

**UNIVERSIDAD TÉCNICA DEL NORTE  
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS CARRERA  
DE INGENIERÍA EN MECATRÓNICA**



**TRABAJO DE GRADO PREVIO LA OBTENCIÓN DEL TÍTULO  
DE  
INGENIERO EN MECATRÓNICA**

**TEMA:**

**“MÁQUINA DE CONTROL NUMÉRICO COMPUTACIONAL PARA  
CORTE DE PLANCHAS DE ALUMINIO POR DESBASTE DE  
MATERIAL: IMPLEMENTACIÓN”**

**AUTOR:**

**HÉCTOR ANDRÉS INBACUÁN TAMBA**

**DIRECTOR:**

**ING. COSME DAMIÁN MEJÍA ECHEVERRÍA**

**IBARRA – ECUADOR**

**2019**



## UNIVERSIDAD TÉCNICA DEL NORTE BIBLIOTECA UNIVERSITARIA

### AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

#### 1. IDENTIFICACIÓN DE LA OBRA

En cumplimiento del Art. 144 de la Ley de Educación Superior, hago la entrega del presente trabajo a la Universidad Técnica del Norte para que sea publicado en el Repositorio Digital Institucional, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO			
<b>Cédula de identidad:</b>	1003639752		
<b>Apellidos y nombres:</b>	Inbacuán Tamba Héctor Inbacuán		
<b>Dirección :</b>	Ibarra, Calle el Limonal 3-60 y el Milagro		
<b>Email :</b>	hainbacuan@utn.edu.ec		
<b>Teléfono fijo:</b>	062542007	<b>Teléfono móvil:</b>	0997548629

DATOS DE LA OBRA	
<b>Título:</b>	“Máquina de control numérico computacional para corte de planchas de aluminio por desbaste de material: Implementación”
<b>Autor:</b>	Inbacuán Tamba Héctor Andrés
<b>Fecha:</b>	17 de mayo de 2019
<b>Programa:</b>	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSGRADO
<b>Título por el que opta:</b>	Ingeniería Mecatrónica
<b>Asesor / director:</b>	Ing. Cosme Mejía

#### 2. CONSTANCIA

El autor manifiesta que la obra concepto de la presente autorización es original y se la desarrollo, sin violar los derechos de autor de terceros, por lo tanto la obra es original y que es el titular de los derechos patrimoniales, por lo que asume la responsabilidad sobre el contenido de la misma y saldrá en defensa de la Universidad en caso de reclamación por parte de terceros.

Ibarra, a los 17 días del mes de mayo de 2019

#### EL AUTOR

Héctor Andrés Inbacuán Tamba



## UNIVERSIDAD TÉCNICA DEL NORTE

### FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

#### CERTIFICO

Que la Tesis previa a la obtención del título de Ingeniero en Mecatrónica con el tema **“MÁQUINA DE CONTROL NUMÉRICO COMPUTACIONAL PARA CORTE DE PLANCHAS DE ALUMINIO POR DESBASTE DE MATERIAL: IMPLEMENTACIÓN”**, ha sido desarrollado y terminado en su totalidad por el Sr. Héctor Andrés Inbacuán Tamba, con cedula de identidad: 100363975-2, bajo mi supervisión para lo cual firmo en constancia.



.....  
Ing. Cosme Mejía  
**DIRECTOR**

## **AGRADECIMIENTO**

Agradezco a Dios por haberme dado la vida y la fuerza necesaria para culminar con mis estudios.

A mis padres Delia Tamba y Luis Luna que siempre me han formado con valores y ejemplo, por brindarme su apoyo incondicional durante toda mi vida, aconsejándome, guiándome por el camino correcto y siempre estar atentos de mi bienestar.

A Lizbeth Yépez que siempre me ha dado ánimo para culminar con mis estudios, por brindarme su amor incondicional y estar a mi lado en los buenos y malos momentos.

Al Ingeniero Cosme Mejía, director de tesis por su confianza y apoyo durante todo el proceso de realización del trabajo de grado

Héctor Inbacuán

## **DEDICATORIA**

Este trabajo va dedicado a mi familia que siempre ha estado pendiente de mí, apoyándome y dándome consejos y palabras de aliento para culminar con mi carrera.

## RESUMEN

En el presente trabajo se muestra el desarrollo de una Interfaz Humano-Máquina (HMI) para el posicionamiento del cero pieza en una Máquina de Control Numérico Computacional (CNC) basada en software y hardware libre. Este trabajo tiene la finalidad de realizar una interfaz amigable con el usuario para que el proceso del posicionamiento del cero pieza se realice de forma rápida y sencilla.

La selección de los elementos se realiza con la utilización de la herramienta QFD (Quality Function Diagram), en la cual colocamos los requerimientos del usuario convertidos a objetivos medibles, se asigna una ponderación a cada uno de los objetivos y al final el diagrama determina cuales son los objetivos principales.

Para la construcción de la estructura mecánica de la máquina se toma los planos obtenidos del trabajo de pregrado llamado “Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Diseño”

## ABSTRACT

In the present work the development of a Human-Machine Interface (HMI) is shown for the positioning of the piece zero in a Computational Numerical Control Machine (CNC) based on software and free hardware. This work has the purpose of making a friendly interface with the user so that the process of positioning the part zero is done quickly and easily.

The selection of the elements is made with the use of the QFD tool (Quality Function Diagram), in which we place the user's requirements converted to measurable objectives, a weighting is assigned to each of the objectives and in the end the diagram determines which They are the main objectives.

For the construction of the mechanical structure of the machine, the drawings obtained from the undergraduate work called "Computational Numerical Control Machine for cutting aluminum plates by roughing material: Design" are taken.

## INDICE

1. IDENTIFICACIÓN DE LA OBRA .....	II
2. CONSTANCIA.....	II
CERTIFICO.....	III
AGRADECIMIENTO .....	IV
DEDICATORIA .....	V
RESUMEN .....	VI
ABSTRACT.....	VII
INTRODUCCIÓN .....	XIV
 CAPÍTULO I .....	 1
1. MARCO REFERENCIAL.....	1
1.1. Investigaciones actuales.....	1
1.2. Desarrollo actual.....	3
1.3. Métodos de manufactura.....	4
1.4. Hardware y software para control de motores paso a paso .....	5
1.4.1. Motores paso a paso .....	5
1.4.2. Control de motores paso a paso .....	8
 CAPÍTULO II.....	 23
2. METODOLOGÍA.....	23
2.1. Despliegue de la función de calidad (QFD) .....	23
2.1.1. Fijación del objetivo .....	23
2.1.2. Voz del usuario .....	23
2.1.3. Voz del ingeniero.....	24
2.1.4. Diagrama de la casa de la calidad (QFD).....	26
2.1.5. Conclusiones de la casa de la calidad .....	27
2.2. Conceptualización.....	27
2.2.1. Análisis funcional .....	27
2.2.2. Desarrollo de los diagramas funcionales .....	28
2.2.3. Análisis de los diagramas funcionales.....	29



2.2.4. Solución de los módulos.....	30
2.2.5. Módulo 1.....	30
2.2.6. Módulo 2.....	40
2.2.7. Propuesta de diseño .....	52
2.3. Diseño de HMI .....	53
2.3.1. Diagrama de flujo .....	54
2.4. Construcción de estructura de la máquina CNC.....	59
2.4.1. Estructura externa .....	59
2.4.2. Sistema de movimiento .....	60
2.5. Implementación de la HMI.....	66
2.5.1. Selección de fuentes de alimentación.....	67
2.5.2. Cálculo de la protección eléctrica.....	68
2.5.3. Implementación elementos electrónicos.....	69
2.5.4. Implementación algoritmo de programación para el control de la máquina CNC.....	72
CAPÍTULO III.....	86
3. PRUEBAS Y RESULTADOS.....	86
3.1. Pruebas de desplazamiento .....	86
3.2. Pruebas de tolerancia del material .....	87
3.3. Obtención de código G de un modelo .....	88
3.3.1. Pasos para obtener código G en Inkscape de una imagen.....	89
3.4. Pruebas de corte de material.....	93
3.5. Tabla de resultados .....	95
4. CONCLUSIONES Y RECOMENDACIONES .....	96
4.1. Conclusiones.....	96
4.2. Recomendaciones .....	97
BIBLIOGRAFÍA .....	98

## ÍNDICE DE FIGURAS

Figura 1.1. Paso simple [11]. .....	6
Figura 1.2. Paso Doble [11]. .....	7
Figura 1.3. Medio Paso [11]. .....	7
Figura 1.4. Diagrama de conexión de una controladora para CNC Modular [14].....	9
Figura 1.5. Placas Arduino [15].....	10
Figura 1.6. CNC Shield v3 [18]. .....	11
Figura 1.7. RAMPS 1.4 [20]. .....	12
Figura 1.8. Raspberry Pi CNC Shield [23]. .....	14
Figura 1.9. Controladora Integrada [25]. .....	15
Figura 1.10. Software Match 3 [27]. .....	17
Figura 1.11. Interfaz TurboCNC [29]. .....	18
Figura 1.12 LinuxCNC Interfaz Axis [31]. .....	19
Figura 1.13. Interfaz cncControl [32]. .....	19
Figura 1.14. Interfaz Pronterface. [34]. .....	20
Figura 2.1. Diagrama de la Función de Calidad .....	26
Figura 2.2. Diagrama Funcional, Nivel 0. ....	28
Figura 2.3. Diagrama Funcional, Nivel 1 .....	28
Figura 2.4. Diagrama Funcional, Nivel 2 y módulos.....	29
Figura 2.5. Switch (Pulsador) [40]. .....	31
Figura 2.6. Comunicación Serial [41]. .....	31
Figura 2.7. Comunicación en Paralelo [42]. .....	32
Figura 2.8. Final de Carrera [44]. .....	33
Figura 2.9. Sensor de presencia magnético [45]. .....	34
Figura 2.10. Sensor de proximidad inductivo [46]. .....	35
Figura 2.11. Computador [47]. .....	41
Figura 2.12. Microcontrolador [48]. .....	42
Figura 2.13. Sistema Embebido [49]. .....	43
Figura 2.14. Software de control CNC [50]. .....	44
Figura 2.15. Tablero de control CNC [52]. .....	45
Figura 2.16. Control Manual CNC. ....	46
Figura 2.17. Interprete Código G libre (Marlin) [53]. .....	47
Figura 2.18. Tarjeta controladora CNC (Mach 3) [55]. .....	48

Figura 2.19. Diagrama de Flujo. ....	54
Figura 2.20. Elementos de Control alámbrico .....	56
Figura 2.21. Estructura externa de la máquina CNC. ....	60
Figura 2.22. Varilla lisa 20mm diámetro (Acero plata) [57]. ....	60
Figura 2.23. Sujetador de pared para guías [58]. ....	61
Figura 2.24. Varilla roscada.....	61
Figura 2.25. Acople flexible ranurado de aluminio [60]. ....	61
Figura 2.26. Rodamiento lineal cerrado y tipo T [77]. ....	62
Figura 2.27. Chumacera de pared 20mm [61]. ....	62
Figura 2.28. Estructura del eje X. ....	63
Figura 2.29. Moldes en MDF.....	64
Figura 2.30. Estructura eje X y Z.(Torre) .....	64
Figura 2.31. Estructura para acoplamiento de rodamientos, tornillo y guías. ....	65
Figura 2.32. Estructura y sistema de movimiento acoplados.....	66
Figura 2.33. Kit husillo 500W con control de velocidad [66]. ....	68
Figura 2.34. Protección eléctrica 20 A [67]. ....	69
Figura 2.35. Gabinete de Control.....	69
Figura 2.36. Interior del gabinete de control y sus fases. ....	70
Figura 2.37. Conector 110V AC.....	71
Figura 2.38. Fase de salida.....	71
Figura 2.39. Archivo Marlin.ino .....	72
Figura 2.40. Función Imagen de inicio personalizada. ....	73
Figura 2.41. Conversión de imagen con herramienta Bitmap Converter [69]. ....	73
Figura 2.42. Pestaña _Bootscreen.h.....	74
Figura 2.43. Configuración de puerto y velocidad de comunicación. ....	74
Figura 2.44. Placas soportadas por Marlin.....	75
Figura 2.45. Configuración de placa.....	75
Figura 2.46. Configuración de fuente de alimentación.....	75
Figura 2.47. Configuración de finales de carrera.....	76
Figura 2.48. Configuración de pasos/mm.....	77
Figura 2.49. Configuración de Avance por defecto.....	78
Figura 2.50. Configuración del área de trabajo.....	78
Figura 2.51. Activación de límites por software.....	79

Figura 2.52. Selección de idioma para LCD.....	79
Figura 2.53. Activación de opción Full Graphics Display.....	80
Figura 2.54. Activación de opción SDSUPPORT. ....	80
Figura 2.55. Configuración de encoder.....	81
Figura 2.56. Activación de la opción SPEAKER. ....	81
Figura 2.57. Activación de la opción INDIVIDUAL_AXIS_HOMING .....	81
Figura 2.58. Configuración del tipo de controlador de LCD.....	82
Figura 2.59. Definición de distancia de retroceso y divisor de velocidad. ....	82
Figura 2.60. Configuración de sección Spindle/Láser. ....	83
Figura 2.61. Fórmula para el cálculo de PWM en relación con las revoluciones del husillo.....	83
Figura 2.62. Configuración de Velocidad de husillo. ....	84
Figura 2.63. Implementación de Funciones en Custom User Menu.....	85
Figura 3.1. Estructura completa de la máquina CNC. ....	86
Figura 3.2. Inserción de imagen en el área de trabajo de Inkscape .....	89
Figura 3.3. Cuadro de diálogo Vectorizar mapa de bits .....	90
Figura 3.4. Definición de puntos de orientación y profundidad de mecanizado. ....	90
Figura 3.5. Cuadro de diálogo Biblioteca de Herramientas.....	91
Figura 3.6. Generación de Herramienta.....	91
Figura 3.7. Opción Trayecto a GCode .....	92
Figura 3.8. Código G generado.....	92
Figura 3.9. Simulación código G mediante NC Viewer [74]. ....	93
Figura 3.10. Selección de archivo desde SD.....	94
Figura 3.11. Corte de pieza. ....	95

