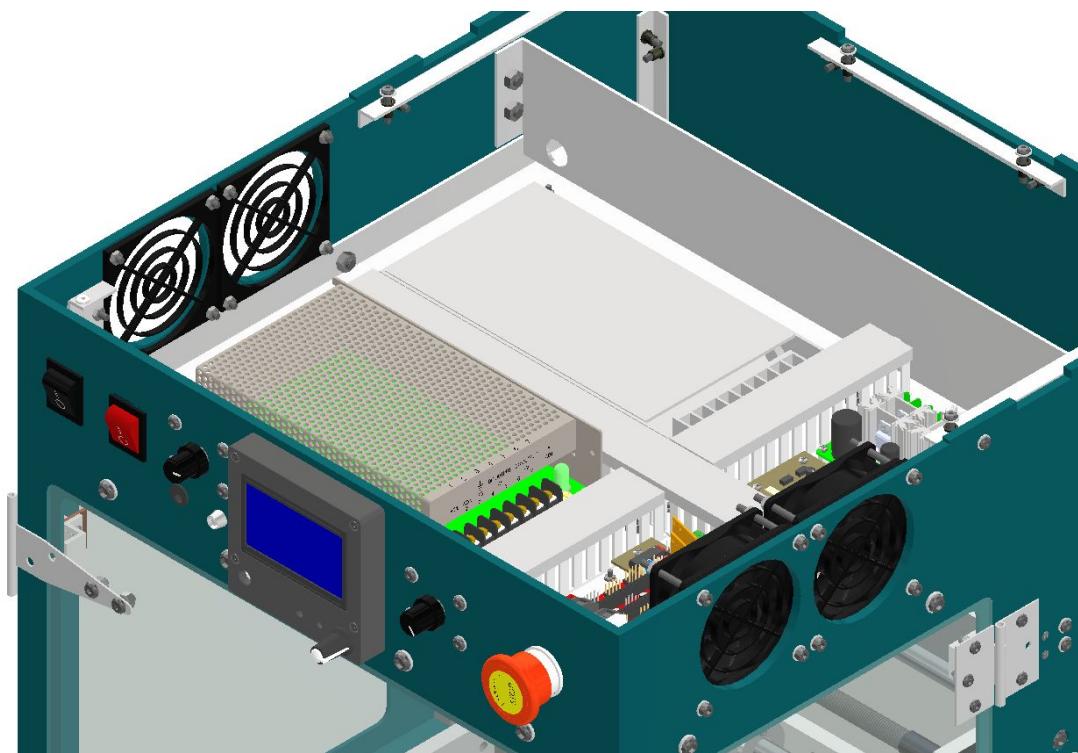
El Área Electrónica se encuentra sobre el Cielo raso y, como mencionamos, tiene el objetivo aislar los componentes electrónicos del área de mecanizado. Evitando eventuales ingresos de viruta, desechos, polvos, etc. Evidentemente, todas las Placas Electrónicas se sujetan mediante espaciadores roscados con el objetivo de asegurar una correcta aislación.

Para garantizar una refrigeración de todos los componentes electrónicos colocamos rejillas de ingreso de aire en el Lateral Izquierdo Carcasa y ventiladores extractores en el Lateral Derecho Carcasa. Al mismo tiempo, la Chapa de Refrigeración y los Cablecanales ranurados generan una adecuada canalización del aire, asegurando la correcta circulación del mismo.

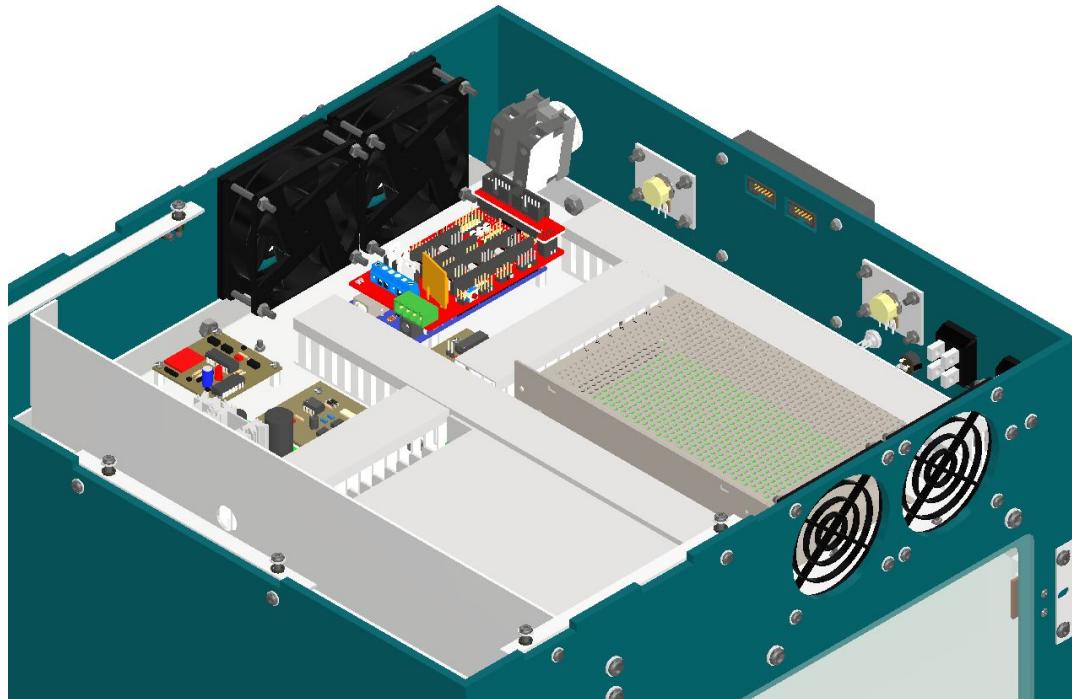
---

<sup>9</sup>

Para observar con mayor detalle el Área Electrónica, ver “Vista Superior y Vista Lateral Derecha Diagrama Topográfico”. G1-E-04 y G1-E-05, respectivamente.



Vista frontal en perspectiva - Área Electrónica.



Vista posterior del Área Electrónica.

### 3.1. Canalización de Cables:

Para canalizar los cables que permiten el funcionamiento de la máquina utilizamos Cablecanales ranurados. Su utilización junto con las cintas helicoidales tiene el objetivo de realizar conexiones lo más prolijas y ordenadas posible, facilitando su intervención ante posibles mantenimientos.

Para un correcto control y gestión de las conexiones existentes dentro del área electrónica, realizamos una planilla en donde se indica todos y cada uno de los cables utilizados, su longitud, su función, etc. Para su consulta, ver Anexo D: *Placas y Cables*.

### 3.2. Placas de Control:

Dentro del Área Electrónica se encuentran todas las placas que controlan nuestra máquina, estas son: Placa Dimmer para Tira LED, Placa Dimmer para Motor DC, Placa de Seguridad y, por último, Placa Inverter. A continuación describirán de forma detallada todas y cada una de ellas.

Nuevamente hemos realizado una planilla con el objetivo de tener un correcta dimensión y control de todos y cada uno de los componentes presentes en todas las Placas de Control. Para su consulta, ver Anexo D: *Placas y Cables*.

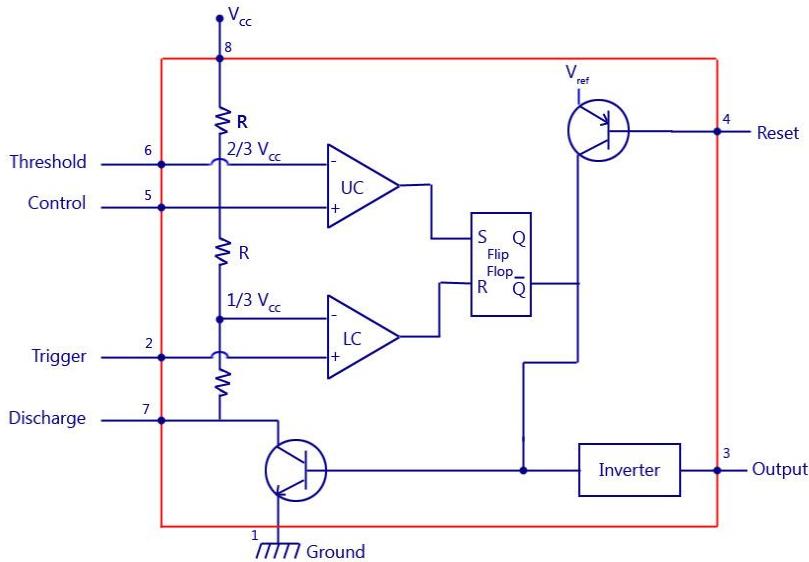
#### 3.2.1. Circuito Dimmer para Tira LED:

Dada la necesidad de iluminar el área de mecanizado para permitir que el operario pueda ver en todo momento el funcionamiento de la máquina, decidimos realizar la instalación de una tira LED colocada en los mencionados Soportes LED. Estos soportes no solo cumplen la función de alojar la tira LED sino que también actúan como difusor, asegurando una excelente iluminación del área en cuestión.

Para controlar el encendido y apagado de la Tira LED el operario debe accionar una llave simple SPDT ubicada en la Botonera y para regular su intensidad, debe girar el potenciómetro colocado para tal fin también ubicado en la Botonera.

Teniendo en cuenta el consumo de corriente de la Tira LED, decidimos diseñar un circuito Dimmer PWM para controlar su intensidad. El potenciómetro mencionado tiene como único objetivo variar el tiempo en el cual los LED's se encuentran prendidos, simulando una disminución en la intensidad lumínica de los mismos. El circuito funciona de la siguiente manera:

El componente crucial es el Circuito Integrado 555 que compara (mediante los amplificadores operacionales que posee en su interior) tercios de la tensión de entrada. De este modo y teniendo en cuenta que la tensión de alimentación es 12V, el Timer evalúa si en su pin Threshold existe más de  $\frac{2}{3}$  de la tensión de entrada, es decir, más de 8V y si en su pin Trigger existen menos de  $\frac{1}{3}$  de la tensión de entrada, es decir, menos de 4V.



*Circuito Interno, Integrado 555 (Timer) - Circuito Dimmer para Tira LED*

En un primer instante, el capacitor se encuentra descargado y la corriente de carga circula por un circuito RC compuesto por la resistencia de 1k, el valor resistivo del potenciómetro, el diodo D1 y, por supuesto, el Capacitor C1. Lógicamente, como el capacitor se encuentra descargado, la tensión en el comparador U2 es mayor en el pin + y, por lo tanto, existe un 1 en la salida del 555.