## ÍNDICE DE TABLAS

Tabla 1-I. Tabla descriptiva de los códigos G [35].	21
Tabla 2-I. Soluciones para el módulo 1.	36
Tabla 2-II. Ponderación de los criterios para el módulo 1.	37
Tabla 2-III. Evaluación de las alternativas, de acuerdo al criterio de Velocidad de Respuesta.	38
Tabla 2-IV. Evaluación de las alternativas, de acuerdo al criterio: Precio.	38
Tabla 2-V. Evaluación de las alternativas, de acuerdo al criterio: Fácil Operación.	39
Tabla 2-VI. Evaluación de las alternativas, de acuerdo al criterio de Fiabilidad.	39
Tabla 2-VII. Evaluación de las alternativas, de acuerdo al criterio de Mantenimiento.	40
Tabla 2-VIII. Tabla de ponderación de los resultados	40
Tabla 2-IX. Combinación de las soluciones del módulo 2.	49
Tabla 2-X. Ponderación de los criterios del módulo 2.	50
Tabla 2-XI. Evaluación las alternativas respecto al criterio de Velocidad de Respuesta.	50
Tabla 2-XII. Evaluación las alternativas respecto al criterio de Precio.	51
Tabla 2-XIII. Evaluación las alternativas respecto al criterio de Velocidad de Implementación.	51
Tabla 2-XIV. Evaluación las alternativas respecto al criterio de Fácil Operación.	52
Tabla 2-XV. Tabla de Resultados para el módulo 2.	52
Tabla 3-I. Prueba de tolerancia del material.	88
Tabla 3-II. Tabla de resultados	95

## INTRODUCCIÓN

### PLANTEAMIENTO DEL PROBLEMA

La tecnología de Control Numérico Computacional (CNC) se ha desarrollado en los últimos años de manera acelerada, adaptándose a varios campos, desde impresoras 3D hasta fresadoras capaces de manejar 5 ejes, haciendo que la manufactura de piezas se realice en un tiempo menor y con una mayor precisión. Estos dos aspectos son de vital importancia en la producción en serie, debido a que se aumenta la cantidad y la calidad de los productos. Estos beneficios han hecho que las empresas de diversos tipos vayan incorporando esta tecnología en sus procesos de producción.

En el Ecuador esta tecnología no se ha aprovechado totalmente, debido a que no se ha desarrollado y para adquirirla debe ser importada de otros países, haciendo que los costos de adquisición sean muy altos. Además, el soporte es limitado o nulo en algunos casos, haciendo muy complicado una asesoría o mantenimiento de la maquinaria.

En la actualidad existen muchos dispositivos basados en hardware y software libre que son capaces de manejar firmwares CNC, su mayor aplicación está dirigida hacia las impresoras 3D, pero realizando algunas modificaciones en la programación del firmware se puede manejar un husillo o rayo láser. Muchas de las modificaciones señaladas ya vienen por defecto en algunos firmwares, que en caso de realizar una CNC Mill o una con rayo láser, solo se tiene que activar la configuración deseada.

## OBJETIVOS

### - **Objetivo General**

- Construir una máquina CNC para corte de planchas de aluminio por desbaste de material.

### - **Objetivos Específicos**

- Construir la estructura de la máquina CNC.
- Construir el sistema de movimiento de la máquina CNC.
- Implementar el algoritmo de programación para el control de la máquina CNC.
- Diseñar e implementar una Interfaz Humano – Máquina (HMI)
- Realizar pruebas de funcionamiento del sistema electrónico y de control acoplado con el sistema mecánico.

### - **Alcance**

La máquina CNC será capaz de realizar cortes en aluminio de hasta 5mm de espesor, con dimensiones máximas de 80x80cm. El sistema contará con finales de carrera para el sistema de movimiento como una protección, además de un botón de paro de emergencia para situaciones imprevistas.

El controlador propuesto para el proyecto será una tarjeta electrónica con hardware y software específico para máquinas CNC, además estará basado en software libre.

Constará de una HMI fácil de manejar, que permita el control manual de la posición de los ejes X, Y, Z para poder configurar el cero pieza.

Se incluirá un manual de funcionamiento de la máquina.

### - **Justificación**

En la ciudad de Ibarra no se encuentra presente de forma significativa la gran industria, la mayor parte son medianas o pequeñas, empleando operarios que al ser seres humanos no pueden garantizar al 100% la calidad y precisión, demandadas por el mercado. Además, no pueden manejar altos volúmenes de producción. Este último es el factor más importante para que una empresa se desarrolle.

La solución para estos inconvenientes son las máquinas, que optimizan el proceso de producción, ya sea solo un proceso o una línea completa. Aumentan la productividad, minimizan los costos, incrementan la calidad del producto terminado, optimizan los procesos de producción llegando a reducir el tiempo empleado en el desarrollo del producto.

La adquisición de una máquina CNC hace que los tiempos de corte disminuyan, además se pueden realizar formas de diversas complejidades. Este tipo de máquinas tienen un elevado precio para las empresas en desarrollo y las más accesibles son las que provienen de China, la mayoría no poseen suficientes garantías y muchas veces tampoco servicio técnico en el medio.

El desarrollo de una máquina CNC en nuestro medio puede reducir los costos de adquisición, la construcción se realiza respecto a las necesidades del usuario, evitando desperdiciar recursos en aspectos secundarios.



## CAPÍTULO I

### 1. MARCO REFERENCIAL

#### 1.1. Investigaciones actuales

Los primeros sistemas de control numérico se construyeron en la década de 1940 y 1950, pero no eran computarizados, sino que poseían circuitos integrados con esquemas fijos de pequeña y mediana escala de integración, que leían los programas de control grabados en cintas perforadas [1].

En la actualidad se desarrolla tecnología para optimizar el proceso de producción, mejorar el acabado, reducir el tiempo de mecanizado, supervisión inteligente. Estos criterios influyen directamente en la calidad, recursos y tiempo de producción empleados en el desarrollo del producto [2].

Gran parte de las empresas tienen sus prioridades establecidas, realizan investigaciones específicas para mejorar sus productos y así brindar a sus clientes un valor agregado. Un claro ejemplo son las empresas tecnológicas como SpaceX que realiza investigación continua para el desarrollo de cohetes reutilizables y hacer el viaje al espacio un servicio turístico [3].

En la mejoría de la eficiencia del proceso productivo influye de manera significativa el hecho de que no se pueden alcanzar incrementos significativos de la productividad a partir de los esquemas de control clásicos. Dos caminos interesantes para abordar la optimización son la introducción de niveles de supervisión y coordinación en los lazos de control instalados, y la posibilidad de añadir nuevos reguladores utilizando como variables controladas aquellas que influyen de manera decisiva en el proceso productivo [4].

Esta investigación hace la comparación uso de la visión artificial, un sistema de control de lazo cerrado y un regulador externo para la optimización del acabado en la mecanización de una pieza.

A la vanguardia de la industria 4.0, TCI Cutting perfecciona y actualiza permanentemente la inteligencia artificial de sus máquinas de corte láser. La meta es computarizar el mundo de la fábrica y conseguir así que los clientes aumenten la productividad ahorrando en energía y materiales. En definitiva, ahorro de costes y aumento de la productividad [5].

El uso de la inteligencia artificial por parte de esta empresa está enfocado en reconocer las dimensiones de la chapa que va a ser mecanizada de forma precisa; sus contornos, orificios existentes eliminando la necesidad de supervisión y garantizando un mejor acabado.

## 1.2. Desarrollo actual

Las máquinas CNC son capaces de mover una herramienta, en hasta cinco grados de libertad, para obtener figuras tridimensionales complejas. Una vez programado el ordenador, la máquina funciona sola permitiendo al operario dedicarse a otras tareas, con la consiguiente optimización del aprovechamiento del personal.

El desarrollo de este tipo de maquinaria tenía como objetivos fundamentales el incremento de la producción, la reducción del coste de mano de obra directa y la mayor precisión de las piezas resultantes del proceso [6].

En la actualidad el aspecto principal en el que se ha enfocado el desarrollo de las nuevas máquinas CNC ha sido el acabado del mecanizado, este ha sido el valor agregado que las industrias de tecnología CNC han ido desarrollando con mayor énfasis.

Un claro ejemplo es la empresa TCI Cutting que incorporará una tecnología de visión artificial avanzada en sus máquinas corte por láser. El objetivo es que las máquinas detecten a la perfección la posición de la chapa a cortar en la máquina de corte láser, así como las figuras y contornos. Además, este sistema permite la monitorización y la automatización total del proceso en la máquina corte por láser [5].

TCI Cutting ha saltado de lleno a la Industria 4.0 con un nuevo modelo productivo: una fábrica inteligente para producir máquinas de corte por láser inteligentes. Los desafíos planteados por la conocida como cuarta revolución industrial se han resuelto con soluciones de software que permiten analizar constantemente datos e informaciones para aumentar la productividad, la flexibilidad y la eficiencia de la producción de las máquinas corte por láser [5].

### 1.3. Métodos de manufactura

La ingeniería de manufactura es una función que realiza el personal técnico, y está relacionada con la planeación de los procesos de manufactura para el desarrollo económico de productos de alta calidad. Su papel principal consiste en preparar la transición del producto desde las especificaciones de diseño hasta la manufactura de un producto físico [7].

Los métodos de manufactura pueden clasificarse en dos grupos: el enfoque tradicional y la manufactura avanzada.

En el enfoque tradicional el desarrollo de un producto se desarrolla de forma secuencial, primero se realiza la fase de diseño, una vez concluida, se da paso a la fase de prototipado y al final la elaboración del producto final. Todas estas fases se ven aisladas entre sí.

La ingeniería concurrente o simultánea al contrario del enfoque tradicional se basa en la sinergia entre las fases de la elaboración de un producto, eso no quiere decir que todas las fases se realicen al mismo tiempo, sino que cada una de las fases pueda aportar a otra para evitar inconvenientes o para agilizar la elaboración del producto, por ejemplo, mientras la fase de diseño se realiza, la fase de elaboración puede aportar con los materiales disponibles, así se pueden evitar inconvenientes en la fabricación del producto final, ahorrando tiempo y dinero a la empresa.

En la actualidad las empresas buscan que sus productos sean llevados al mercado en el menor tiempo posible y con la mayor calidad, para cumplir con estos objetivos se hace uso de la ingeniería concurrente o ingeniería simultánea, que involucra a la ingeniería de diseño, ingeniería de manufactura y en algunos casos otras áreas de la empresa en el desarrollo del producto, a diferencia del enfoque tradicional en el cual se realiza la fase de diseño del producto sin tomar en cuenta si la empresa cuenta con los recursos necesarios para elaborar ese producto. Las fases del desarrollo del producto se realizan de forma independiente sin tomar en cuenta los inconvenientes que se puedan generar a lo largo de la elaboración del producto final.

## **1.4. Hardware y software para control de motores paso a paso**

### **1.4.1. Motores paso a paso**

Los motores paso a paso son utilizados en la actualidad en muchas aplicaciones debido a que poseen una gran precisión, a diferencia de los demás tipos de motores de corriente continua.

Los motores paso a paso tienen un comportamiento del todo diferente al de los motores de corriente continua. En primer lugar, no giran libremente por sí mismos. Los motores paso a paso, como lo indica su nombre, avanzan girando por pequeños pasos. También difieren de los motores de CC en la relación entre velocidad y torque (un parámetro que también es llamado "par motor" o "par de giro"). Los motores de CC no son buenos para ofrecer un buen torque a baja velocidad sin la ayuda de un mecanismo de reducción. Los motores paso a paso, en cambio, trabajan de manera opuesta: su mayor capacidad de torque se produce a baja velocidad [8].

Se dividen en dos categorías principales: de imán permanente y de reluctancia variable. También existe una combinación de ambos, a los que se les llama híbridos [9].

Los de imán permanente son los más conocidos, generalmente son usados en las impresoras, en el mecanismo de apertura y cierre de las unidades de CD-DVD, etc. Como su nombre indica, poseen un imán que aporta el campo magnético para la operación.

Los motores paso a paso de imán permanente se dividen en distintos tipos, diferenciados por el tipo de bobinado. Existen entonces motores paso a paso de imán permanente: unipolares, bipolares y multifase. Para hacer girar a los motores se les debe aplicar un pulso, este pulso hará que el motor realice un desplazamiento angular determinado que puede ser desde  $1.8^\circ$  hasta  $90^\circ$ , depende de cada motor y del sistema de control aplicado [8].

Para que el motor paso a paso realice un movimiento deseado, se necesita que las bobinas del motor se activen en una secuencia determinada, si el orden de la secuencia se invierte,

el giro del motor también se invertirá, existen 3 tipos de secuencias para que el motor realice un movimiento [10].

Las secuencias son:

- **Paso simple:** Esta secuencia de pasos es la más simple de todas y consiste en activar cada bobina una a una y por separado, con esta secuencia de encendido de bobinas no se obtiene mucha fuerza ya que solo es una bobina cada vez la que arrastra y sujeta el rotor del eje del motor [11]. Se ilustra en la Figura 1.1.

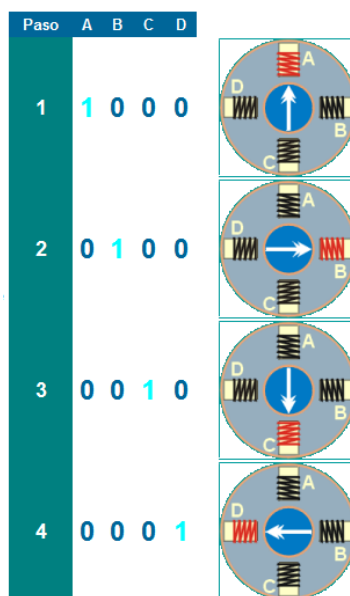


Figura 1.1. Paso simple [11].

- **Paso doble:** Con el paso doble activamos las bobinas de dos en dos con lo que hacemos un campo magnético más potente que atraerá con más fuerza y retendrá el rotor del motor en el sitio. Los pasos también serán algo más bruscos debido a que la acción del campo magnético es más poderosa que en la secuencia anterior [11]. Esta configuración es mostrada en la Figura 1.2.

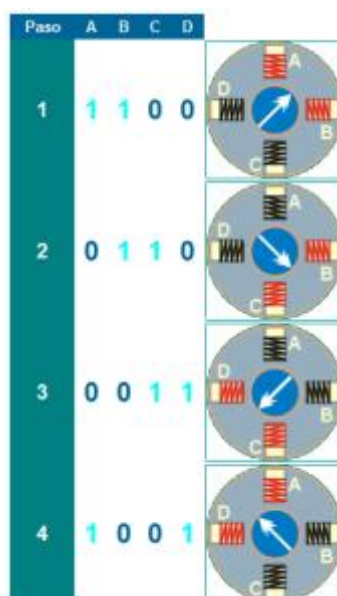


Figura 1.2. Paso Doble [11].

- Medio Paso:** Combinando los dos tipos de secuencias anteriores podemos hacer moverse al motor en pasos más pequeños y precisos y así pues tenemos el doble de pasos de movimiento para el recorrido total de  $360^\circ$  del motor [11]. En la Figura 1.3 se ilustra la secuencia de medio paso.

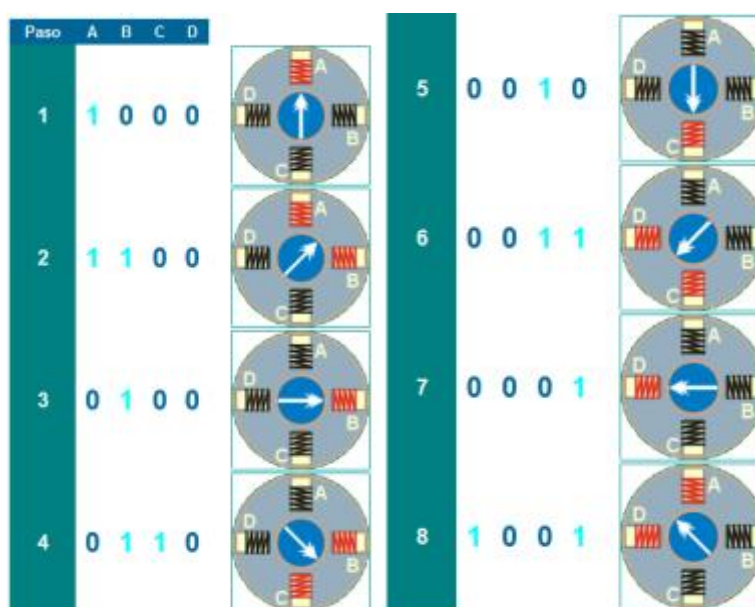


Figura 1.3. Medio Paso [11].

## 1.4.2. Control de motores paso a paso

### 1.4.2.1. Hardware

Los motores paso a paso al necesitar de una secuencia ordenada de señales es necesario tener elementos electrónicos para permitir su giro, se detallan a continuación:

- **Microcontrolador con Controlador:** El microcontrolador es el encargado de generar la secuencia determinada de pulsos para que el motor gire [12], mientras que el controlador tiene la función de tomar una señal de control de baja corriente y luego convertirla en una señal de corriente más alta que pueda conducir un motor [13]. Su función es la de convertir la señal enviada desde el microcontrolador, en secuencia de conmutación para la activación de las bobinas de un motor.

En la actualidad existen placas electrónicas para el control de los motores paso a paso que se conectan a un computador y mediante un software enviar las instrucciones haciendo el trabajo del microcontrolador. El software que se emplee depende de la placa controladora y puede o no ser licenciado.

A continuación, se muestran las placas controladoras y el software que existen en el mercado para el control de motores paso a paso.

#### - **Controladora modular**

Es una placa electrónica que como su nombre lo indica se compone de módulos, desde la interfaz o método de conexión, controladores de los motores, sensores y actuadores, se encuentran separados, una desventaja de este tipo de controladoras es que ocupan mucho espacio. Una gran ventaja es su versatilidad, si se requiere realizar una modificación o una actualización no es necesario cambiar todo el sistema sino solo cambiar o añadir el elemento deseado. Generalmente una controladora posee al menos una interfaz para la comunicación con el ordenador y la cantidad de controladores dependen del número de motores a utilizar.



Además de manejar los controladores de los motores la mayoría de las controladoras para CNC tienen características adicionales, por ejemplo: entradas para finales de carrera, botón de emergencia, protecciones, etc.

#### - Diagrama de conexión de una controladora modular

En la Figura 1.4 se muestra los elementos que forma parte de un sistema con una controladora modular.

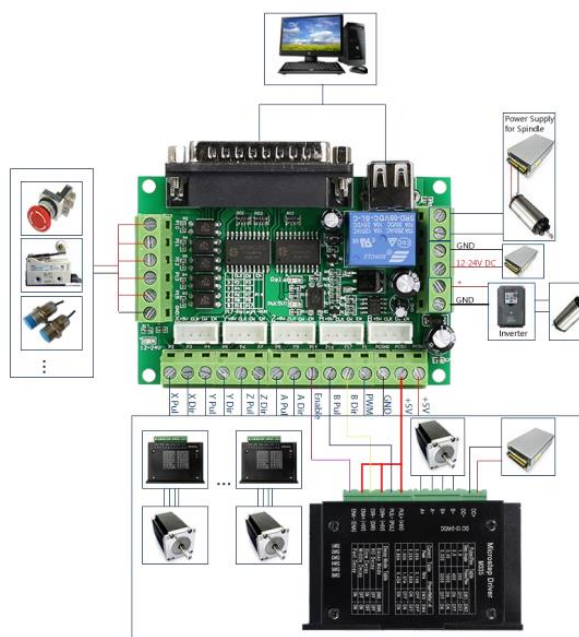


Figura 1.4. Diagrama de conexión de una controladora para CNC Modular [14].

Dentro de las controladoras modulares existe un subgrupo, donde la controladora se conecta a una placa electrónica. Es llamada Shield, el tipo dependerá de la placa electrónica que se vaya a usar. La placa electrónica hace el trabajo del ordenador, convirtiendo los comandos en secuencia de accionamiento para las bobinas de los motores.

Existen varias placas electrónicas que pueden ser utilizadas en una máquina CNC, a continuación, se mencionan algunas:

## - Arduino



Figura 1.5. Placas Arduino [15].

Es una placa electrónica basada en hardware y software libre, lo que significa que tanto como los diagramas electrónicos y el código fuente son públicos y pueden ser modificados según la necesidad del proyecto [16]. La familia Arduino tiene una amplia gama de placas electrónicas como se puede apreciar en la Figura 1.5.

La placa Arduino puede interactuar con su entorno gracias a que posee entradas donde se puede conectar sensores de luz, temperatura, etc. Y de acuerdo a estos sensores se puede controlar elementos de salida, como leds, motores, entre otros actuadores. Arduino es muy versátil, ya que se puede añadir alguna característica a nuestro proyecto solo añadiendo el elemento deseado, también si se desea realizar otra aplicación, se lo puede reprogramar por completo e implementar los sensores y actuadores para cumplir con el nuevo objetivo.

Arduino tiene una amplia gama de placas que, dependiendo de la magnitud del proyecto, cada una tiene sus ventajas y desventajas, todo depende de las necesidades del proyecto.

## - Shield

Es diseñada para ir sobre la placa electrónica Arduino siendo un complemento, que adiciona características al Arduino, generalmente son de hardware libre y son fabricadas para una cumplir con una aplicación específica [17].

Una CNC shield puede manejar controladores de motores paso a paso, finales de carrera, husillo, z probe, entre otras características adicionales que dependerán del tipo de CNC shield. Las más usadas para la placa Arduino son:

### - Arduino CNC Shield v3

Diseñada para ser usada sobre cualquier placa Arduino, pero se recomienda usarlo sobre Arduino uno o Arduino Leonardo, se utiliza generalmente para el desarrollo de máquinas CNC para grabado e impresoras 3D. Nos permite controlar hasta 4 motores, soporta la conexión de finales de carrera, relés y otros sensores [18].

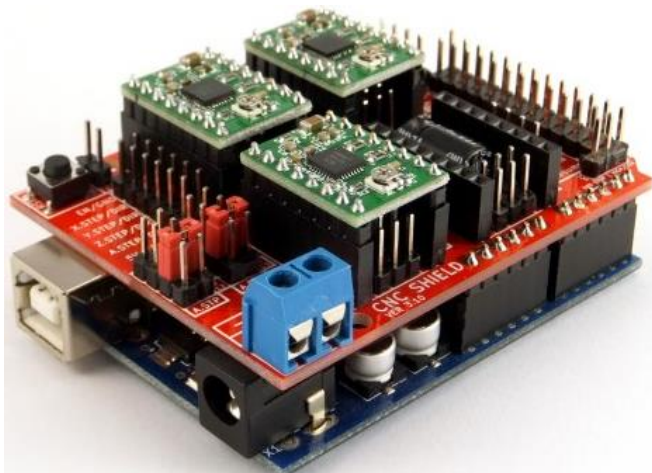


Figura 1.6. CNC Shield v3 [18].

La Figura 1.6 muestra la CNC shield montada sobre Arduino UNO. Las principales características son:

- Soporte para 4 ejes (X, Y, Z, A)
- 2 conexiones para finales de carrera para cada eje (6 en total)

- Salida "Spindle enable" y "direction"
- Salida "Coolant enable"
- Compatible con Pololu A4988 y DRV8825
- Jumpers para control de micro-stepping (Los controladores como el DRV8825 soportan hasta 1/32 para más precisión)
- Diseño compacto
- Los motores pueden ser conectados con bornes tipo Molex de 4 pines
- Alimentación: 12-36V DC. (Dependiendo de los controladores utilizados) [18].

#### - RAMPS 1.4

Diseñada para ser usada exclusivamente sobre el Arduino Mega, generalmente se la utiliza para el desarrollo de impresoras 3D con capacidad de manejar hasta 2 extrusores, pero en actualizaciones recientes se incrementó su versatilidad permitiendo manejar un husillo para el desarrollo de una fresadora, también puede controlar laser [19].

Como se muestra en la Figura 1.7 la shield RAMPS tiene un mayor tamaño que su análoga para Arduino Uno, esto implica que posee más puertos de entrada/salida que pueden ser usados para conectar pantallas lcd, sensores de temperatura, servomotores, entre otras características adicionales.

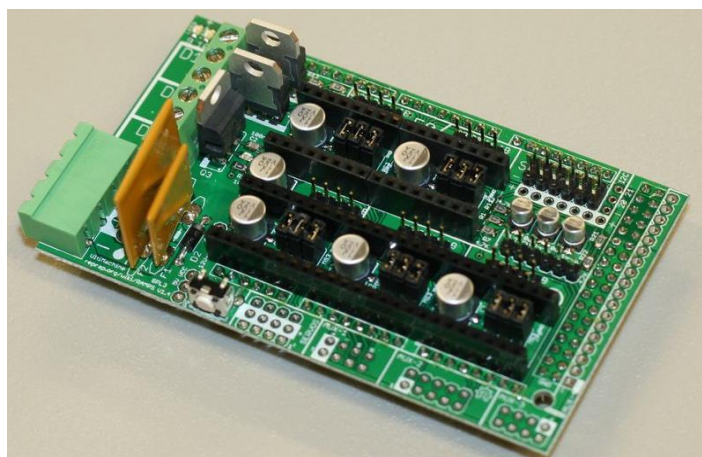


Figura 1.7. RAMPS 1.4 [20].

Las características de esta shield son las siguientes:

- Se conecta al Arduino Mega 2560
- Compatible con los controladores para motores stepper Pololu A4988 /DRV8825. Puedes conectar hasta 5 controladores.
- Tres salidas de potencia con MOSFETs Toshiba (cama caliente, extrusor, ventilador, o doble extrusor)
- Todas las salidas de potencia están protegidas por Diodos Schottky
- Permite doble Extrusor
- Protegida por un Fusible reseteable de 5A
- Circuito independiente de 11A protegidos por otro fusible reseteable, para la cama caliente
- Pines para 3 termistores
- Pines para 6 límites
- Salida i2c y SPI
- Posibilidad de conectar una pantalla LCD
- Salidas para conexiones adicionales y servos
- Puedes conectar 2 motores para la Z directamente en la placa [21].

#### - **Raspberry Pi**

La Raspberry Pi es un ordenador de bajo coste y tamaño reducido, tanto es así que cabe en la palma de la mano, pero puedes conectarle un monitor y un teclado para interactuar con ella exactamente igual que cualquier otra computadora [22].