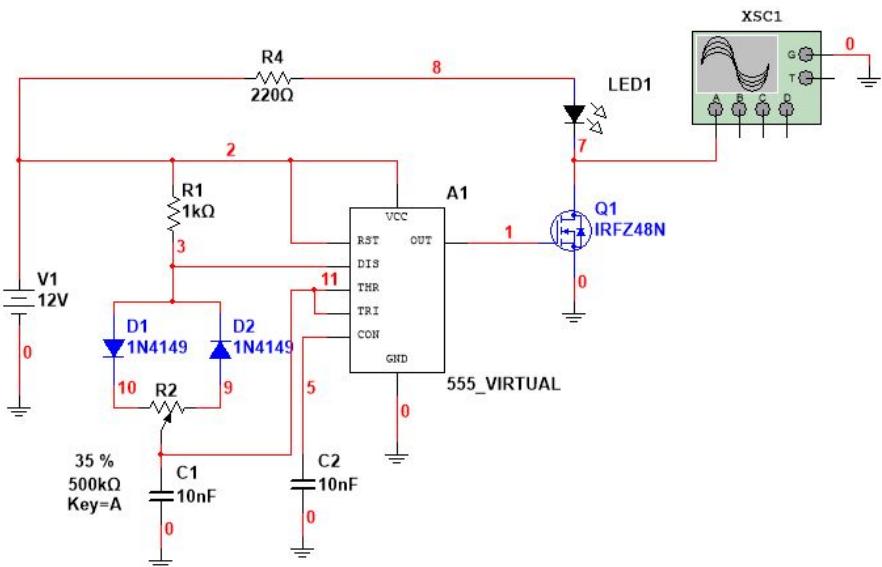
A medida que el capacitor se carga, la tensión en el pin + del comparador U1 aumenta hasta superar el valor existente en el pin -. Esto genera que en la salida Q exista un 0 y, por lo tanto, en la salida Q negada exista un 1. Esta última excita la base del transistor de carga interno del 555.

Vemos como dependiendo del valor resistivo del potenciómetro, la constante de tiempo del capacitor varía, es decir, varía el tiempo de carga y descarga. Como cada curva está asociada a un nivel lógico alto y a un nivel lógico bajo, dependiendo de la posición del potenciómetro se modula el ancho de pulso, obteniendo una señal PWM

El MOSFET Q1 (IRFZ48N) funciona como “llave On/Off” de la Tira LED soportando los valores de corriente de la misma.

Para hacer más estable el funcionamiento del circuito, colocamos el capacitor C2 en el pin Control del Integrado 555.

En la siguiente figura se puede apreciar la simulación realizada en el software NI Multisim. Cabe destacar que en la práctica se debe colocar una resistencia limitadora a la salida del Timer 555 para evitar su destrucción.



Simulación en Multisim - Circuito Dimmer para Tira LED

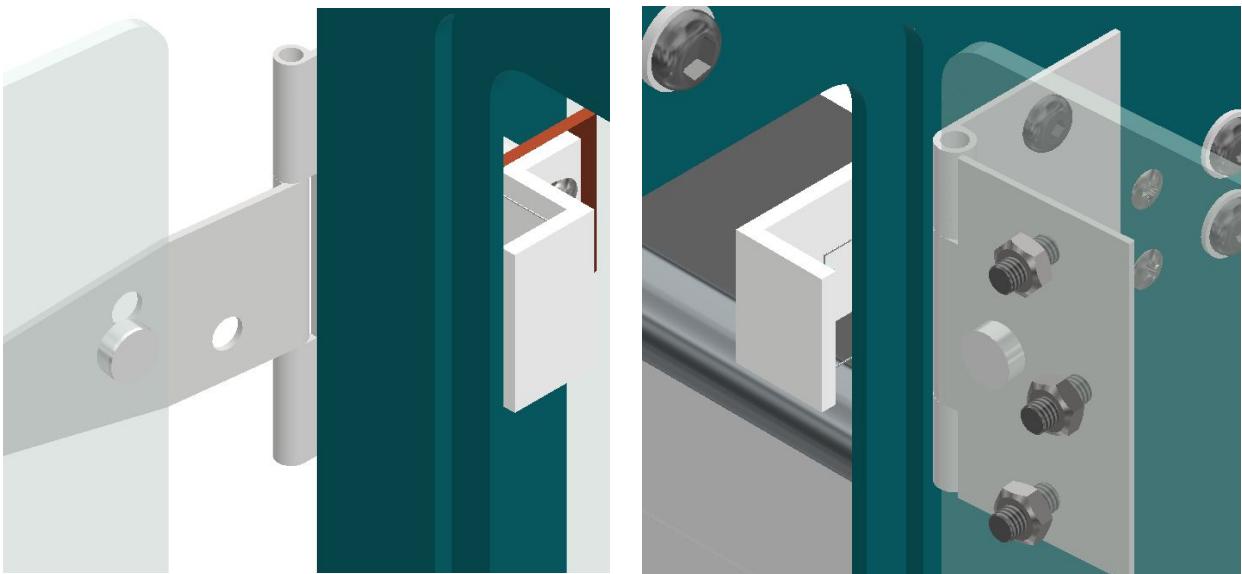
### 3.2.2. Circuito de Seguridad:

Este circuito tiene el objetivo de aumentar la seguridad operativa de la máquina. Su función es detener o evitar el arranque del Motor DC en el caso de que la Puerta Frontal y/o la Puerta Lateral Derecha estén abiertas. De este modo, se garantiza la seguridad del operario evitando eventuales accidentes en el momento de manipulación de la máquina. Si el circuito se activa, la máquina se detiene completamente y se muestra el siguiente mensaje en la Pantalla ubicada en la Botonera:

“¡Emergencia! Presionar el botón de Reset”

Este sistema es posible mediante la utilización de los anteriores mencionados imanes de neodimio colocados en las Puertas Frontales y Lateral Derecha y mediante la utilización de contactos por campo magnético denominados “Reed Switch”.

Los “Reed Switch” se encuentran normalmente en un estado “abierto” (no conduce) y, ante la presencia de un campo magnético, cambian de estado a “cerrado” (conduce). Para asegurar su integridad creamos, mediante diseño 3D, las “Carcasa para Reed Switch”. Estas piezas tienen la función de proteger tanto a los Reed Switch como a las placas sobre las que van soldados. En las siguientes imágenes se pueden observar las piezas mencionadas.



*Imán de neodimio junto con el Reed Switch, su Placa y su Carcasa ubicados en la Puerta Frontal.*

*Imán de neodimio junto con el Reed Switch, su Placa y su Carcasa ubicados en la Puerta Lateral Derecha.*

El funcionamiento de este circuito se resumen en la siguiente tabla de verdad:

- Entrada A: Reed Switch Normalmente Abierto
- Entrada B: Motor DC
- Salida: Accionamiento del Sistema de Emergencia

Entrada A	Entrada B	Salida ( $A \cdot B$ )
0	0	0
0	1	0
1	0	0
1	1	1

*Tabla de verdad - Circuito de Seguridad.*

- Caso  $A=0; B=0 \Rightarrow$  Puertas colocadas y Motor Apagado;
- Caso  $A=0; B=1 \Rightarrow$  Puertas colocadas y Motor Encendido (Funcionamiento Normal);
- Caso  $A=1; B=0 \Rightarrow$  Puertas no colocadas y Motor Apagado;
- Caso  $A=1; B=1 \Rightarrow$  Puertas no colocadas y Motor Encendido (¡Emergencia!).

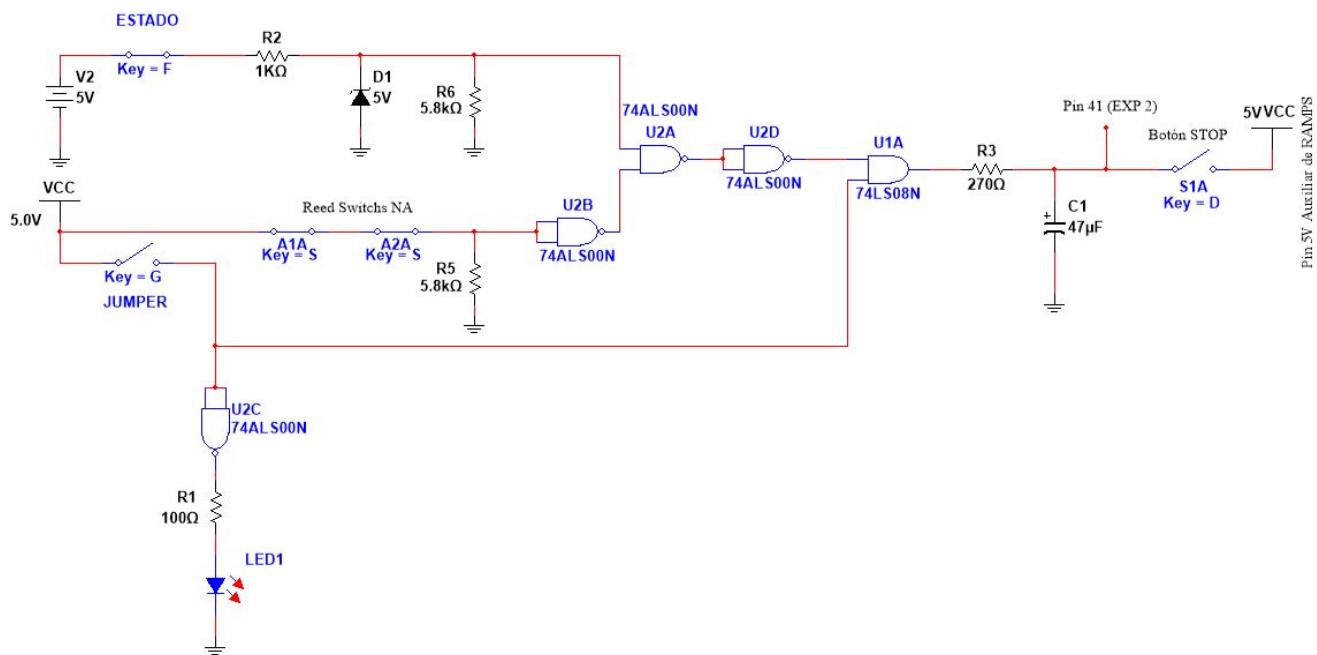
Supongamos el caso en el que las Puertas se encuentran abiertas (0 lógico en la Entrada A) y el Motor DC encendido (1 lógico en la Entrada B). La compuerta U2B envía un 1 a la AND formada por las dos compuertas U2A y U2D. Al mismo tiempo, el Motor encendido

envía un 1 a la otra entrada de la compuerta AND formada por las NANDs mencionadas. Siempre y cuando el Jumper se encuentre conectado, la compuerta AND U1A enviará un 1 al pin Reset D41, generando que se muestre el mensaje mencionado en la Pantalla y deteniendo toda la máquina hasta que se reinicie RAMPS.

En el caso de que las Puertas se encuentren cerradas (1 lógico en la entrada A), la compuerta NAND U2B tendrá como salida un 0 lógico con lo cual, sin importar el estado del Motor DC (Entrada B), no se enviará un 1 lógico al pin Reset D41.

Es importante destacar que el circuito posee un Jumper para su desactivación. En el caso de necesitar realizar algún tipo de mantenimiento que requiera la eliminación del circuito de seguridad, simplemente se debe quitar el Jumper. Su desconexión provocará que se encienda un diodo LED rojo de alta luminosidad indicando que se está realizando una operación riesgosa.