También posee puertos de entrada/salida para conectar sensores o actuadores lo que la hace ideal para realizar proyectos de electrónica, robótica, domótica y ahora con la ayuda de una shield también puede utilizarse para realizar máquinas CNC.

En la Figura 1.8 se muestra la Shield CNC compatible con Raspberry Pi.

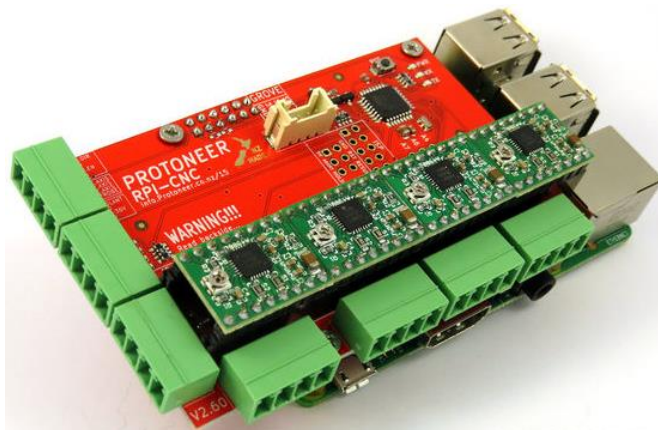


Figura 1.8. Raspberry Pi CNC Shield [23].

Las características principales de la Raspberry Pi CNC Shield son:

- Compatible con GRBL V1.1. (Firmware de código abierto que convierte los comandos de código G en señales de paso a paso)
- 3 ejes (XYZ) + 4º que puede clonar X / Y / Z
- Utiliza controladores paso a paso compatibles con Pololu A4988 extraíbles. (A4988, DRV8825 y otros)
- Utiliza conectores de tornillo extraíbles para que sea muy fácil de cablear.
- Final de parada para cada eje.
- Control de intensidad de husillo / láser de velocidad variable (pin de eje PWM y pin de dirección)
- Pin de control del refrigerante. (Pin de salida extra)
- Jumpers para establecer el Micro-Stepping para cada controlador paso a paso. (Algunos controladores como el DRV8825 pueden hacer hasta 1/32 micro pasos)
- Funciona con 12-30V DC. (Por el momento, solo los controladores Pololu DRV8825 pueden manejar hasta 36 V, así que tenga en cuenta el voltaje de funcionamiento cuando se enciende la placa).
- Tablero ensamblado listo para funcionar.
- Compatible con modelos de Raspberry Pi:
  - Raspberry Pi 3 y B +
  - Raspberry Pi 2
  - Raspberry Pi B + (El que tiene los encabezados de 40 pines) [24].

## - Controladora Integrada

Es una placa electrónica que contiene la interfaz, controladores y entradas/salidas para elementos adicionales incorporados, como se ilustra en la Figura 1.9, esto hace que los elementos puedan compartir la alimentación, a diferencia de las controladoras modulares no se necesita realizar conexiones, ocupan menos espacio, puede resultar más económico, estas características resultan ser muy ventajosas al momento de construir una máquina CNC [25].



Figura 1.9. Controladora Integrada [25].

Una de las principales desventajas que tiene este tipo de controladoras es que, si se necesita cambiar o actualizar una función, es necesario realizar el cambio completo del sistema ya que todos los elementos se encuentran integrados en la controladora.

Una controladora ya sea modular o integrada tiene al menos una interfaz, es el método que se utiliza para comunicarse con el ordenador. Las interfaces pueden ser opto acopladas lo que permite una mayor seguridad

Existen diversos tipos de interfaz a continuación se mencionan los más usados:

- **Puerto LTP:** También conocido como puerto paralelo, ha sido uno de los estándares más usados desde los años 70, en la actualidad sigue siendo usado por muchas placas controladoras para CNC, debido a su sencillez y su gran velocidad de comunicación [25]. Una de las mayores desventajas es que la mayoría de los ordenadores de escritorio actuales ya no son fabricados con estos puertos. Esto implica que se debe adquirir una

tarjeta externa que contenga este puerto, además, la configuración puede resultar un tanto complicada.

- **Puerto Serie:** También es un estándar muy usado y reconocido desde hace mucho tiempo, este tipo de comunicación permite una buena comunicación a una distancia relativamente grande [25].
- **Puerto USB:** El puerto USB actualmente está en auge, permite la comunicación a velocidades muy altas, con el tiempo será el estándar de comunicación, por ahora no todos los softwares de controladoras CNC son compatibles [25].

#### 1.4.2.2. Software

Cada controladora tiene un software específico para interpretar las instrucciones y convertirlas en señales eléctricas para el movimiento de los motores.

El software puede clasificarse en privado o cerrado y libre o abierto.

##### 1.4.2.2.1. Software Privado

Se refiere a cualquier programa informático en el que los usuarios tienen limitadas las posibilidades de usarlo, modificarlo o redistribuirlo (con o sin modificaciones, o cuyo código fuente no está disponible o el acceso a éste se encuentra restringido) [26].

Como se mencionó anteriormente las controladoras de hardware cerrado más usadas son las de Mach3, este mismo nombre lo lleva el software encargado de controlar a los motores, su uso es parcialmente licenciado, se puede usar en modo de prueba por tiempo indefinido, este modo tiene las opciones básicas para el control de la máquina CNC, pero para usar las opciones avanzadas es necesario adquirir la licencia.

Al necesitar que la controladora se conecte con el computador mediante un puerto paralelo, el software para el control de los motores paso a paso debe ser instalado sobre un computador.



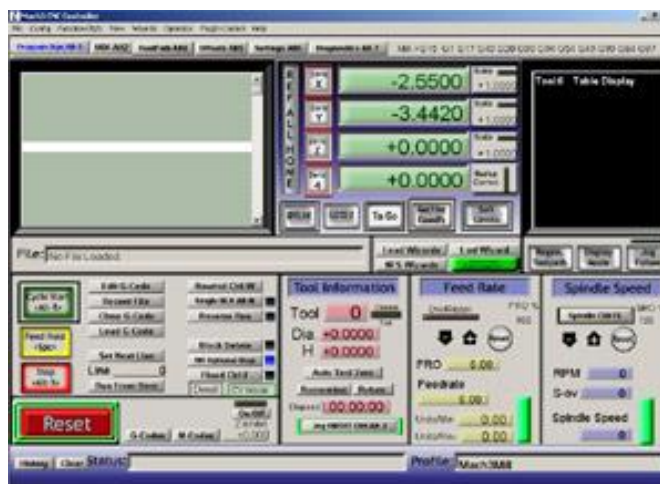


Figura 1.10. Software Match 3 [27].

En la Figura 1.10 se muestra la interfaz de Match3 en la cual podemos controlar los motores paso a paso, visualizar la posición, configurar la velocidad del husillo, la velocidad de avance, entre otras opciones. También es el encargado de enviar las instrucciones a la tarjeta controladora.

#### 1.4.2.2.2. Software Libre

Es llamado libre o abierto porque cualquier persona puede tener acceso a su código fuente, analizarlo y modificarlo de acuerdo a sus necesidades. El término libre no significa que sea gratuito, gran parte del software libre es gratis, debido a que es desarrollado por varias personas y subido a la nube [28].

Muchas veces hay colaboradores que modifican o añaden características a un software para una aplicación específica, y al ser dueños del nuevo código, están en todo el derecho si deciden dar gratuitamente las nuevas características.

En el ámbito de software CNC existe una gran variedad de programas que se usan para el control de los motores paso a paso, existe software que no es compatible con varias controladoras. A continuación, se muestran algunos de los programas de software libre usados para el control de motores paso a paso:

## - TurboCNC

Un programa que puede manejar hasta 8 ejes, se caracteriza por su interfaz sencilla y su alta precisión en los movimientos. Funciona en computadores de bajos recursos, uno de los mejores programas de software abierto, pero para acceder a todas sus características y a su código fuente es necesario desembolsar \$60, adquiriendo la licencia también se tiene acceso a soporte técnico [29]. Es necesario tener una conexión del computador con la controladora a través de un puerto serial. En la Figura 1.11 se muestra la interfaz que posee este software.



Figura 1.11. Interfaz TurboCNC [29].

## - LinuxCNC

Es un software para computador que permite el control de máquinas herramienta tales como fresadoras, tornos, robots tipo puma o Scara y cualquier otro tipo de máquina de hasta 9 ejes. LinuxCNC es software libre con un código fuente abierto [30].

Este programa trabaja sobre Ubuntu, está escrito en el lenguaje Python, además tiene varias interfaces para elegir, lo que lo hace más versátil, tiene soporte por parte de la comunidad. La comunicación con los motores paso a paso se la puede realizar mediante un puerto serial o una tarjeta PCI Express, cada una con su configuración respectiva.

La interfaz de este programa se puede visualizar en la Figura 1.12.

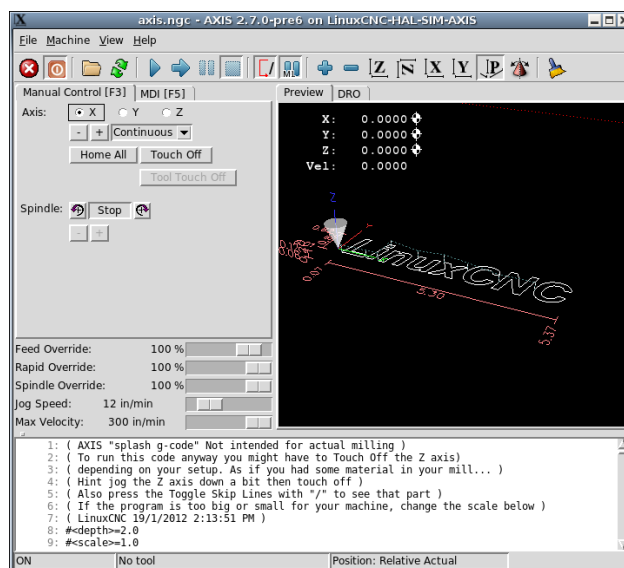


Figura 1.12 LinuxCNC Interfaz Axis [31].

## - cncControl

Es un programa diseñado para controlar máquinas CNC basadas en la plataforma Arduino, es capaz de controlar máquinas laser, el desplazamiento de los motores PAP mediante el mouse, teclado o incluso un gamepad, permite enviar instrucciones desde la interfaz, también se puede cargar y modificar un fichero con las instrucciones de código G. Para la comunicación con Arduino es necesario instalar un firmware en la placa y conectarlo a través de su puerto USB [32]. Su interfaz se muestra en la Figura 1.13.

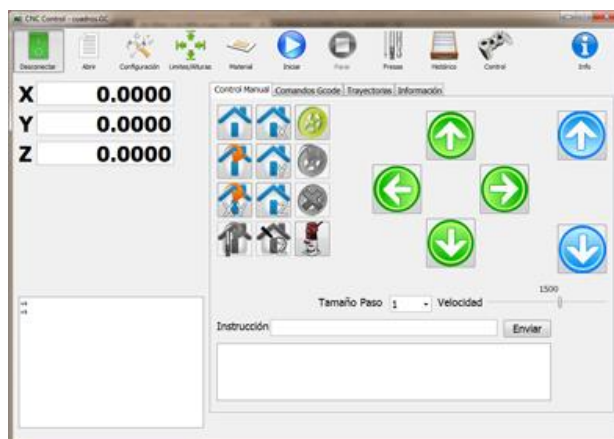


Figura 1.13. Interfaz cncControl [32].

cncControl es compatible con tres placas de la familia Arduino: Uno, Nano y Mega. La placa que se elija depende de las características que va a tener la máquina CNC.

## - Pronterface

Este software es parte de un paquete llamado Printron, es el encargado de controlar los motores PAP está desarrollado en Python, pero puede ser instalado tanto en Windows como en Mac Os, en su interfaz tiene un panel para el movimiento de los ejes, se puede enviar líneas de código mediante un editor de línea, se puede apagar los motores, establecer el avance de los motores en la parte inferior tiene un panel donde se pueden monitorizar sensores [33].

Pronterface está diseñado para el control de una impresora 3D, pero puede utilizarse para el control de una fresadora o máquina laser. A continuación, se muestra la interfaz de este programa en la Figura 1.14.

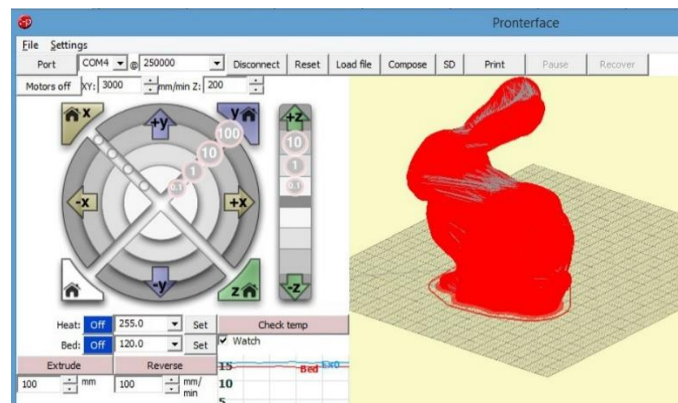


Figura 1.14. Interfaz Pronterface. [34]

Todos los programas mencionados anteriormente son los más usados, existen una gran variedad de programas similares, pero todos buscan cumplir el mismo objetivo; controlar una máquina CNC.

## 1.5. Interprete código G

El G-Code (o código G, en castellano) es el nombre de un lenguaje de descripción de operaciones para máquinas de control numérico por ordenador (CNC) que puede ser usado también como lenguaje de programación para controlar estos dispositivos para simplificar operaciones utilizando, por ejemplo, bucles [35].