A continuación se puede observar el presente circuito:



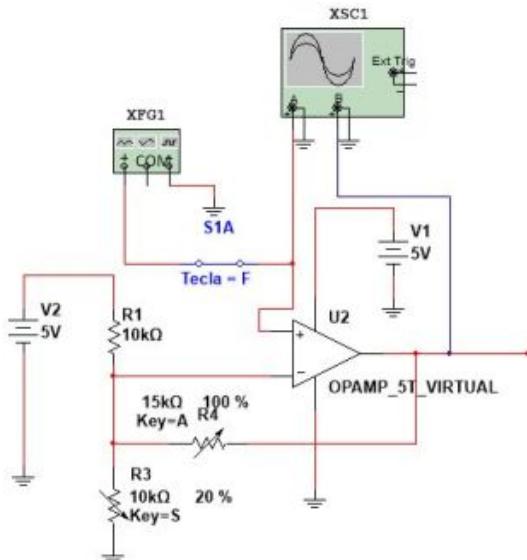
Simulación en Multisim - Circuito de Seguridad.

#### - Schmitt Trigger y Diodo Zener:

Al diseñar el circuito, supusimos que iba a ser posible la utilización de D9 como Entrada B. Evidentemente, esto no es viable dado que la misma emite una señal PWM de 0 a 12V, por lo que, si el nivel de señal fuese menor a un 1 lógico el circuito no funcionaría o, también, si el valor de la señal fuera mayor a 7V el circuito se destruiría.

Nuestra primera solución a este problema fue diseñar un circuito capaz de entregar un 1 lógico siempre que exista más

de 1V en su entrada. Este circuito se puede ver en la siguiente figura:



*Simulación en Multisim - Circuito Schmitt Trigger.*

La implementación de este circuito poseía como desventaja que su correcto ajuste y calibración gastaría nuestro recurso más escaso, el tiempo. Es por esto que, luego de una extensa investigación, optamos por utilizar un diodo Zener para fijar el valor de tensión en 5V. Este tipo de diodo tiene la particularidad de permitir la circulación de corriente a través de él al llegar a un determinado valor de tensión, con lo cual, constituye la solución solicitada.

Utilizando este circuito, siempre que el valor de tensión sea mayor a 5V, todo lo que esté conectado en paralelo a él recibirá 5V. Conectando nuestras compuertas NAND de esa manera impedimos la destrucción de las mismas ante posibles niveles de tensión altos provenientes de D9.

### 3.2.3. Circuito Inverter - Parada de Emergencia:

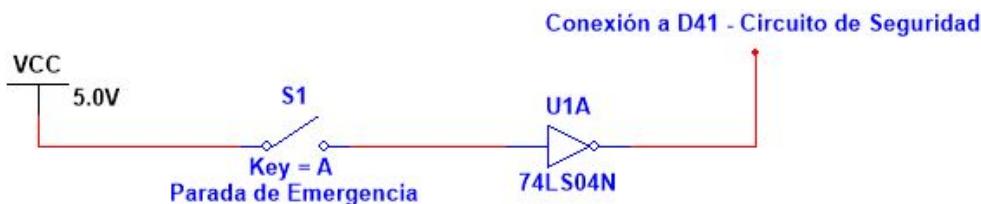
Entendemos que es fundamental la colocación de una Parada de Emergencia en este tipo de máquinas. Puesto que, como mencionamos, RAMPS ya posee un pin dedicado a detener la máquina en caso de emergencia, decidimos conectar la Parada de Emergencia al mismo.

Puesto que la gran mayoría de las Paradas de Emergencias disponibles en el mercado son NC, es decir, normalmente cerradas, diseñamos un circuito cuya función es invertir la señal y, de ese modo, excitar el pin mencionado.

Es así como presentamos nuestro Circuito Inverter compuesto por una compuerta inversora o NOT 7404. De este modo, al presionar la Parada de Emergencia, la compuerta recibirá un 0 lógico y enviará un 1 lógico al pin 41, deteniendo la máquina y mostrando, el siguiente mensaje en la Pantalla:

“¡Emergencia! Presionar el botón de Reset”

A continuación se puede ver el Circuito Inverter.



*Simulación en Multisim - Circuito Inverter.*

### 3.2.4. Regulador de Velocidad del Motor del Husillo:

En un principio la regulación de la velocidad del Motor DC se realizaría mediante control PWM, utilizando el módulo MKS MOS. Consultando la hoja de datos del mismo, notamos que esto no era posible debido a que su optoacoplador no soportaba 48V (tensión de funcionamiento del Motor).

Ante la dificultad, barajamos dos posibles soluciones. La primera era manufacturar un circuito MKS propio, ajustando los componentes a los valores requeridos. La segunda, comprar un módulo regulador de velocidad.

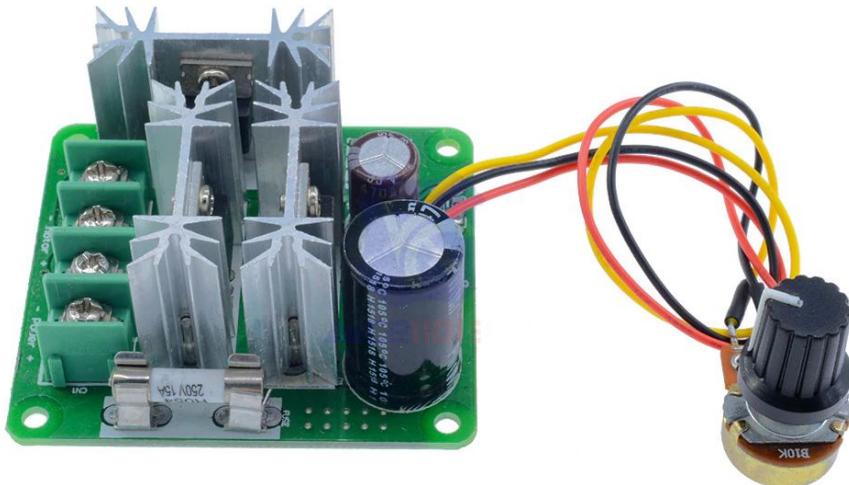
Teniendo en cuenta, nuevamente, que el tiempo es nuestro recursos más escaso, optamos por la segunda opción.

De este modo, la velocidad del Motor del Husillo se controla mediante un potenciómetro ubicado en la Botonera. El módulo seleccionado posee dos borneras de alimentación conectadas a 48V y dos borneras de salida conectadas, evidentemente, al Motor DC.

Para permitir el uso del Motor desde la aplicación de Control CNC GCode Controller<sup>10</sup>, colocamos un relé que tiene conectado en su bobina D9 y su llave conmutadora está conectada en serie con la alimentación del Motor DC. De este modo, hasta que no exista tensión en D9 no se podrá ni encender ni variar la velocidad del Motor.

<sup>10</sup>

Para más información dirigirse al apartado “CNC GCode Controller” situado en “Software y Firmware”

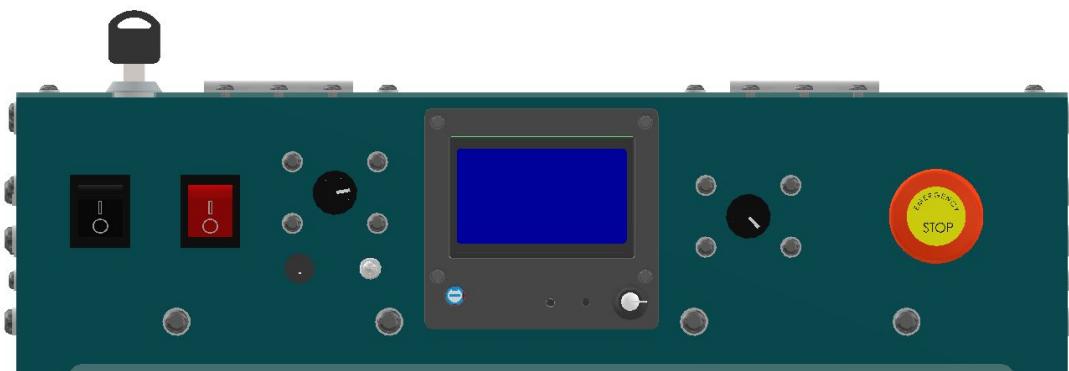


*Regulador de Velocidad del Motor del Husillo.*

Es importante destacar que todas las conexiones tanto eléctricas como electrónicas se encuentran plasmadas en un Diagrama de Bloques General y en un Diagrama de Electrónico (G1-E-01 y G1-E-02, respectivamente). Para su consulta dirigirse al documento Diagramas Eléctricos y Electrónicos.

#### 4. **Botonera<sup>11</sup>:**

Montada sobre el Frente Carcasa se encuentra la Botonera. La misma tiene el objetivo de permitir y facilitar el control de ciertos parámetros fundamentales para la utilización de la máquina. Al mismo tiempo, su conveniente ubicación facilita la conexión a todos los componentes electrónicos situados en el Área Electrónica.



*Botonera - Electrónica, Electricidad y Control*

Los componentes ubicados en la Botonera son: Tecla On/Off de la Máquina, Tecla On/Off de las Tira LED, Potenciómetro Dimmer para Tira LED, Jack 2,5 mm, Porta LED, Display Controlador, Potenciómetro Dimmer para Husillo y, por último Parada de Emergencia.

<sup>11</sup> Para observar con mayor detalle la Botonera, ver “Detalle Botonera Diagrama Topográfico”. G1-E- 06.

Nótese como la Botonera se encuentra distribuida de tal forma que aquellos elementos destinados al control de la máquina se encuentran a la izquierda y aquellos elementos destinados a la potencia de la misma a la derecha.

#### 4.1. Jack y Porta LED:

Estos elementos son necesarios para realizar la Auto-nivelación previa al mecanizado<sup>12</sup>. Para la misma debemos conectar las pinzas cocodrilo al Jack ubicado en la Botonera. Por otro lado, el Porta LED tiene la función de alojar el LED que se prenderá al hacer contacto, indicando que el correcto funcionamiento de las pinzas cocodrilo.

#### 4.2. Pantalla Controladora LCD Full Graphic Smart Controller:

La utilización de esta pantalla nos permite controlar múltiples aspectos del funcionamiento de la máquina, desde movimientos milimétricos en cada eje hasta enviar un G-Code mediante la inserción de una tarjeta SD. Como se puede ver en la imagen de la Botonera, esta pantalla posee una carcasa realizada mediante impresión 3D, el objetivo de la misma es evitar daños ocasionados por eventuales golpes.

Al mismo tiempo, este módulo posee un encoder rotativo con un botón para seleccionar, variar y controlar el mismo. Posee un buffer que emite sonidos al momento de presionar el botón y un pequeño botón que funciona como parada de emergencia.

Su conexión a RAMPS se realiza mediante dos cables planos de diez vías cada uno y un módulo adaptador que proporciona dos conectores IDC macho. En la siguiente imagen se puede observar la Pantalla Controladora junto con sus cables y el módulo mencionado.



*Pantalla Controladora junto con sus cables y su módulo adaptador.*

<sup>12</sup>

Para más información dirigirse al apartado “Utilización del Software” situado en “CNC GCode Controller”.

## 5. Softwares y Firmware:

La utilización de Softwares de Control y de diseño CAD resulta fundamental a la hora de utilizar nuestra máquina, pues los mismos son necesarios tanto para realizar los diseños de las placas PCB como para generar los G-Code y enviarlos. Para esto, recomendamos la utilización de tres programas:

- Software de Diseño de PCB (KiCAD, Autodesk Eagle, Altium, etc.);
- FlatCAM;
- CNC GCode Controller.

A grandes rasgos, el proceso de creación y envío de G-Code es el siguiente. Inicialmente se debe diseñar el PCB en un programa CAD, nosotros utilizamos Autodesk Eagle. A continuación se debe exportar la capa de cobre en formato Gerber (extensión “.grb”) para obtener toda la información de cada una de las capas de nuestro diseño, coordenadas, espesores, etc. y exportar los agujeros en formato Excellon (extensión “.xln”). Una vez obtenidos estos archivos, se deben cargar los mismo en FlatCAM para generar los vectores y crear el G-Code.

Por último, solo resta cargar el G-Code con el objetivo de controlar la máquina. Para esto debemos conectar nuestra computadora a la misma mediante USB-B y utilizar la aplicación CNC GCode Controller.

Si bien existe la posibilidad de enviar el G-Code mediante una tarjeta SD colocada en la Pantalla Controladora, no recomendamos esta opción puesto que las limitaciones respecto a la utilización de la aplicación son abismales, existiendo la posibilidad de obtener malos resultados a la hora del mecanizado.

A continuación se describen las funciones más importantes de los programas mencionados junto con los parámetros más relevantes a definir.

### 5.1. FlatCAM:

FlatCAM es una aplicación que nos permite vectorizar los diseños de PCB's y generar el archivo G-Code. Esto lo conseguiremos realizando los siguientes pasos.

Comenzaremos definiendo cada parámetro a configurar:

#### 5.1.1. Parámetros a definir para el agujereado de pads:

- **Cut Z:** Profundidad de corte de la herramienta;
- **Travel Z:** Distancia en el eje Z en la cual se desplazará la herramienta cuando no esté mecanizando;
- **Tool Change:** Se utiliza en caso de tener que cambiar la herramienta debido a los diferentes diámetros de los agujeros;
- **Tool Change Z:** Altura a la cual se realizará el cambio de herramienta;

- **End move Z:** Altura de la herramienta cuando termina el mecanizado (se debe tener en cuenta el recorrido máximo en el eje Z);
- **Feedrates:** Velocidad de avance en mm/min;
- **Spindle Speed:** Velocidad del husillo en RPM;
- **Dwell:** Tiempo (en segundos) necesario para que el husillo alcance la velocidad establecida;
- **Drill Tool dia:** Diámetro de la herramienta de agujereado.

#### 5.1.2. Parámetros a definir para el Enrutamiento de Pistas:

- **Tool dia:** Diámetro de la herramienta. Para definir este parámetro es común la utilización de la calculadora de FlatCAM;
- **Passes:** Número de pasadasca esta casilla, se genera un único archivo “\_iso” que combina todas las pasadas. Caso contrario, se generará un archivo ”\_iso” por cada pasada;
- **Bounding Box:** Genera la geometría de la placa;
- **Cutout Tool:** Función que sirve para cortar la placa;
- **Boundary Margin:** Distancia de los márgenes para el corte total del cuerpo generado en el "Bounding Box";
- **Rounded Geo:** Se marca la casilla en caso de que las esquinas sean curvas. El radio de la curva es igual a la distancia de los márgenes.
- **Cut Z:** Profundidad de corte;
- **Multi-Depth:** Se marca la casilla si queremos que en cada pasada haya una profundidad de corte diferente;
- **Travel Z:** Distancia en el eje Z en la cual se desplazará la herramienta cuando no esté mecanizando;
- **Tool Change:** Se utiliza en caso de tener que cambiar la herramienta;
- **Tool Change Z:** Altura a la cual se realizará el cambio de herramienta

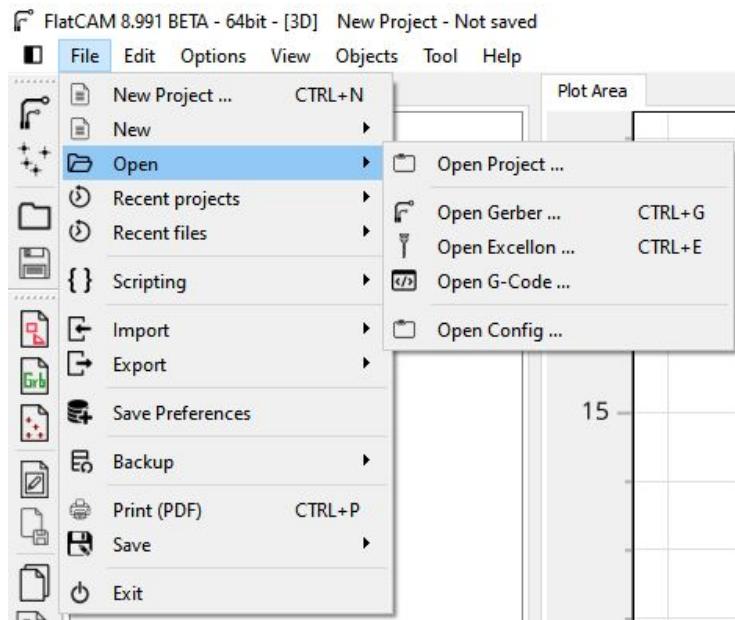
- **End move Z:** Altura de la herramienta cuando termina el mecanizado (se debe tener en cuenta el recorrido máximo en el eje Z);
- **Feedrates:** Velocidad de avance en mm/min;
- **Spindle Speed:** Velocidad del husillo en RPM;
- **Dwell:** Tiempo (en segundos) necesario para que el husillo alcance la velocidad establecida;

#### 5.1.3. Parámetros a definir para el Corte de la Placa de Cobre:

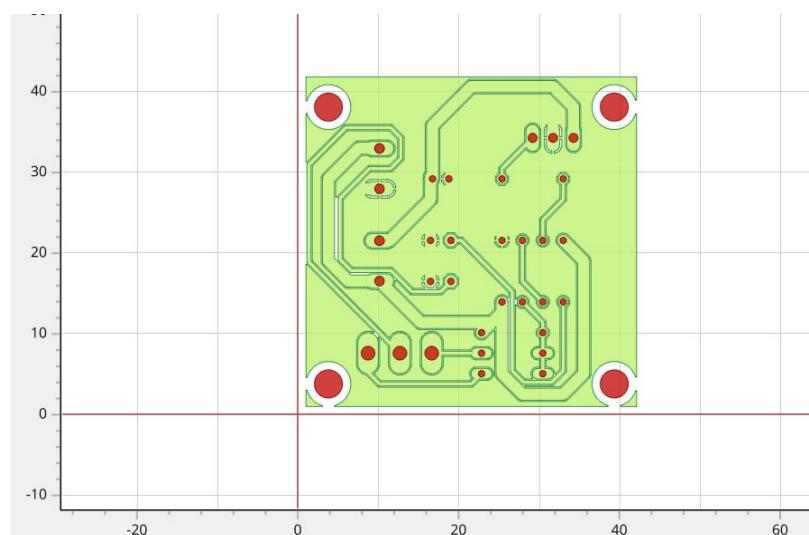
- **Tool dia:** Diámetro de la herramienta. Para definir este parámetro es común la utilización de la calculadora de FlatCAM;
- **Cut Z:** Profundidad de corte;
- **Multi-Depth:** Se marca la casilla si queremos que en cada pasada haya una profundidad de corte diferente;
- **Margin:** Se determina si el corte se realiza justo en el borde de la geometría de la placa. Caso contrario, para valores positivos se realizará más lejos y para valores negativos más cerca;
- **Gap size:** Se determina la realización y el tamaño de “puentes” al momento de mecanizar el corte;
- **Gaps:** Se determina la cantidad y la ubicación de los “puentes” mencionados;

#### 5.1.4. Utilización del Software

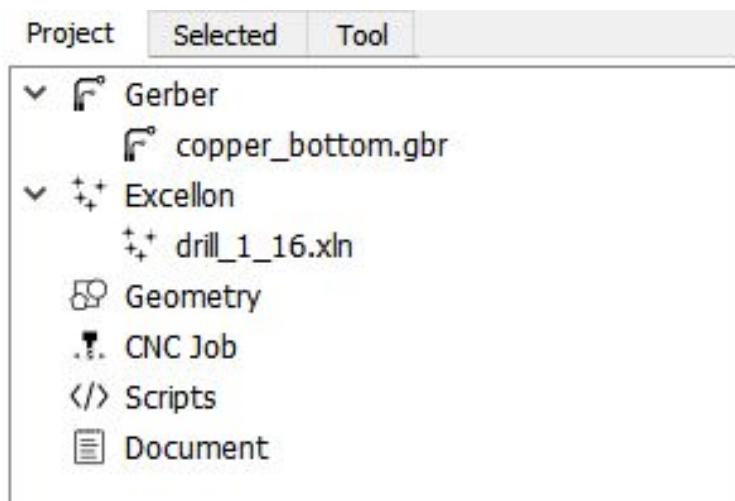
- ##### 5.1.4.1. Se realiza la carga del archivo Gerber previamente realizado en nuestro Software CAD. Se debe seleccionar la pestaña “File”, luego “Open” y, finalmente, “Open Gerber”. Este último paso también abrirá el archivo Excellon.



Una vez cargados los archivos .gbr y .xln. Se previsualiza la placa PCB en la grilla derecha. Es importante destacar que la esquina inferior izquierda de la PCB debe coincidir con las coordenadas de origen de FlatCAM. Esto se logra haciendo lo mismo en el Software de diseño CAD antes utilizado.

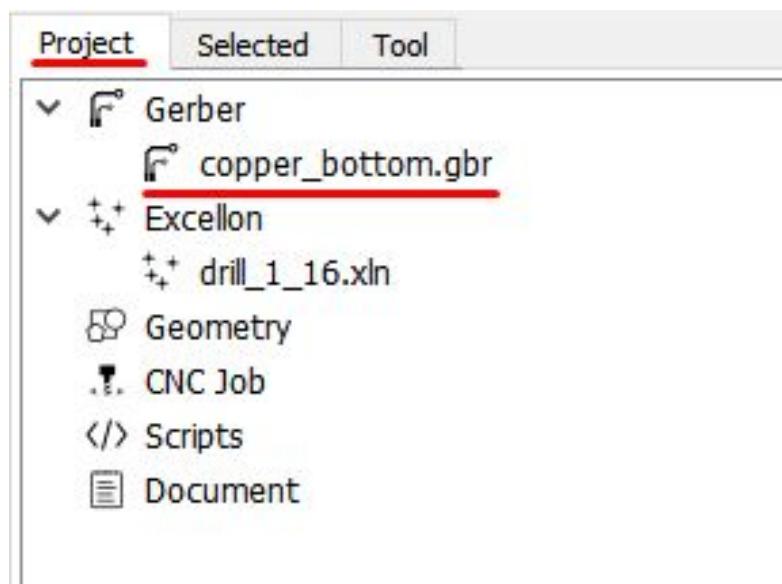


A la izquierda se encuentra el panel de navegación mediante el cual se seleccionan los distintos parámetros a configurar.