Desde que se estandarizó en los 80 como ISO 6983, el G-Code es el lenguaje más utilizado en la fabricación por ordenador. Aunque existen ampliaciones e implementaciones de algunos

fabricantes, así como estándares alternativos al internacional en algunos países, es más que probable que cualquier máquina de control numérico moderna sepa interpretar el G-Code básico. El G-Code se almacena en formato texto, es decir, puede leerse (y modificarse) con un editor de texto plano, aunque lo más habitual es que se genere y se visualice desde una aplicación de modelado y/o fabricación 3D o alguna herramienta o accesorio específico [36].

El código G es un lenguaje de programación de alto nivel en la que se trabaja con coordenadas cartesianas, pero debe ser transformado en instrucciones para que los motores PAP realicen el desplazamiento deseado, de esta parte se encarga el intérprete de código G, toma cada una de las líneas de código y las transforma en la secuencia de accionamiento para que el motor PAP se mueva a la posición establecida, después de ser transformado la secuencia de accionamiento es enviada a los controladores para que se realice el movimiento.

A continuación, se muestra en la

Tabla 1-I los comandos de código G más usados:

*Tabla 1-I. Tabla descriptiva de los códigos G [37].*

<b>Comando</b>	<b>Descripción</b>
<b>N</b>	Número de Secuencia
<b>G</b>	Funciones Preparatorias
<b>X</b>	Comando para el Eje X
<b>Y</b>	Comando para el Eje Y
<b>Z</b>	Comando para el Eje Z
<b>R</b>	Radio desde el Centro Especificado
<b>A</b>	Ángulo contra los Punteros del Reloj desde el Vector +X
<b>I</b>	Desplazamiento del Centro del Arco del Eje X
<b>J</b>	Desplazamiento del Centro del Arco del Eje Y
<b>K</b>	Desplazamiento del Centro del Arco del Eje Z
<b>F</b>	Tasa de Alimentación
<b>S</b>	Velocidad de Giro
<b>T</b>	Número de Herramienta
<b>M</b>	Funciones Misceláneas

Los programas para el control de una máquina CNC que son instalados en un computador tienen integrados el intérprete en su código fuente, en cambio, las controladoras conformadas por placas electrónicas como Arduino, necesitan que se cargue el firmware, además, de tener el intérprete también posee varias subrutinas para la configuración tanto del hardware como del software.

Existe una infinidad de firmwares que pueden ser cargados a las placas, el que se elija depende de la necesidad que se necesite cubrir, pues existen varios tipos dependiendo de la aplicación, por ejemplo, para una impresora 3D, CNC Mill, etc.

## CAPÍTULO II

### 2. METODOLOGÍA

La construcción de la estructura mecánica de la máquina CNC será realizada a partir de los planos obtenidos del trabajo de pregrado llamado “Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Diseño”.

#### 2.1. Despliegue de la función de calidad (QFD)

Es un método globalizador cuyo objetivo principal es asegurar que se tiene en cuenta la voz del usuario o cliente, a la vez que constituye una ayuda para la planificación de la calidad durante todo el ciclo de vida [38].

El primer paso para desarrollar la casa de la calidad es conocer la voz del usuario, este objetivo se lo puede cumplir de diferentes formas, una de ellas es dialogar directamente con la(s) persona(s) a la cual va dirigido el producto. En la conversación se le pide al usuario que manifieste cuales son los requerimientos que satisfagan sus necesidades y que posteriormente por medio del criterio del ingeniero serán transformados a requerimientos técnicos medibles para ser evaluados y determinar los requerimientos más importantes.

##### 2.1.1. Fijación del objetivo

¿Qué características debe reunir una máquina CNC para corte de metal?

##### 2.1.2. Voz del usuario

Después de haber realizado una reunión con el usuario se realizó una lista en las cuales se plasman los requerimientos con los que debe contar la máquina.

- Que no falle en las medidas. (5) [Desempeño]
- Que sea capaz de cortar planchas de aluminio. (5) [Usabilidad]
- Que el posicionamiento del cero pieza sea sencillo. (3) [Usabilidad]
- Que sea de bajo costo. (4) [Deseable]

- Que no se choque la torre con la estructura. (5) [Desempeño]
- Que el corte sea rápido. (3) [Desempeño]
- Que use software y hardware libre. (3) [Deseable]
- Que sea segura. (4). [Desempeño]
- Que sea de fácil operación. (4) [Usabilidad]
- Que sea fácil de construir. (2) [Deseable]
- Que sea de fácil mantenimiento. (3) [Desempeño]

Al obtener los parámetros establecidos por el usuario, se pondera en un rango de 0-5 los requerimientos según la importancia, además, se clasifico las características en 3 grupos (Usabilidad, Desempeño y Deseable), las características de Usabilidad son aquellas que hacen que la interacción entre el operario y la máquina sea lo más fácil posible.

Las características de desempeño son aquellas que hacen que la máquina cumpla con su objetivo. Y por último las características deseables son aquellas que no influyen en el desempeño de la máquina, pero generan un sentimiento de satisfacción en el usuario, y en caso de que no se cumplan no generan insatisfacción.

Una vez ponderados y clasificados los requerimientos obtenidos del usuario se procede a traducirlos en características técnicas con el criterio de ingeniería, una vez transformadas estas características son medibles.

### 2.1.3. Voz del ingeniero

- Precisión [mm].
- Potencia de motor husillo [W].
- Tiempo de encerar la máquina [s]
- Selección de materiales [USD].
- Tiempo de respuesta de protecciones. [s].
- Tiempo de corte [s].
- Valor de adquisición del controlador. [USD]
- Porcentaje de falla. [%]



- Comunicación bidireccional (HMI). [capacitación del operario].
- Planos de construcción detallados.
- Piezas estandarizadas, disponibles [uniones y partes funcionales]  
en el mercado local.

Al terminar de transformar los requerimientos de la voz del ingeniero a la voz del ingeniero se procede a ubicarlos en la matriz del QFD. Una vez puesto los requerimientos tanto del usuario como del ingeniero, en la matriz se ubican los pesos de acuerdo a su importancia y la relación que tienen con respecto a los demás requerimientos. También se establece si existe correlación entre las características técnicas. Se realiza una valoración de cada uno de los requerimientos tomando como referencia la máquina a realizar y máquinas existentes.



En la Figura 2.1 se muestra el diagrama de la función de la calidad, en el cual se encuentran los parámetros técnicos establecidos a partir de la voz del usuario, su importancia, la relación existente entre ellos, los objetivos que se buscan alcanzar, la dificultad que implica alcanzarlos.

### **2.1.5. Conclusiones de la casa de la calidad**

Como resultado la matriz QFD muestra el nivel importancia que tiene cada uno de los requerimientos, se deben tomar en consideración aquellos que posean el mayor porcentaje, debido a que, estos son de vital importancia para el desarrollo de la máquina.

A continuación, se muestran los requerimientos con mayor porcentaje de importancia:

- Selección de materiales (Bajo costo) con un porcentaje del 18,7%.
- Respuesta de las Protecciones (Evitar choque de torre con la estructura) con un 17,8%.
- Tiempo de encerar la máquina (Posicionamiento de cero pieza sencillo) con un 13,9%.

El diseño de la máquina debe basarse en estos tres requerimientos, los cuales serán considerados como principales.

## **2.2. Conceptualización**

### **2.2.1. Análisis funcional**

El análisis funcional es técnica propuesta por el ingeniero estadounidense Lawrence D. Miles, cuyo propósito es el de separar la acción que se efectúa del componente o mecanismo, para de este modo buscar nuevas soluciones a un mismo problema. El análisis funcional logra obtener mejores productos a un menor costo [39].

En esta sección se procede a definir cuáles son las funciones primarias y secundarias de la máquina, esto permite diferenciar las funciones que son necesarias para que se cumpla el objetivo (primarias) y las funciones que ayudan a cumplir satisfactoriamente el objetivo.

La función primaria de la máquina es realizar corte por Control Numérico Computacional de planchas de aluminio.

### 2.2.2. Desarrollo de los diagramas funcionales

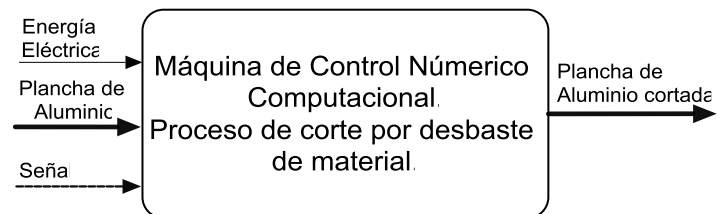


Figura 2.2. Diagrama Funcional, Nivel 0.

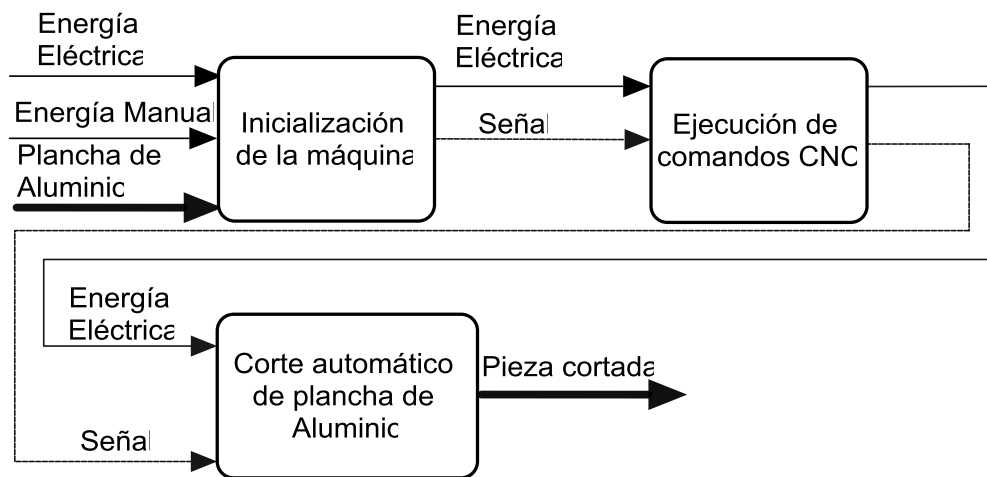


Figura 2.3. Diagrama Funcional, Nivel 1

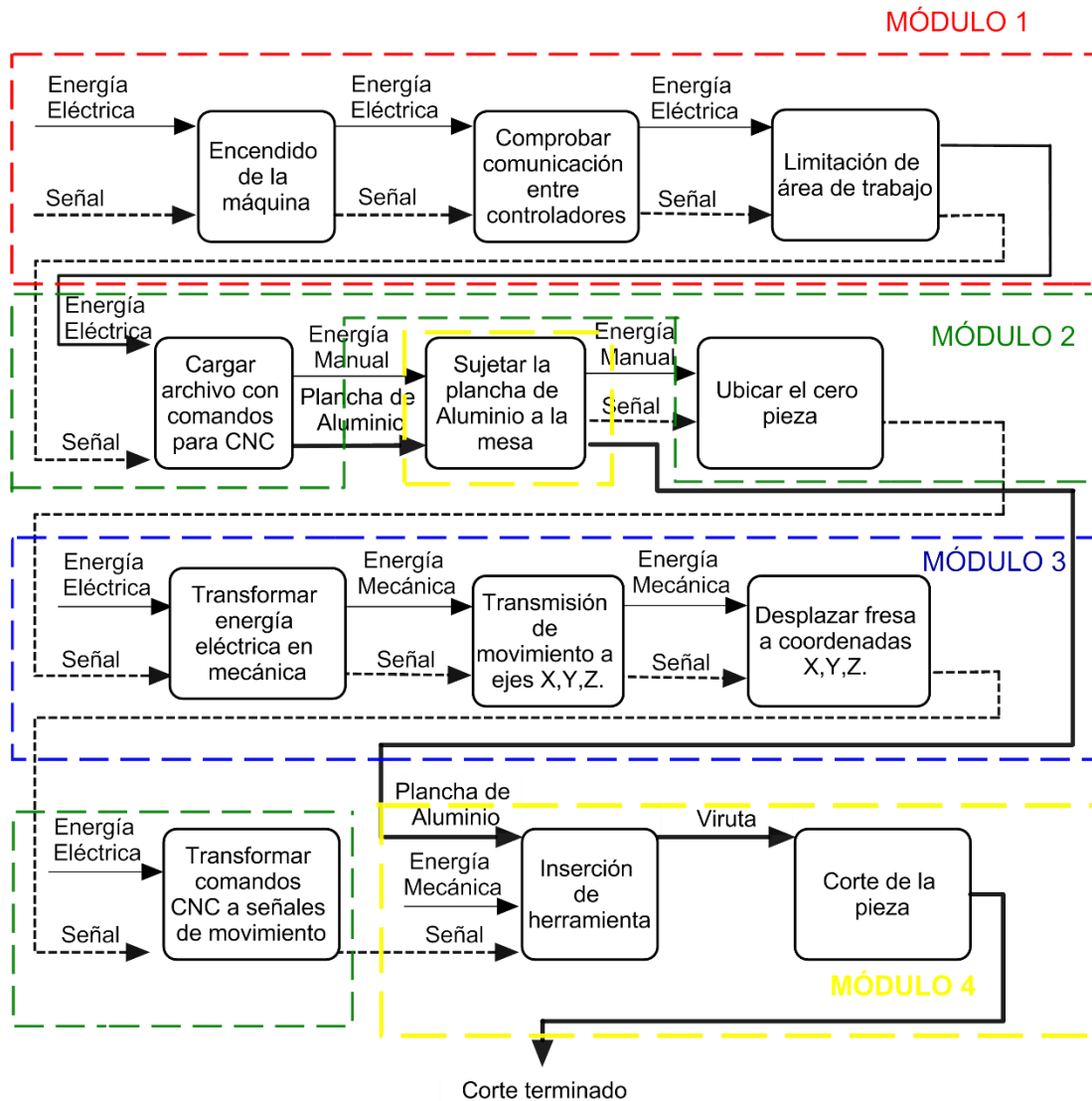


Figura 2.4. Diagrama Funcional, Nivel 2 y módulos.