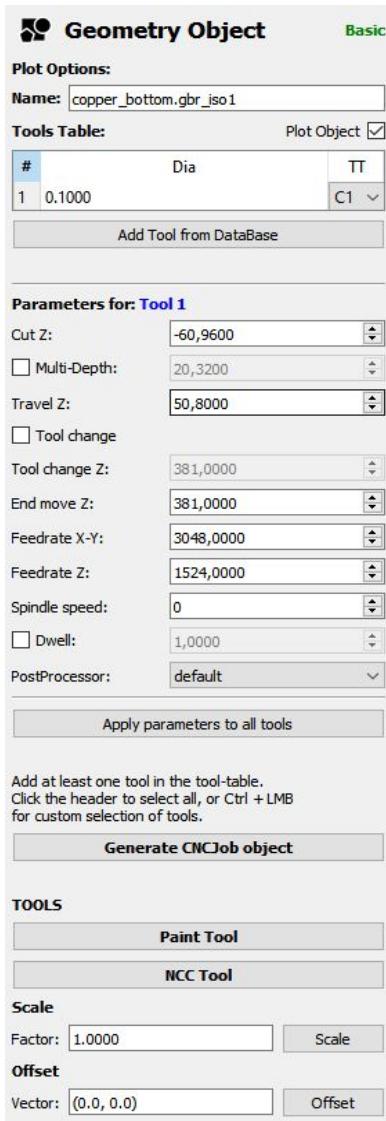
Pestaña “Project” dentro del panel de navegación - FlatCAM.

- 5.1.4.2. Ahora se debe realizar el enrutamiento de las pistas. Para ello se debe seleccionar el archivo Gerber en la pestaña “Project” y luego entrar en la pestaña “selected”. A continuación, se debe indicar la cantidad de pasadas y el diámetro de la herramienta. Por último, seleccionamos la opción “Generate Isolation Geometry” para que la aplicación trace los caminos que hará la herramienta a la hora de enrutar las pistas.





- 5.1.4.3. Luego de generar la geometría del enrutamiento, se desplegarán en el panel de navegación una serie de parámetros a configurar. Luego de ello, seleccionamos “Generate CNCJob object”.



De este modo, se genera el G-Code. Existe la posibilidad de visualizarlo en FlatCAM o de guardarlo.

- 5.1.4.4. Una vez realizado el enrutamiento de las pistas, debemos generar el taladrado de los agujeros para el montaje de los componentes. Se debe seleccionar el archivo Excellon en la pestaña “*Project*” y luego dirigirse a la pestaña “*Selected*”. Aquí es donde se observará una lista de todos los agujeros de nuestro PCB junto con sus diámetros

**Excellon Object** Basic

**Plot Options:**  Solid

**Name:**

**Tools Table:**  Plot

#	Diameter	Drills	Slots	P
6	0.8130	22		<input checked="" type="checkbox"/>
5	1.1180	3		<input checked="" type="checkbox"/>
4	1.1940	4		<input checked="" type="checkbox"/>
3	1.7000	3		<input checked="" type="checkbox"/>
2	3.3020	1		<input checked="" type="checkbox"/>
1	3.5000	4		<input checked="" type="checkbox"/>
<b>Total Drills</b>		<b>37</b>		
<b>Total Slots</b>		<b>0</b>		

En esta pestaña se deben configurar todos los parámetros solicitados. Luego, en el apartado “*Mill Holes*”, se debe introducir el diámetro de la herramienta a utilizar. Haciendo clic en “*Mill Drills Geo*” se generará el trazado del camino que recorrerá la herramienta.

**Create CNC Job**

Cut Z:

Travel Z:

Tool change

Tool change Z:

End move Z:

Feedrate Z:

Spindle speed:

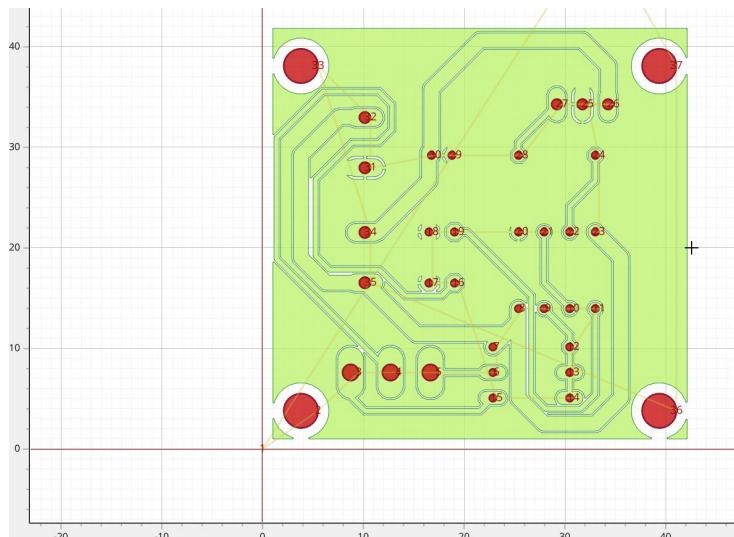
Dwell:

Preprocessor:

**Create Drills GCode**

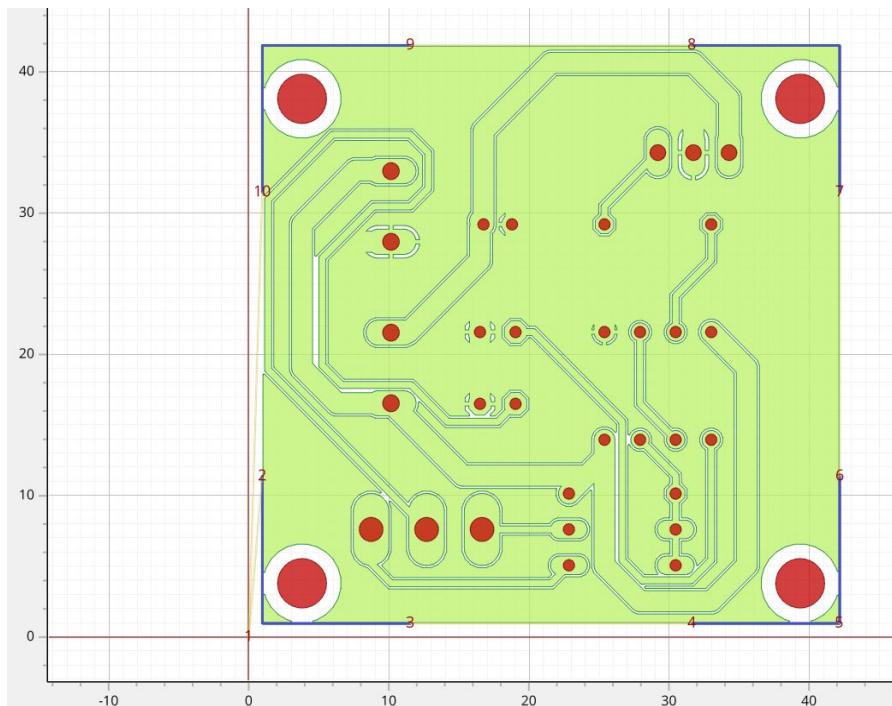
**Mill Holes**

Drill Tool dia:



Aquí se puede observar el recorrido que hará la herramienta para taladrar.

- 5.1.4.5. Luego de generar el recorrido de la herramienta, debemos realizar el archivo “CNC Job” haciendo clic en “Generate CNCJob object”
- 5.1.4.6. Finalmente, solo resta, guardar el G-Code generado.
- 5.1.4.7. Por último, debemos generar el G-Code del corte de la placa PCB. Para esto, se debe volver a seleccionar el archivo Gerber y hacer clic en “Cutout Tool” en la pestaña “Selected”
- 5.1.4.8. Se deben completar los parámetros que se muestra en la pantalla y luego hacer clic en “Generate Rectangular Geometry”. Al realizar esto, se creará un archivo “\_cutout” en la pestaña “Project”
- 5.1.4.9. Al seleccionar este nuevo archivo y hacer clic en “Generate CNCJob object” dentro de la pestaña “Selected” obtendremos el G-Code necesario para el corte.



*Las partes azules representan el movimiento de la herramienta al momento de hacer el corte. Nótese que la misma no recorre todo el borde de la placa, estas porciones no mecanizadas son los antes mencionados puentes.*

5.1.4.10. Por último pero no por ello menos importante, solo resta guardar el archivo G-Code recién generado.

## 5.2. CNC GCode Controller:

Esta aplicación la utilizamos como medio para comunicarnos con nuestra máquina. Básicamente, el software envía comandos y coordenadas que posteriormente la máquina interpretará y traducirá en movimientos en cada uno de los ejes, llevando a cabo el enrutamiento, los agujereados y el corte de la placa PCB. Resulta interesante destacar que este software fue creado únicamente para la adaptación de Marlin a este tipo de aplicaciones.

A continuación se plasmarán una serie de instrucciones necesarias para la utilización de este software, incluyendo recomendaciones útiles para nuestra aplicación.

Su interfaz se divide en siete pestañas, de las cuales seis serán de suma importancia.

- **Simple Controls:**

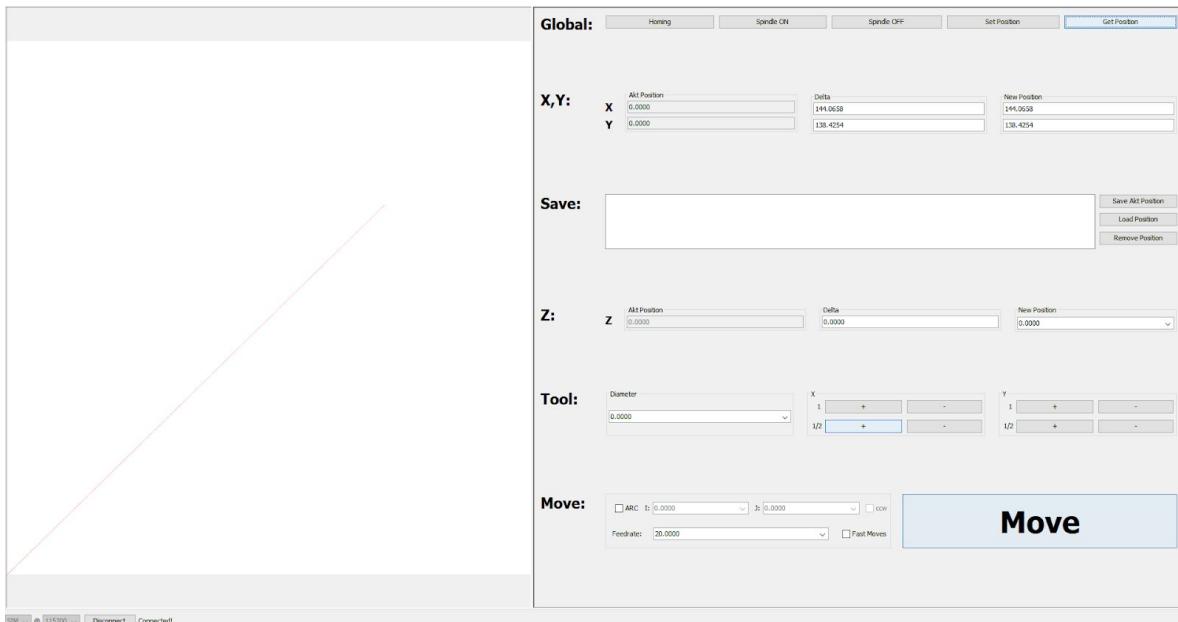
Esta pestaña tiene el objetivo de permitir el control de movimientos básicos de la máquina, como mover la herramienta en los tres ejes, realizar el homing (tanto de un eje como de los tres), prender y apagar el husillo e iniciar nuestro G-Code inicial.



Pestaña Simple Controls - CNC GCode Controller.