### 2.2.3. Análisis de los diagramas funcionales

En el diagrama funcional nivel 0 mostrado en la Figura 2.2, se presenta la función primaria, realizar corte de planchas de aluminio, para ello se necesita de material, energía manual y señales de control que serán generadas por los controladores.

En la Figura 2.3 se muestra el diagrama funcional de nivel 1, aquí existen más funciones, estas especifican de manera general los procesos que se realizarán para obtener el corte de la plancha de aluminio.

En el diagrama funcional de nivel 2, mostrado en Figura 2.4, existen las funciones mucho más detalladas, el análisis de cada una de las funciones es fundamental para el desarrollo de la máquina, para el cumplimiento de varias funciones es necesario la intervención de la energía manual.

#### **2.2.4. Solución de los módulos**

Para establecer los diferentes módulos, se analiza el diagrama funcional nivel 2, ilustrado en la Figura 2.4, la división se ha realizado separando la parte mecánica de la electrónica. Estas divisiones modulares permiten analizar cada uno de los procedimientos que realiza la máquina para cumplir su objetivo.

En este trabajo se realiza el análisis de los módulos 1 y 2, que comprenden a la parte electrónica.

#### **2.2.5. Módulo 1**

Este módulo contiene las siguientes funciones:

- Encendido de la máquina
- Comprobar la comunicación entre controladores
- Limitación del área de trabajo.

A continuación, se proponen posibles soluciones para cada una de las funciones de este módulo y establecer diferentes alternativas de solución, evaluarlas y determinar cuál de ellas es la mejor para cumplir con el objetivo planteado.

##### **2.2.5.1. Encendido de la máquina**

La mejor manera de realizar esta función es con un switch que se encuentra ilustrada en la Figura 2.5, sin importar el tipo, pues la función de este componente es dejar pasar la energía eléctrica hacia los diferentes componentes de la máquina.



Figura 2.5. Switch (Pulsador) [40].

### 2.2.5.2. Comprobar la comunicación entre controladores

Existen muchos tipos de comunicación por los cuales se puede enviar y recibir información de un controlador a otro, cada tipo tiene sus ventajas, cada uno está diseñado para trabajar en condiciones específicas.

#### A. Comunicación Serial

Es un protocolo de comunicación estándar, en el cual se envía y recibe información binaria, además esta se transmite bit a bit. En la Figura 2.6 se muestra como es enviada la información bajo este tipo de comunicación. Las comunicaciones serial proporcionan una forma rápida y sencilla de que un microcontrolador interactúe con el ordenador u otros dispositivos. También es una herramienta útil para depurar el código [41].

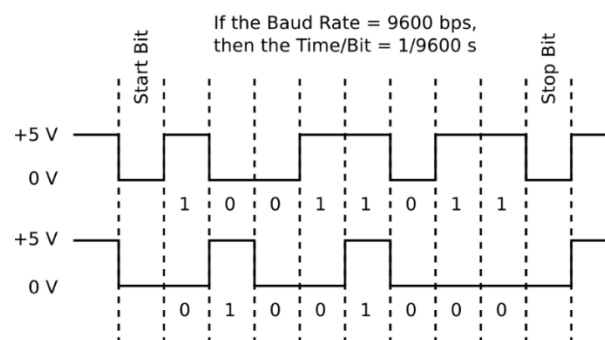


Figura 2.6. Comunicación Serial [41].

#### Ventajas

- La mayoría de los controladores y computadores tienen puertos de comunicación serial.

- Necesita menos líneas de transmisión que la comunicación en paralelo para transmitir la misma información.
- Los controladores al tener puertos de comunicación serial no necesitan circuitos extra para transmitir información.
- Es más sencillo de implementar que otros métodos.

### Desventajas

- La información se transmite bit a bit, por lo que se necesita una frecuencia más alta para enviar los mismos datos que la comunicación paralela.
- Si se incrementa la velocidad de transmisión la distancia eficiente disminuye.

## B. Comunicación en paralelo

Es un protocolo de comunicación que permite transmitir información de un dispositivo a otro, este tipo de comunicación envía paquetes de información, generalmente de 8 bits (1 byte) como se muestra en la Figura 2.7, haciendo que la transmisión de información sea rápida [42].

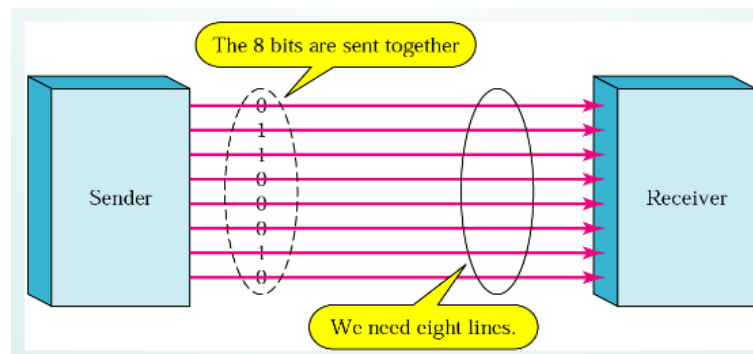


Figura 2.7. Comunicación en Paralelo [42].

### Ventajas

- Se puede enviar una mayor cantidad de información que la comunicación serial.
- Al enviar una mayor cantidad de información es más rápida que la comunicación serial.



## Desventajas

- Su implementación es más costosa que la comunicación serial.
- Debe tener el mismo número de líneas que la cantidad de información a enviarse.
- Puede saturar el canal de transmisión debido a la gran cantidad de información.
- Dependiendo del tipo de controlador puede necesitar un circuito externo.

### 2.2.5.3. Limitación del área de trabajo

Se divide en límites establecidos por hardware y software. A continuación, se analiza la limitación del área de trabajo por hardware.

Es una función muy importante, porque brinda protección a los sistemas mecánicos de sufrir choques o accidentes en el proceso de corte. Además de la limitación mecánica se puede realizar la configuración del área de trabajo a nivel de software.

#### A. Final de carrera

Es un sensor electromecánico que posee 2 configuraciones, normalmente cerrado (NC) y normalmente abierto (NA), detecta la posición de un elemento móvil mediante accionamiento mecánico [43]. En la Figura 2.8 se muestra un final de carrera.



Figura 2.8. Final de Carrera [44].

## Ventajas

- Fácil instalación.
- Es capaz de trabajar en altos rangos de voltaje.
- Es inmune a corrientes parasitas.

## Desventajas

- Si se trabaja con mecanismos de alta velocidad, su detección puede resultar a destiempo.
- Se puede presentar rebotes de señal.
- Si su configuración es NA y existe alguna falla mecánica o de señal, el sensor no detectará.

## B. Sensor de presencia (magnético)

Detecta los campos magnéticos que provocan los imanes o las corrientes eléctricas. El principal es el llamado interruptor Reed; consiste en un par de láminas metálicas de materiales ferromagnéticos metidas en el interior de una cápsula que se atraen en presencia de un campo magnético, cerrando el circuito [45]. En la Figura 2.9 se muestra el sensor de presencia magnético.



Figura 2.9. Sensor de presencia magnético [45].

## Ventajas

- Puede detectar la presencia de un objeto sin necesidad de tener contacto físico.
- Resistente a la humedad y polvo.
- Versatilidad de manejar varios rangos de voltaje.

## Desventajas

- Los campos magnéticos que se encuentren alrededor pueden afectar a su respuesta.
- Su implementación debe ser realizada de forma cuidadosa, para que no exista elementos que puedan interferir en su medición.

## C. Sensor de presencia (inductivo)

Un sensor de proximidad inductivo puede detectar objetos metálicos que se acercan al sensor, sin tener contacto físico con los mismos. Los sensores de proximidad inductivos se clasifican en los siguientes tres tipos, de acuerdo con su principio de funcionamiento: el tipo de oscilación de alta frecuencia que utiliza la inducción electromagnética; el tipo magnético que emplea un imán; y el tipo de capacitancia que aprovecha los cambios en la capacidad eléctrica [46]. Este sensor es ilustrado en la Figura 2.10.

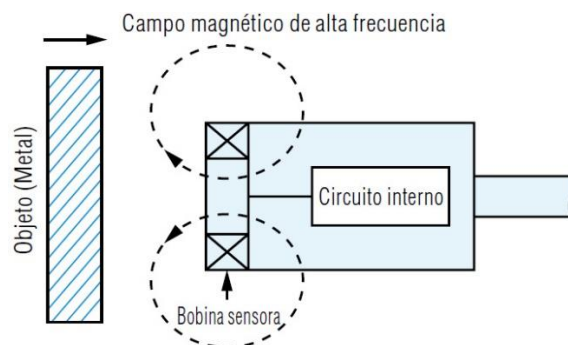


Figura 2.10. Sensor de proximidad inductivo [46].

## Ventajas

- No necesita contacto físico para su activación.
- Tiene un alto grado de resolución y sensibilidad
- Fácil integración con sistemas industriales.

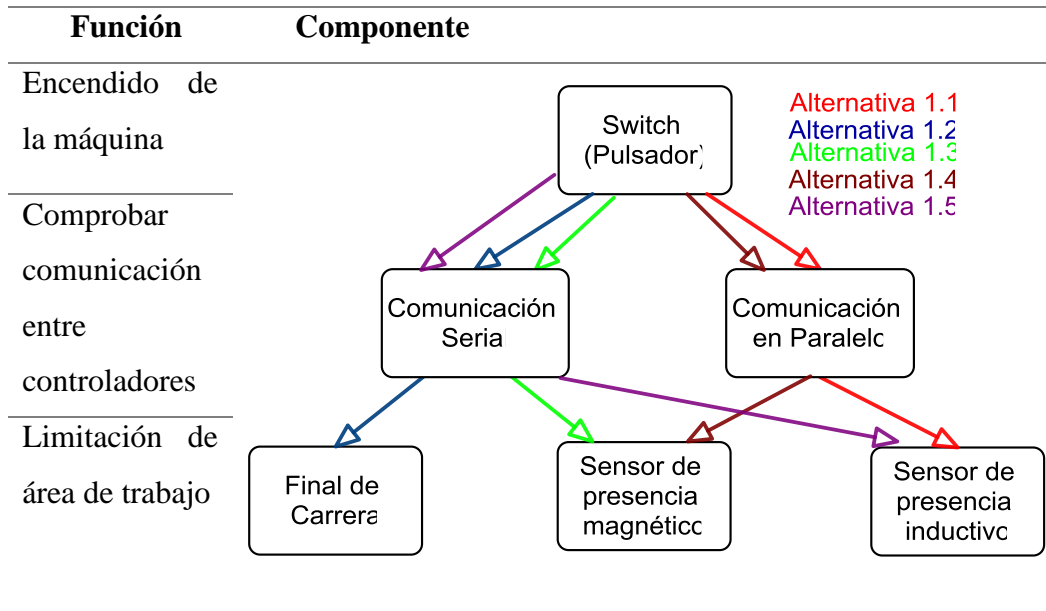
## Desventajas

- Su resolución puede verse comprometida si existen resistencias parasitas.
- Los campos magnéticos pueden afectar su funcionamiento.

### 2.2.5.4. Combinación de las soluciones del Módulo 1.

Las posibles soluciones para el Módulo 1 se muestran en la Tabla 2-I

Tabla 2-I. Soluciones para el módulo 1.



Las diferentes alternativas para solución de las funciones del Módulo 1 son las siguientes:

- **Alternativa 1.1.-** Un switch junto a comunicación en paralelo y sensor de presencia inductivo.
- **Alternativa 1.2.-** Un switch junto a comunicación serial y final de carrera.
- **Alternativa 1.3.-** Un switch junto a comunicación serial y sensor de presencia magnético.
- **Alternativa 1.4.-** Un switch junto a comunicación en paralelo y sensor de presencia magnético.
- **Alternativa 1.5.-** Un switch junto a comunicación serial y sensor de presencia inductivo.

### 2.2.5.5. Solución para el módulo 1.

Para determinar la solución más factible se aplicará el método ordinal corregido de criterios ponderados.

En la siguiente lista se encuentra los criterios de evaluación más importantes:

- **Velocidad de respuesta:** Las máquinas de Control Numérico Computacional (CNC) trabajan con comandos que son ingresados por parte del usuario, trabajan con rangos de tolerancia muy pequeñas, debido a esto, la respuesta ante cualquier inconveniente que pueda presentarse debe ser lo más rápida posible.
- **Precio:** La máquina al estar dirigida a la pequeña industria su diseño y construcción deben tener un costo accesible.
- **Fácil operación:** La máquina deber ser lo más fácil de usar, para que el operador pueda realizar trabajos de corte sin una capacitación exhaustiva.
- **Fácil mantenimiento:** Para tener un rendimiento óptimo de la máquina esta debe ser de fácil mantenimiento, ya sea preventivo o correctivo, que el usuario pueda realizarlo sin mucho conocimiento técnico, si se necesita mantenimiento correctivo, es necesario que las piezas a reemplazar sean fáciles de encontrar en el mercado local.
- **Fiabilidad:** Al ser una máquina automática, debe tener un alto rendimiento y reducir al mínimo los paros innecesarios.