#### - Advanced Controls:

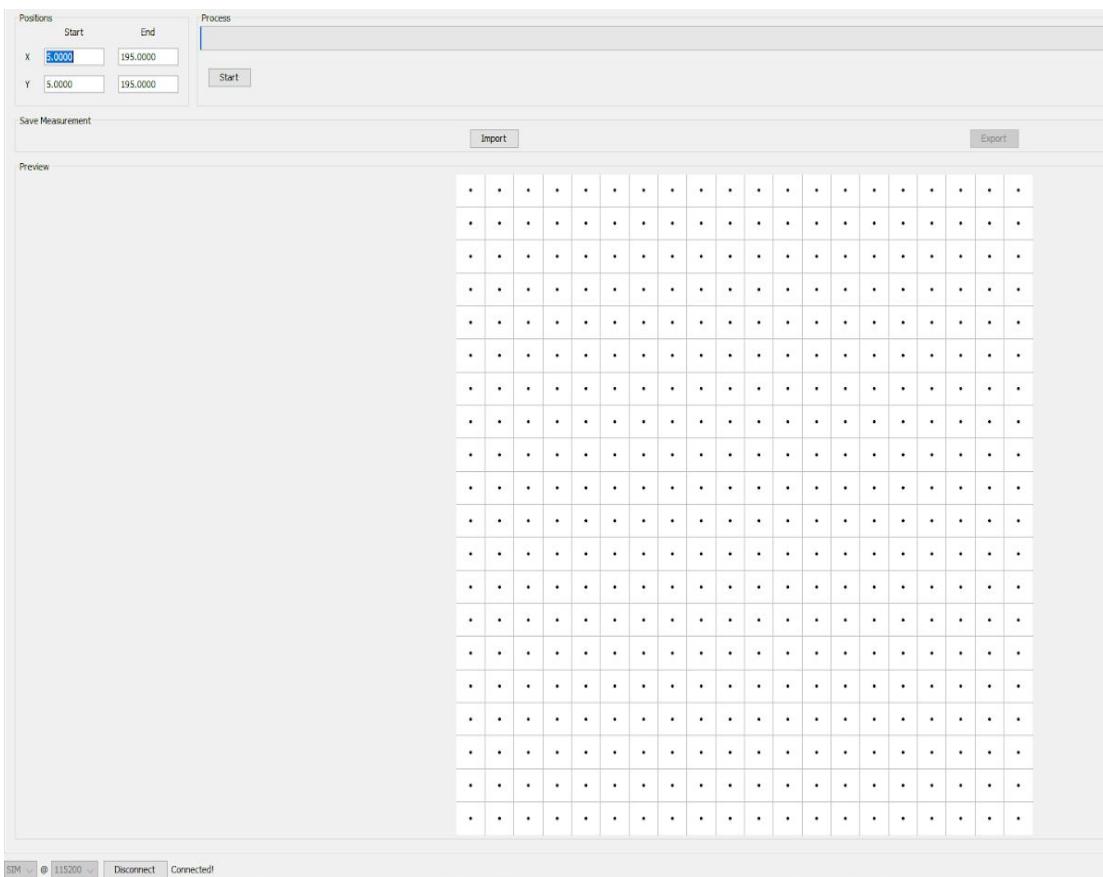
En esta pestaña se encuentran los controles avanzados de la máquina, por ejemplo el avance de la herramienta a una posición específica, indicación de la posición actual de la herramienta, hacer homing general, realizar movimientos complejos de arcos y guardar posiciones previas de la herramienta para, eventualmente, volver a cargarlas. Sumado a este amplio abanico de controles manuales, existe, en la parte izquierda, un visualizador del recorrido de la herramienta.



Pestaña Advanced Controls - CNC GCode Controller.

### - Auto Level:

En esta pestaña controlaremos y configuraremos todo lo referido a la Auto-nivelación. Para mayor información, dirigirse al siguiente apartado “*Utilización del Software*”.

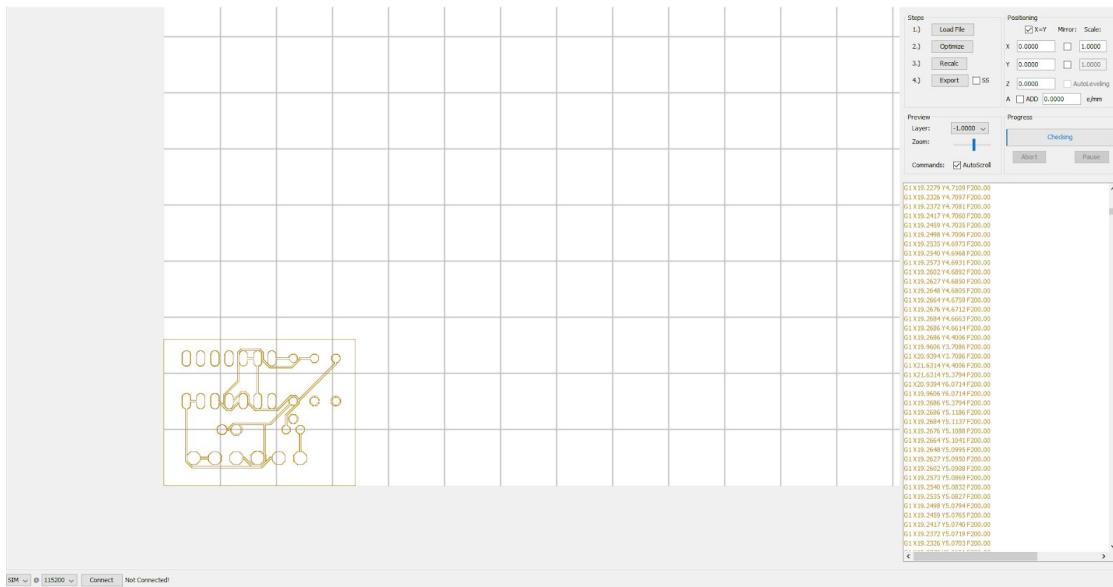


*Pestaña Auto Level - CNC GCode Controller.*

### - CNC Milling:

Mediante esta pestaña cargamos nuestro G-Code con la conveniente posibilidad de optimizarlo. A su vez, en la parte izquierda, se muestra una cuadrícula en donde se describen los movimientos de nuestra herramienta.

Es importante destacar que podremos visualizar el G-Code en la parte inferior derecha, dando la posibilidad de realizar ediciones sobre el mismo.

*Pestaña CNC Milling - CNC GCode Controller.*

#### - Communication:

En esta pestaña se pueden observar todos los comandos que son enviados a la máquina. Dando la posibilidad de enviar comandos manualmente.

The screenshot shows the 'CNC Communication' window. It displays a list of G-code commands in a text-based communication log. One command, 'N3 G28 \*48', is highlighted with a blue background. The log includes both transmitted and received commands, such as 'N1 M110 \*2' and 'ok command: N1 M110 \*2'. The interface has a simple text input field at the bottom for sending new commands.

```

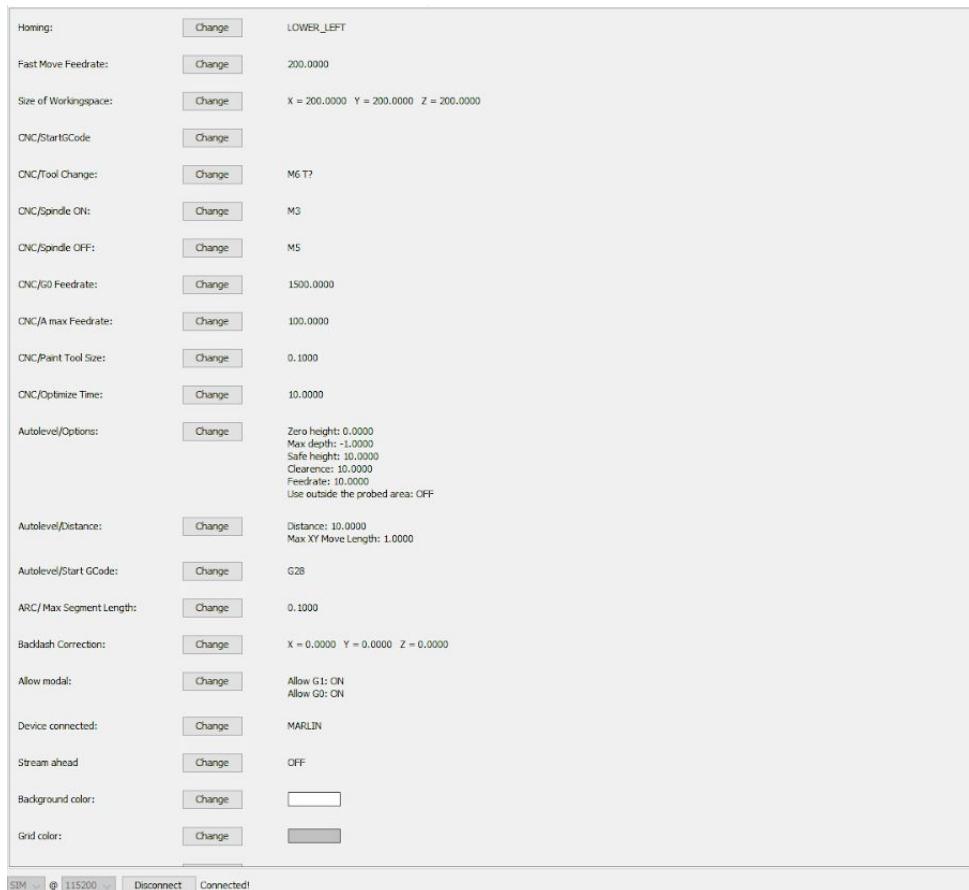
<- 1=>[N1 M110 *2]
<- M110
--> ok command: N1 M110 *2
<- 2=>[N2 G90 *50]
<- G90
--> ok command: N2 G90 *50
<- s
<- 3=>[N3 s *14]
--> ok command: N3 s *14
<- s
<- 4=>[N4 s *9]
--> ok command: N4 s *9
<- s
<- 5=>[N5 s *8]
--> ok command: N5 s *8
<- 1=>[N1 M110 *2]
<- M110
--> ok command: N1 M110 *2
<- 2=>[N2 G90 *50]
<- G90
--> ok command: N2 G90 *50
<- G28
<- 3=>[N3 G28 *48]
--> ok command: N3 G28 *48

```

*Pestaña Communication - CNC GCode Controller.*

### - Settings:

Este apartado tiene el objetivo de permitir la configuración de todo el software en función de las especificaciones de la máquina.



Pestaña Settings- CNC GCode Controller.

#### 5.2.1. Utilización del Software:

5.2.1.1. Dirigirse a la pestaña “Settings” y modificar en base a nuestras necesidades los siguientes apartados:

- **Homing:** Modifica la ubicación del Homing general. Este parámetro dependerá del punto de origen seleccionado al momento de realizar la placa PCB;
- **Size of Workingspace:** Este parámetro puede definirse como un “final de carrera virtual”, pues su objetivo es conocer el recorrido efectivo total que puede realizar la herramienta en cada uno de los ejes, otorgando la posibilidad de evitar que la misma impacte al colocar una coordenada que supere el límite mencionado;

- **CNC/StartGCode:** En este apartado tenemos la posibilidad de introducir un código que se ejecutará antes de comenzar el G-Code;
- **CNC/Spindle ON:** Establecemos con que comando se enciende el Motor del Husillo, lo correcto es colocar M3;
- **CNC/Spindle OFF:** Establecemos con que comando se apaga el Motor del Husillo, lo correcto es colocar M5;
- **Autolevel/Options:** Mediante este parámetro podemos modificar la altura cero, el máximo recorrido en el eje Z, la altura de retirada, la altura de finalizado y la velocidad de la herramienta durante la ejecución de la Auto-nivelación;
- **Autolevel/Distance:** Definiremos la distancia entre los distintos puntos de contacto para la Auto-nivelación;
- **Autolevel/StartGcode:** Configuraremos el código inicial antes de comenzar la autonivelación al momento de iniciar este proceso. Por defecto está puesto el comando G28, es decir, el homing general;
- **Backlash Correction:** Este parámetro nos permite corregir de forma virtual el Backlash que se genera en cada uno de los ejes. Es notable como constituye un excelente complemento con el sistema Anti-Backlash ya descrito;
- **Device Connected:** Aquí se coloca el tipo de firmware que se utiliza, en nuestro caso, Marlin;
- **Fast Move Feedrate:** Configuraremos la velocidad de avance de los movimientos rápidos realizados manualmente mediante la pestaña “*Advanced Controls*”;
- **CNC/Tool Change:** Este parámetro es necesario en caso de utilizar distintas herramientas para mecanizar, puesto que podemos seleccionar el comando con el cual efectuaremos los cambios de las mismas;
- **CNC/G0 Feedrate:** Configura la velocidad de avance con el comando G0. Es importante destacar que Marlin no hace

distinción entre G0 y G1, en cambio, varía la velocidad de avance para cada movimiento en función del valor de “F” colocado en el G-Code del mismo;

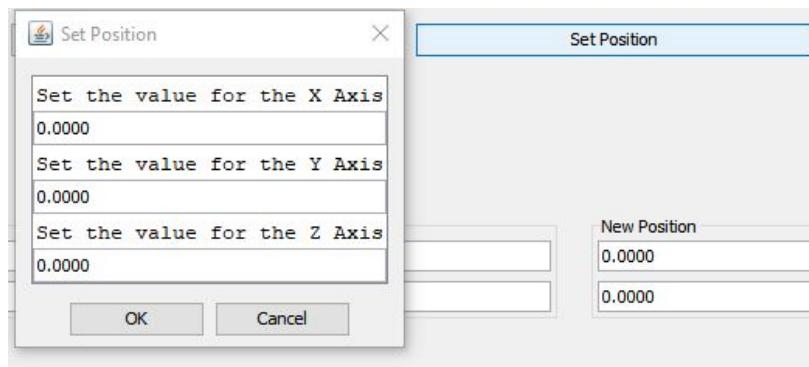
- **CNC/Paint Tool Size:** Con esta opción se define el diámetro de la herramienta. Este parámetro es útil a la hora de utilizar la aplicación en su estado de simulación.

5.2.1.2. Una vez configurada la aplicación y realizada la conexión a nuestra computadora, debemos establecer el puerto a la cual conectamos la máquina, establecer los baudios y presionar el botón “*connect*”.

5.2.1.3. A continuación se debe realizar un Homing de todos los ejes y, luego, colocar manualmente la herramienta en una esquina de la placa de cobre virgen previamente colocada y fijada en la Mesa.



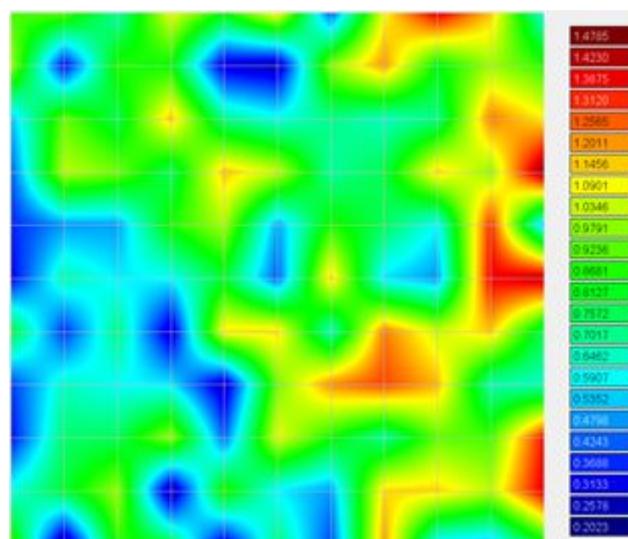
5.2.1.4. Una vez posicionada la herramienta en la esquina deseada, debemos dirigirnos a la pestaña “*Advanced Controls*” y hacer clic sobre “*Set Position*”. A continuación se desplegará un cartel que solicita ingresemos el valor de la nueva posición. Como el objetivo es establecer la esquina mencionada como el nuevo origen de la máquina, debemos colocar 0 y presionar “*Ok*”.



- 5.2.1.5. En este paso debemos realizar la Auto-nivelación. Para ello seleccionaremos la pestaña “*Auto Level*” y, en el apartado “*Position*”, colocaremos las dimensiones de la porción de placa virgen que utilizaremos para el mecanizado.

De este modo, se mostrará una grilla llena de puntos indicando las posiciones en donde la herramienta realizará el mapeo de alturas. Haciendo clic en “*Start*”, la herramienta comenzará a moverse haciendo contacto múltiples veces con la placa de cobre

Luego de realizar la Auto-nivelación, observaremos el mapa de alturas generado en el cual se puede distinguir la altura de todos y cada uno de los puntos mostrados en la grilla antes mencionada.



Mapa de alturas junto con su escala - CNC GCode Controller.

Es importante mencionar que, para realizar una correcta Auto-nivelación, se deben seguir las instrucciones descritas en el Diagrama de Flujo General. Para su consulta dirigirse al documento Diagramas Eléctricos y Electrónicos.

5.2.1.6. Una vez realizada correctamente la Auto-nivelación, podemos comenzar el proceso de mecanizado. Para ello debemos dirigirnos a la pestaña “*CNC Milling*” y cargar el archivo G-Code. A continuación podremos observar el trazado de los movimientos de la herramienta.

Al hacer clic en “*Optimize*” la aplicación optimizará el código cargado y, luego, indicará el tiempo ahorrado. Actualizamos el código oprimiendo “*Recalc*” y marcamos la casilla “*Autoleveling*”. Solo resta presionar el botón “*Milling*” para comenzar el mecanizado.

### 5.3. Firmware Marlin:

A continuación plasmaremos ciertos parámetros básicos que se deben conocer a la hora de utilizar Marlin.

#### 5.3.1. Instrucciones:

- **#define:** Es una de las instrucciones principales. Informa a Marlin aquellos valores y variables que se encuentran definidos. Generalmente, se coloca la variable y luego su valor.
- **#undef:** Es #define pero negado, es decir, quita una variable previamente definida.
- **#ifdef y #endif:** Es una combinación #define y #undef. El microprocesador solo lee las líneas que se encuentra dentro del #ifdef. Tiene principio y fin. Solo se utiliza si la variable utilizada dentro de esta condición estuvo previamente definida con #define.
- **#ifndef y #endif:** A diferencia del anterior la condición se ejecuta si la variable dentro de ella no se encontraba definida previamente.
- **Velocidad del puerto serie:**  
La velocidad que viene por defecto en Marlin es 250.000 baudios, comunicación rápida y con errores mínimos. No obstante, algunas placas no son capaces de manejar estos valores. En este caso lo ponemos en 115.200.  
`#define BAUDRATE 250000`

### 5.3.3. Configuraciones para nuestra aplicación:

Las siguientes configuraciones se encuentran en la carpeta **Configuración.h**

- Definir tipo de placa controladora, en nuestro caso, RAMPS 1.4
- Colocar los datos tanto del autor como de la fecha de compilación.

```
#define STRING_VERSION_CONFIG_H __DATE__ " "
__TIME__ // build date and time. (una vez subido el Marlin
al Arduino, se actualiza la fecha y hora a ese momento)
```

```
#define STRING_CONFIG_H_AUTHOR "(none, default
config)" // Who made the changes.
```

En la última línea es necesario cambiar lo que se encuentra entre comillas (*None, default config*) con la razón social de la empresa.

- Definir el puerto serie de nuestra computadora.

```
#define SERIAL_PORT 0
```

- Definir la utilización de las resistencias Pull Up en los finales de carrera. Es recomendable activarlas en caso de utilizar un simple final de carrera sin ningún tipo de filtros.

```
#define ENDSTOPPULLUPS // Comment this out (using/ at
the start of the line) to disable the EndStop pullup
resistors.
```

En caso de necesitar la activación de solo algunas Pull Up, comentamos la línea previamente citada y descomentamos las que necesitamos.

```
#ifndef ENDSTOPPULLUPS
//define EndStop settings: Individual pullups. Will be ignored if
ENDSTOPPULLUPS is defined
// #define ENDSTOPPULLUP_XMAX
// #define ENDSTOPPULLUP_YMAX
// #define ENDSTOPPULLUP_ZMAX
// #define ENDSTOPPULLUP_XMIN
// #define ENDSTOPPULLUP_YMIN
```

---

```
// #define ENDSTOP_PULLUP_ZMIN
#endif
```

- Verificar si los finales de carrera envían un 1 lógico al ser pulsados, caso contrario, se debe invertir su lógica funcional. Esto se logra colocando “false” o “true” en las siguientes líneas.

```
const bool X_MIN_ENDSTOP_INVERTING = true; // set
to true to invert the logic of the EndStop.
const bool Y_MIN_ENDSTOP_INVERTING = true; // set
to true to invert the logic of the EndStop.
const bool Z_MIN_ENDSTOP_INVERTING = true; // set
to true to invert the logic of the EndStop.
const bool X_MAX_ENDSTOP_INVERTING = true; // set
to true to invert the logic of the EndStop.
const bool Y_MAX_ENDSTOP_INVERTING = true; // set
to true to invert the logic of the EndStop.
const bool Z_MAX_ENDSTOP_INVERTING = true; // set
to true to invert the logic of the EndStop
```

- Verificar que los movimientos de cada ejes se encuentren en la dirección correcta. Caso contrario, se debe invertir el estado de las siguientes líneas.

```
#define INVERT_X_DIR true      // for Mendel set to
false, for Orca set to true
#define INVERT_Y_DIR false     // for Mendel set to
true, for Orca set to false
#define INVERT_Z_DIR true      // for Mendel set to
false, for Orca set to true
```

- Fijar la velocidad de desplazamiento a la hora de realizar un Homing

```
#define HOMING_FEEDRATE {50*60,      50*60,
4*60, 0} // set the homing speeds
(mm/min)
```

- Definir un Homing ultrarrápido. Esta acción consiste en el desplazamiento simultáneo de los ejes. Esta función se activa descomentando la siguiente línea que se encuentra en la carpeta de Configuration\_adv.h:

*//#define QUICK\_HOME //if this is defined, if both x and y are to be homed, a diagonal move will be performed initially.*

- Con el objetivo de evitar grandes vibraciones, se deben definir aceleraciones bajas para los Motores Paso a Paso. Para modificar este valor se deben alterar los valores de la siguiente variable definida:

```
#define      DEFAULT_MAX_ACCELERATION
{9000,9000,100,10000}
#define DEFAULT_ACCELERATION 3000
```

La primera línea define las aceleraciones máximas posibles que pueden alcanzar los Ejes en mm/s<sup>2</sup>. En cambio, la segunda, define las aceleraciones por defecto que tendrán los Motores Paso a Paso en caso de que no se especifiquen en el G-code.