Después de haber establecidos los criterios para la evaluación de las diferentes alternativas, se realiza una ponderación de cada uno, para establecer el nivel de importancia en el desarrollo del diseño de la máquina. En la Tabla 2-II se realiza la ponderación de los criterios establecidos para la evaluación del módulo 1, cada uno de los criterios son evaluados de acuerdo con su importancia.

Tabla 2-II. Ponderación de los criterios para el módulo 1.

Criterio	Respuesta	Precio	Operación	Fiabilidad	Mantenimiento	$\Sigma+1$	Ponderación
<b>Respuesta</b>		0,5	0,5	0,5	1	3,5	0,2414
<b>Precio</b>	0,5		1	0,5	1	4	0,2759
<b>Operación</b>	0,5	0		0	0,5	2	0,1379
<b>Fiabilidad</b>	0,5	0	1		0,5	3	0,2069
<b>Mantenimiento</b>	0	0	0,5	0,5		2	0,1379
					Suma	14,5	1

*Precio > Respuesta > Fiabilidad > Operación = Mantenimiento*

A continuación, se realiza la evaluación de las diferentes alternativas respecto a cada uno de los criterios. En la Tabla 2-III se muestra el análisis de acuerdo al criterio Velocidad de respuesta.

Tabla 2-III. Evaluación de las alternativas, de acuerdo al criterio de Velocidad de Respuesta.

<b>Velocidad Respuesta</b>	Alternativa 1	Alternativa 2	Alternativa 3	Alternativa 4	Alternativa 5	$\Sigma+1$	Ponde- ración
Alternativa 1		1	0,5	0,5	1	4	0,2667
Alternativa 2	0		1	0	1	3	0,2000
Alternativa 3	0,5	0		0	0,5	2	0,1333
Alternativa 4	0,5	1	1		1	4,5	0,3000
Alternativa 5	0	0	0,5	0		1,5	0,1000
					Suma	15	1

***Alternativa 4 > Alternativa 1 > Alternativa 2 > Alternativa 3 > Alternativa 5***

El criterio que se va a evaluar a continuación es el precio, mostrado en la Tabla 2-IV.

Tabla 2-IV. Evaluación de las alternativas, de acuerdo al criterio: Precio.

<b>Precio</b>	Alternativa 1	Alternativa 2	Alternativa 3	Alternativa 4	Alternativa 5	$\Sigma+1$	Pondera- ción
Alternativa 1		0	0	0,5	0	1,5	0,1000
Alternativa 2	1		1	1	1	5	0,3333
Alternativa 3	1	0		0	0,5	2,5	0,1667
Alternativa 4	0,5	0	1		0	2,5	0,1667
Alternativa 5	1	0	0,5	1		3,5	0,2333
					Suma	15	1

***Alternativa 2 > Alternativa 5 > Alternativa 3 = Alternativa 4 > Alternativa 1***

El siguiente criterio que se va a evaluar es la fácil operación. Se muestra en la Tabla 2-V.

Tabla 2-V. Evaluación de las alternativas, de acuerdo al criterio: Fácil Operación.

<b>Fácil Operación</b>	Alternativa 1	Alternativa 2	Alternativa 3	Alternativa 4	Alternativa 5	$\Sigma+$	Ponderación
Alternativa 1		0	0	0,5	0	1,5	0,1000
Alternativa 2	1		1	1	0,5	4,5	0,3000
Alternativa 3	1	0		1	0,5	3,5	0,2333
Alternativa 4	0,5	0	0		0	1,5	0,1000
Alternativa 5	1	0,5	0,5	1		4	0,2667
					Suma	15	1

**Alternativa 2 > Alternativa 5 > Alternativa 3 > Alternativa 4 = Alternativa 1**

La fiabilidad es el siguiente parámetro evaluado en la Tabla 2-VI.

Tabla 2-VI. Evaluación de las alternativas, de acuerdo al criterio de Fiabilidad.

<b>Fiabilidad</b>	Alternativa 1	Alternativa 2	Alternativa 3	Alternativa 4	Alternativa 5	$\Sigma+1$	Ponderación
Alternativa 1		0,5	0,5	1	0,5	3,5	0,2333
Alternativa 2	0,5		1	1	0,5	4	0,2667
Alternativa 3	0,5	0		0,5	0	2	0,1333
Alternativa 4	0	0	0,5		0	1,5	0,1000
Alternativa 5	0,5	0,5	1	1		4	0,2667
					Suma	15	1

**Alternativa 2 = Alternativa 5 > Alternativa 1 > Alternativa 3 = Alternativa 4**

En la Tabla 2-VII, se evalúa el criterio del mantenimiento.

Tabla 2-VII. Evaluación de las alternativas, de acuerdo al criterio de Mantenimiento.

<b>Mantenimiento</b>	Alternativa 1	Alternativa 2	Alternativa 3	Alternativa 4	Alternativa 5	$\Sigma+1$	Ponderación
Alternativa 1		0	0	0,5	0,5	2	0,1333
Alternativa 2	1		1	1	1	5	0,3333
Alternativa 3	1	0		1	0,5	3,5	0,2333
Alternativa 4	0,5	0	0		0	1,5	0,1000
Alternativa 5	0,5	0	0,5	1		3	0,2000
					Suma	15	1

***Alternativa 2 > Alternativa 3 > Alternativa 5 > Alternativa 1 > Alternativa 4***

Después de realizar la evaluación de las alternativas con los criterios, se establece un análisis de los resultados para establecer cuál es la mejor solución para el módulo 1.

Tabla 2-VIII. Tabla de ponderación de los resultados

<b>Resultados</b>	Respuesta	Precio	Operación	Fiabilidad	Mantenimiento	$\Sigma$	Prioridad
Alternativa 1	0,24x0,26	0,28x0,10	0,14x0,10	0,21x0,23	0,14x0,13	0,17	<b>3=4=5</b>
Alternativa 2	0,24x0,20	0,28x0,33	0,14x0,30	0,21x0,27	0,14x0,33	0,28	<b>1</b>
Alternativa 3	0,24x0,13	0,28x0,17	0,14x0,23	0,21x0,13	0,14x0,23	0,17	<b>3=4=5</b>
Alternativa 4	0,24x0,30	0,28x0,17	0,14x0,10	0,21x0,10	0,14x0,10	0,17	<b>3=4=5</b>
Alternativa 5	0,24x0,10	0,28x0,23	0,14x0,27	0,21x0,27	0,14x0,20	0,21	<b>2</b>

Después de haber realizado la evaluación de los resultados, la Tabla 2-VIII muestra que la mejor solución para el módulo 1 es la Alternativa 2.

### 2.2.6. Módulo 2

El módulo 2 contiene las siguientes funciones:

- Cargar archivos con comandos CNC
- Ubicar cero pieza.
- Transformar comandos CNC a señales de movimiento



Al igual que en el módulo 1 se procede a establecer las posibles soluciones para las funciones del módulo 2.

### 2.2.6.1.Cargar archivos con comandos CNC

Esta función es de suma importancia, debido a que, los comandos CNC tienen que ser enviados de un controlador principal hacia el controlador CNC, encargado de transformar estos comandos en señales de movimiento. A continuación, se muestra las alternativas para realizar esta función.

#### A. Computadora

Sistema digital con tecnología microelectrónica capaz de procesar datos a partir de un grupo de instrucciones denominado programa [47]. Está conformada por hardware y software, que permite al usuario interactuar con ella, es capaz de recibir información, procesarla y mostrar los resultados al usuario mediante periféricos de entrada/salida. Posee puertos USB por los cuales es capaz de enviar información a diversos dispositivos externos.



*Figura 2.11. Computador [47].*

#### Ventajas

- Enviar la información a dispositivos externos es fácil.
- Se puede crear los comandos CNC y enviarlos sin tener dispositivos intermedios.
- Se puede instalar software específico para monitorear el estado de la máquina CNC.

## Desventajas

- Costo elevado de adquisición.
- Si el controlador que convierte los comandos CNC en señales de movimiento es de licencia pagada, se necesita comprar la licencia para instalar y usar el software para el computador.

## B. Microcontrolador

Es un circuito integrado digital que puede ser usado para muy diversos propósitos debido a que es programable. Está compuesto por una unidad central de proceso (CPU), memorias (ROM y RAM) y líneas de entrada y salida (periféricos) [48]. En la Figura 2.12 se muestra algunos microcontroladores.

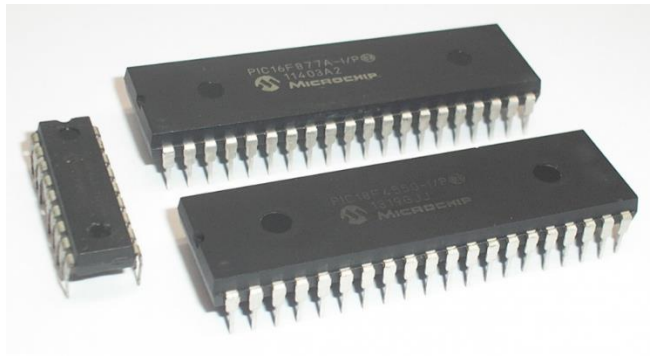


Figura 2.12. Microcontrolador [48].

## Ventajas

- Su costo de adquisición es bajo.
- Se puede enfocar a una aplicación específica.
- Su implementación se puede realizar en espacios reducidos.
- Tiene una gran flexibilidad, debido a que se puede expandir el sistema añadiendo instrucciones.

## Desventajas

- Se necesita conocimiento de lenguajes de programación.

- Necesita de un computador y software específico para realizar la programación.
- Necesita de un grabador para transferir el programa del computador al microcontrolador.
- Si se desea realizar aplicaciones complejas necesita de hardware extra.

### C. Sistema Embebido

Es un sistema electrónico diseñado para realizar funciones específicas en tiempo real. Al contrario de lo que ocurre con las computadoras, las cuales tienen un propósito general, ya que están diseñadas para cubrir un amplio rango de necesidades y los Sistemas Embebidos se diseñan para cubrir necesidades específicas [49].

En la Figura 2.13 se muestra un sistema embebido.



Figura 2.13. Sistema Embebido [49].

#### Ventajas

- Costo de adquisición bajo.
- Posee una gran versatilidad al poseer varios módulos a parte del microcontrolador.
- Existen sistemas embebidos libres.
- Fácil transmisión del programa del computador al sistema embebido.

#### Desventajas

- Se necesita conocimiento sólidos en el área de programación.
- Necesita de un computador para programar el sistema embebido

### 2.2.6.2. Ubicar el cero pieza

Para ubicar el cero pieza es necesario enviar señales a los motores para que la herramienta se desplace sin necesidad de cargar un archivo. Esta función es importante porque antes de realizar el proceso de corte es necesario que la herramienta se encuentre lo más cerca posible la materia prima. A continuación, se muestran posibles soluciones para cumplir con este requisito.

#### A. Software

Existe software específico para computadores los cuales se comunican directamente con el controlador encargado de transformar los comandos CNC en señales de movimiento. Permite al usuario mover la herramienta gracias a una interfaz gráfica, en la Figura 2.14 se muestra un programa utilizado para el control de una máquina CNC.

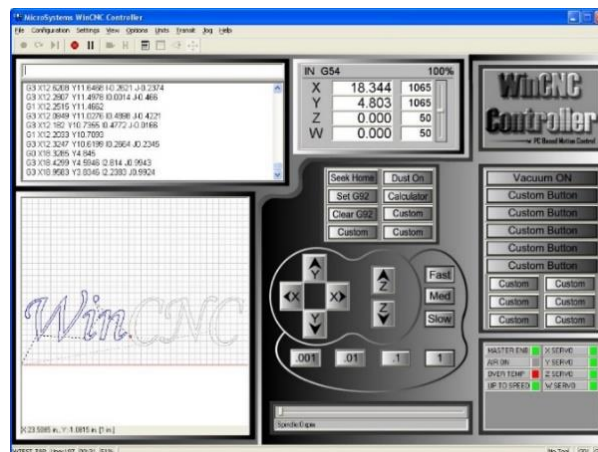


Figura 2.14. Software de control CNC [50].

#### Ventajas

- Es de fácil uso.
- No es necesario tener conocimiento de programación, porque el programa se encarga de comunicarse con el controlador.
- Al ser una interfaz visual, es amigable con el usuario.

## Desventajas

- Se necesita un computador.
- El computador necesita estar cerca de la máquina para un posicionamiento de cero pieza correcto.
- Si el software es de código cerrado, es necesario pagar la licencia para su utilización.

## B. Tablero de Control

Ellos integran diversas interfaces para caja de alimentación, eje, los equipos de mano, y construido en el PLC, adopta, con almacenamiento en disco de alta capacidad, y proporciona la función para USB, Ethernet DNC y extensión del programa y el intercambio [51]. En la Figura 2.15 se muestra un tablero de control.



Figura 2.15. Tablero de control CNC [52].

## Ventajas

- Fácil manejo.
- No necesita conocimientos de programación.
- Fácil implementación.

## Desventajas

- Necesita ubicarse cerca a la máquina.
- Debe tener compatibilidad con el controlador intérprete.

### C. Control alámbrico manual

El control manual es un dispositivo que puede desplazarse gracias a su cable, permitiendo al usuario ubicarse en frente de la máquina y así encerrar la máquina de una manera más sencilla. En la Figura 2.16 se muestra un control alámbrico manual para el posicionamiento del cero pieza en una máquina CNC.