- Existe la posibilidad de variar la frecuencia de las salidas D8; D9 y D10. Por defecto, estas tres salidas poseen frecuencias PWM bajas para evitar interferencias con el hardware. De todos modos, modificando la siguiente línea, se puede cambiar el valor de la frecuencia mencionada:

```
#define SOFT_PWM_SCALE 0
```

Al reemplazar el 0 por un 1, la frecuencia PWM se duplicará. Se debe tener en cuenta que, al realizar esta modificación, la resolución de la frecuencia (128, por defecto) disminuye. Esta función es muy útil a la hora de realizar ajustes en el Circuito de Seguridad y se encuentra en la carpeta **Configuration.h**.

#### 5.3.4. LCD (Full Graphic Smart Controller):

- Definir el lenguaje de la pantalla. La siguiente línea de código se encuentra en el archivo **lenguaje.h**:

```
//#define      LANGUAGE_INCLUD
GENERATE_LANGUAGE_INCLUDE(en)
```

Por defecto se encuentra en inglés (en), lo cambiaremos a español (es).

- Como existe la posibilidad de insertar una tarjeta SD, debemos activar su slot desde el archivo **Configuration.h**.

```
//#define SDSUPPORT // Enable SD Card Support in Hardware Console
```

Si existen problemas con la tarjeta a la hora de iniciar el código cargado en la misma, una posible solución es colocar el lector de tarjeta SD en modo lectura lenta. Para ello, definimos la siguiente línea:

```
//#define SDSLOW // Use slower SD transfer mode (not normally needed -- uncomment if you're getting volume init error).
```

Al mismo tiempo, es recomendable activar la opción para poder verificar errores de envío y recepción de datos en la tarjeta SD. Para su activación, la línea a comentar es:

```
//#define SD_CHECK_AND_RETRY // Use CRC checks and retries on the SD communication.
```

- Configurar la cantidad de pasos que debemos rotar el encoder rotativo para poder navegar entre los menús de la pantalla. Esto se realiza mediante las siguientes dos variables:

```
//#define ENCODER_PULSES_PER_STEP 1 // Increase if you have a high resolution encoder
#define ENCODER_STEPS_PER_MENU_ITEM 5 // Set according to ENCODER_PULSES_PER_STEP or your liking
```

- Descomentar la variable definida que corresponde a nuestro modelo de pantalla. La línea es la siguiente:

```
//#define
REPRAP_DISCOUNT_FULL_GRAPHIC_SMART_CONTROL
LER
```

Es importante mencionar que debemos descargar e incluir la librería **U8glib**, la cual se encuentra incluida en la Carpeta “Programas”

- Ajustar la frecuencia y la duración de Buzzer que posee la pantalla como feed

#### 5.4. Comandos G:

Los Comandos o Códigos G se utilizan para programar funciones relacionadas con el tipo de movimiento u operación. Para nuestra aplicación, lo más utilizados son los siguientes:

- **G0 y G1:** Puesto que Marlin no distingue entre uno y otro, para especificar la velocidad de avance se utiliza la letra F.

Ejemplo:

*G1 X12 F2000*

En esa línea de comando se establece un movimiento línea de 12 mm en el eje X a una velocidad de 2000 mm/min. Entonces, la utilización de G0 tiene el objetivo de diferenciar un movimiento mecanizado de uno realizado en el aire.

- **G2 y G3:** Realiza movimiento de arco, siendo G2 para arco en dirección horaria y G3 en dirección antihoraria. Se especifica el punto final en el eje X, el punto final en el eje Y. Con I se especifica el centro de la circunferencia en el eje X y con J en el eje Y.

Ejemplo:

*G2 X90.6 Y13.8 I5 J10*

- **G4:** Da la instrucción de realizar un retardo por un determinado tiempo. Se puede expresar tanto en segundos como en milisegundos.

Ejemplo:

*G4 S1* (determina una espera de 1 **segundo**)

*G4 P1000* (determina una espera de 1000 **milisegundos**, es decir, 1 segundo)

Es importante destacar que este comando no funciona como delay ya que siguen ejecutándose otras instrucciones al mismo tiempo.

- **G21:** Con este comando determina la utilización del sistema de unidades métrico.

Ejemplo:

*G21*

*G1 X12 F200*

- **G20:** Con este comando determina la utilización del sistema de unidades anglosajón.

Ejemplo:

*G20*

*G1 X12 F200*

- **G28:** Este comando tiene el objetivo de realizar Homing. Se puede especificar en qué eje se realizará el Homing o, directamente, no especificar nada y realizar un Homing para todos los ejes.

Ejemplo:

*G28* (Homing realizado en todos los ejes)

*G28 X Y* (Homing realizado únicamente en el eje X e Y)

- **G90:** Este comando indica que todos los valores de los ejes son absolutos, es decir, que están todos referidos al punto de origen X0; Y0; Z0.

Ejemplo:

*G90*

*G1 X12 F1200*

- **G91:** Este comando indica que todos los valores de los ejes son incrementales o relativos. Es decir, se toma como referencia para el siguiente movimiento las posición actual.

Ejemplo:

*G91*

*G1 X12 F1200*

- **G92:** Es tal vez el comando más importante de todos teniendo en cuenta nuestra aplicación, con él estaremos cambiando las coordenadas reales en las cuales se encuentra la herramienta. En caso de no especificar la nueva coordenada, Marlin interpretará que esa posición de la herramienta es el nuevo origen de la máquina.

Ejemplo:

*G92 X12 Y30 Z2* (especificamos coordenadas, por lo tanto, la posición actual de la herramienta será X12; Y30; Z2).

*G92* (no se realizan especificaciones, por lo tanto, ahora la posición actual de la herramienta es el origen).

### 5.5. Comandos M:

Los Comandos o Códigos M se utilizan para programar funciones específicas de la máquina. Para nuestra aplicación, los más utilizados son los siguientes:

- **M3:** Ordena el encendido del Motor del Husillo
- **M5:** Ordena la detención del Motor del Husillo
- **M114:** Muestra la posición actual de la herramienta.

Ejemplo:

*G0 X25 Y14*

*M114*

# Resumen de Especificaciones

<u>ESPECIFICACIÓN</u>	<u>DATOS</u>
Placa de Control: Arduino Mega + Ramps 1.4	Firmware: Marlin
Motor DC	48V - 300W
Regulación de velocidad del Husillo	3000 a 12000 rpm
Motores Paso a Paso	NEMA 17 - 7,1 kg.cm
Área de útil mecanizado	200 x 200 mm
Programa de Control.	CNC GCode Controller.
Desplazamiento en eje X.	220 mm
Desplazamiento en eje Y.	210 mm
Desplazamiento en eje Z.	60 mm
Alimentación.	220V - 50Hz
Sistema de desplazamiento	Buje de Resina Acetal - Guia Rectificada.
Sistema Anti-Backlash	Tanto mecánico (Tuerca-Resorte) como virtual (CNC GCode Controller).
Conexión a computadora	USB - B
Pantalla Controladora.	LCD Full Graphic Smart Controller
Parada de Emergencia .	300 ms
Ventilación en Área Electrónica	Ventiladores Cooler; 80x80 mm
Fácil acceso al Área Electrónica.	-
Iluminación del Área de Mecanizado.	Tiras LED con Soporte LED difusor
Circuito de Seguridad	-
Bandeja recoge viruta	-

## **Conclusiones**

En este proyecto anual se logró diseñar un Router CNC para placas PCB de manera completamente satisfactoria. Puesto que en su proceso de diseño siempre se priorizaron los requerimientos funcionales, la implementación de nuestra máquina tiene ventajas sobre otros métodos convencionales, logró desarrollar un diseño que cumple con todas las exigencias por parte de la empresa patrocinadora.

El método de diseño seleccionado permitió dividir nuestro proyecto en cuatro etapas: Diseño Conceptual, Configuración, Diseño en Detalle y, por último, Planeación de cara a la manufactura. Al mismo tiempo, en el Grupo de Trabajo se realizaron múltiples divisiones de tareas con el objetivo de realizar día a día avances continuos, sólidos y funcionales. Preservando siempre la Constancia, la Exigencia y el Compromiso.

Lo que comenzó siendo una simple idea de una “Solución Electromecánica”, se convirtió en un ensamble 3D con más de 1000 componentes, en 100 planos tanto mecánicos como eléctricos y electrónicos, en más de 60 diagramas de operaciones, en 4 circuitos de control, generando más de 50 GB de datos y un sinfín de avances y resultados completamente realizadores comprendidos en más de 5000 horas de trabajo en un periodo de, prácticamente, 8 meses.

# Oportunidades de Mejora

En este apartado colocaremos aquellas cuestiones que, consideramos, pueden ser mejoradas y/o re-diseñadas, teniendo en cuenta, nuevamente, que el diseño de una máquina es un proceso en el cual se toman decisiones con la información incompleta para luego volver al punto de inicio y realizar las mejoras pertinentes.

## 1. Realización de MKS MOS propio:

La la manufactura de una PCB del circuito MKS MOS pero con los componentes adecuados para nuestra aplicación otorga dos claras ventajas. Por un lado, una reducción significativa de costos y, por el otro, la posibilidad de controlar la velocidad del Motor DC mediante PWM.

## 2. Implementación del circuito Schmitt Trigger:

Como se mencionó, la utilización del diodo Zener en el Circuito de Seguridad implica que D9 entregue, como mínimo, 5V. La solución a este problema es manufaturando el circuito Schmitt Trigger, aumentando la sensibilidad del circuito.

## 3. Investigación y utilización del software Chilipeppr:

Recientemente conocimos la existencia de un software denominado “Chilipeppr”. El mismo nos permite generar y enviar el G-Code en función de un diseño CAD de una PCB. Su principal ventaja con respecto a FlatCAM radica en que es capáz de generar un archivo completo de taladrado, enrutado de pistas y corte.

## 4. Reducción de la cantidad de buloneria utilizada:

Consideramos que, teniendo en cuenta el costo de la buloneria y cuestiones estéticas, se debería reducir la cantidad de tornillos utilizados, sobre todo, en el Subconjunto Carcasa.

## 5. Transportabilidad:

Una de las desventajas de la máquina es su baja capacidad de transportabilidad provocada por en la misma no se ha generado una estructura que permita sujetar la máquina y, de ese modo, transportarla.

## 6. Tornillo trapezoidal:

Creemos que es necesario reconsiderar la utilización de un tornillo trapezoidal para todos los tornillos de movimientos. Otorgando una mayor resistencia mecánica, generando un aumento en la velocidad de desplazamiento y, como consecuencia, una disminución en los tiempos de mecanizado.

## **Bibliografía (Biblioteca Virtual)**

La “*Biblioteca Virtual*” es una Carpeta Compartida de Google Drive en la cual se encuentran todos documentos de consulta que utilizamos a lo largo del proyecto. A continuación se adjunta el vínculo para acceder a la misma.

- [Biblioteca Virtual - Grupo 1](#)