*Figura 2.16. Control Manual CNC.*

#### **Ventajas**

- Permite al usuario desplazarse con el control y encerrar la máquina de una forma sencilla
- Posee switches rotativos que permiten la selección de eje y velocidad.
- Tiene un encoder incremental, que permite el control de la posición de los ejes.

#### **Desventajas**

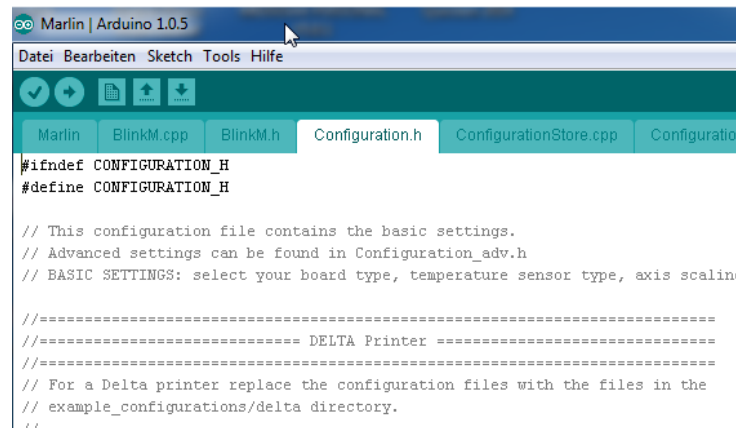
- Necesita un controlador para procesar los datos enviados desde el control.
- Se necesita conocimiento de programación para implementarlo.

#### **2.2.6.3.Transformar comandos CNC a señales de movimiento**

Esta función es una de las más importantes, pues los comandos CNC deben ser transformados a señales de movimiento para que la parte mecánica realice los movimientos necesarios para el corte de la pieza.

## A. Intérprete de código G (libre)

Los intérpretes de código G libres son Firmwares que se graban sobre un controlador para transformar los comandos CNC, que generalmente con códigos G, en señales de movimiento. En la Figura 2.17 se muestra el firmware Marlin, que puede ser cargado a la placa Arduino Mega.



```

Marlin | Arduino 1.0.5
Datei Bearbeiten Sketch Tools Hilfe
Marlin | BlinkM.cpp | BlinkM.h | Configuration.h | ConfigurationStore.cpp | Configuratio
#ifndef CONFIGURATION_H
#define CONFIGURATION_H

// This configuration file contains the basic settings.
// Advanced settings can be found in Configuration_adv.h
// BASIC SETTINGS: select your board type, temperature sensor type, axis scaling

//=====
//===== DELTA Printer =====
//=====
// For a Delta printer replace the configuration files with the files in the
// example_configurations/delta directory.
//

```

Figura 2.17. Intérprete Código G libre (Marlin) [53].

### Ventajas

- Al ser libre, se puede encontrar de forma gratuita.
- Es configurable, dependiendo de la aplicación de la máquina CNC.
- El costo adquisición de Shield para el controlador es bajo.

### Desventajas

- Se necesita conocimiento de programación para realizar la configuración.
- El controlador al ser un sistema embebido no es recomendado para aplicaciones industriales.

## B. Tarjetas controladoras CNC

Las tarjetas controladoras CNC están diseñadas exclusivamente para el control de movimiento de los ejes de una máquina CNC, al ser construidas para este propósito, pueden trabajar en ambientes industriales [54].



Figura 2.18. Tarjeta controladora CNC (Mach 3) [55].

### Ventajas

- Es de fácil implementación.
- No se necesita conocimientos de programación para su configuración
- Para su configuración se incluye un software para pc
- Se puede incluir otros dispositivos para un mayor control.

### Desventajas

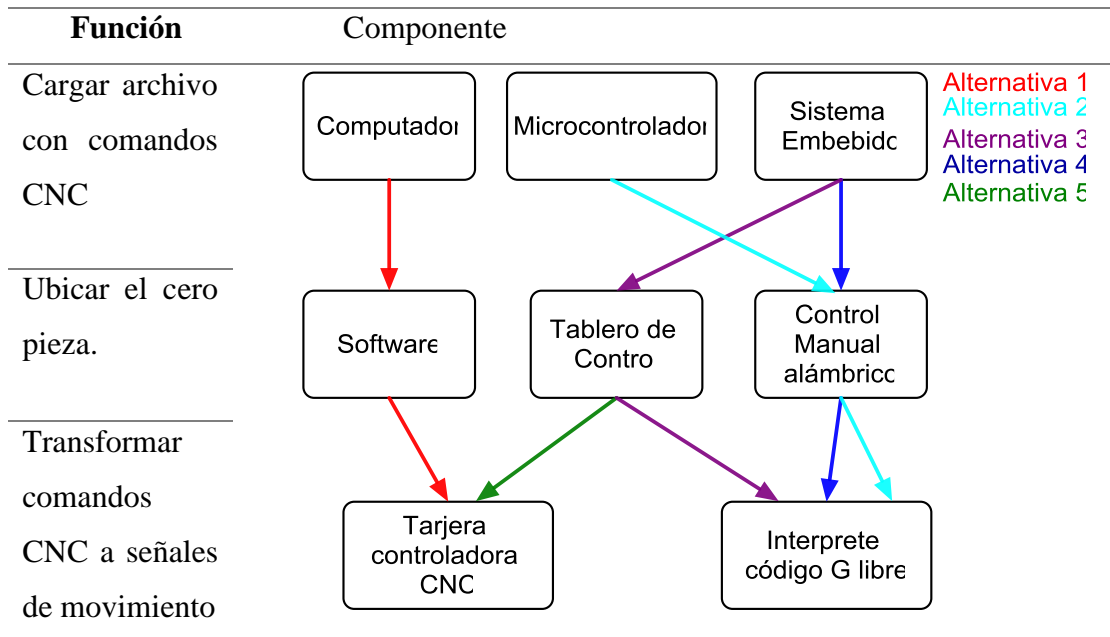
- El costo de adquisición es mayor que un controlador libre.
- Se necesita de un computador o de un módulo HMI para su funcionamiento.

#### 2.2.6.4. Combinaciones del módulo 2.

A continuación, se muestra en la Tabla 2-IX las posibles soluciones para las funciones del módulo 2.



Tabla 2-IX. Combinación de las soluciones del módulo 2.



Las alternativas para la solución de las funciones del módulo 2 son las siguientes:

- **Alternativa 1:** Computador junto con software y tarjeta controladora CNC.
- **Alternativa 2:** Microcontrolador junto a Control manual alámbrico e interprete código G libre.
- **Alternativa 3:** Sistema embebido junto a Tablero de control e intérprete de código G libre.
- **Alternativa 4:** Sistema embebido junto a control manual alámbrico e intérprete de código G libre.
- **Alternativa 5:** Tablero de control junto a Tarjeta controladora CNC.

#### 2.2.6.5. Evaluación del módulo 2

Para determinar la solución más factible se aplicará el método ordinal corregido de criterios ponderados. En la siguiente lista se encuentra los criterios de evaluación más importantes:

- **Velocidad de respuesta:** Es necesario una respuesta rápida por parte de los controladores para el posicionamiento del cero pieza.

- **Precio:** La máquina está dirigida a la pequeña industria, debido a esto, el factor económico es imprescindible.
- **Implementación:** La implementación debe ser lo más sencillo posible.
- **Fácil Operación:** La operación de la máquina debe ser lo más sencilla posible para que el usuario se sienta cómodo con la máquina.

A continuación, se procede a realizar una ponderación de cada uno de los criterios para establecer el nivel de importancia que tienen en el desarrollo del diseño de la máquina. La evaluación de cada uno de los criterios se muestra en la Tabla 2-X.

Tabla 2-X. Ponderación de los criterios del módulo 2.

Criterio	Respuesta	Precio	Implementación	Operación	$\Sigma+1$	Ponderación
Respuesta		0,5	0,5	0,5	2,5	0,2500
Precio	0,5		1	1	3,5	0,3500
Implementación	0,5	0		0,5	2	0,2000
Operación	0,5	0	0,5		2	0,2000
				Suma	10	1

***Precio > Respuesta > Implementación = Operación***

La Tabla 2-XI muestra el análisis del criterio de la Velocidad de Respuesta.

Tabla 2-XI. Evaluación las alternativas respecto al criterio de Velocidad de Respuesta.

Velocidad Respuesta	Alternativa 1	Alternativa 2	Alternativa 3	Alternativa 4	Alternativa 5	$\Sigma+1$	Ponderación
Alternativa 1		1	0,5	0,5	0,5	3,5	0,2333
Alternativa 2	0		0,5	0,5	0	2	0,1333
Alternativa 3	0,5	0,5		0,5	0,5	3	0,2000
Alternativa 4	0,5	0,5	0,5		0,5	3	0,2000
Alternativa 5	0,5	1	0,5	0,5		3,5	0,2333
					Suma	15	1

***Alternativa 1 = Alternativa 5 > Alternativa 3 = Alternativa 4 > Alternativa 2***

El siguiente criterio evaluado es el precio, mostrado en la Tabla 2-XII.

Tabla 2-XII. Evaluación las alternativas respecto al criterio de Precio.

<b>Precio</b>	Alternativa 1	Alternativa 2	Alternativa 3	Alternativa 4	Alternativa 5	$\Sigma+1$	Ponde- ración
Alternativa 1		0	0	0	0	1	0,0667
Alternativa 2	1		0,5	0	1	3,5	0,2333
Alternativa 3	1	0,5		0,5	1	4	0,2667
Alternativa 4	1	1	0,5		1	4,5	0,3000
Alternativa 5	1	0	0	0		2	0,1333
					Suma	15	1

***Alternativa 4 > Alternativa 3 > Alternativa 2 > Alternativa 5 > Alternativa 1***

En la Tabla 2-XIII se muestra el análisis de las alternativas en base al criterio de la Implementación.

Tabla 2-XIII. Evaluación las alternativas respecto al criterio de Velocidad de Implementación.

<b>Implementación</b>	Alternativa 1	Alternativa 2	Alternativa 3	Alternativa 4	Alternativa 5	$\Sigma+1$	Ponde- ración
Alternativa 1		1	1	0,5	0,5	4	0,2667
Alternativa 2	0		0	0	0	1	0,0667
Alternativa 3	0	1		0	0	2	0,1333
Alternativa 4	0,5	1	1		0,5	4	0,2667
Alternativa 5	0,5	1	1	0,5		4	0,2667
					Suma	15	1

***Alternativa 1 = Alternativa 4 = Alternativa 5 > Alternativa 3 > Alternativa 2***

El último criterio a ser evaluado es la fácil operación, este análisis es mostrado en la Tabla 2-XIV.

Tabla 2-XIV. Evaluación las alternativas respecto al criterio de Fácil Operación.

<b>Fácil Operación</b>	Alternativa 1	Alternativa 2	Alternativa 3	Alternativa 4	Alternativa 5	$\Sigma+1$	Ponderación
Alternativa 1		0,5	0,5	0	0,5	2,5	0,1667
Alternativa 2	0,5		0,5	0,5	0	2,5	0,1667
Alternativa 3	0,5	0,5		0,5	0	2,5	0,1667
Alternativa 4	1	0,5	0,5		0,5	3,5	0,2333
Alternativa 5	0,5	1	1	0,5		4	0,2667
						Suma	15
							1

**Alternativa 5 > Alternativa 4 > Alternativa 3 = Alternativa 2 = Alternativa 1**

Después de realizar la evaluación de cada una de las alternativas respecto a cada criterio de evaluación, se realiza una tabla de análisis de resultados para determinar la mejor alternativa para la solución del módulo 2.

Tabla 2-XV. Tabla de Resultados para el módulo 2.

<b>Resultados</b>	Respuesta	Precio	Implementación	Operación	$\Sigma$	Prioridad
Alternativa 1	0,25x0,23	0,35x0,07	0,20,x0,27	0,20,x0,17	0,17	<b>4</b>
Alternativa 2	0,25x0,13	0,35x0,23	0,20,x0,07	0,20,x0,17	0,16	<b>5</b>
Alternativa 3	0,25x0,20	0,35x0,27	0,20,x0,13	0,20,x0,17	0,20	<b>3</b>
Alternativa 4	0,25x0,20	0,35x0,30	0,20,x0,27	0,20,x0,23	0,26	<b>1</b>
Alternativa 5	0,25x0,23	0,35x0,13	0,20,x0,27	0,20,x0,27	0,21	<b>2</b>

En la Tabla 2-XV se muestra a la alternativa 4 como la mejor solución para el cumplimiento de las funciones del módulo 4.

### 2.2.7. Propuesta de diseño

Después de realizar el despliegue de función de la calidad, el análisis funcional, la descomposición y solución de módulos se define la mejor alternativa para el desarrollo de la parte electrónica de la máquina CNC para corte de planchas de aluminio.

Los elementos que componen la solución son:

- **Switch:** Para el encendido de la máquina.
- **Comunicación serial:** Para enviar la información entre controladores.
- **Finales de carrera:** Como protecciones mecánicas, para evitar colisiones.
- **Sistema embebido (Arduino):** Para cargar los archivos con comandos CNC, además de recibir y procesar la información obtenida del Control manual.
- **Control Manual Alámbrico:** Para el fácil posicionamiento del cero pieza.
- **Intérprete de código G libre:** Para transformar los comandos de CNC en señales de movimientos. El intérprete seleccionado es el firmware Marlin que posee grandes prestaciones y puede ser cargado en un controlador Arduino Mega junto con la shield RAMPS.

### 2.3. Diseño de HMI

El diseño de la HMI se realiza para cubrir uno de los principales requerimientos de la máquina, el fácil posicionamiento del cero pieza, para el cual se empleará el control manual alámbrico, para que los datos obtenidos sean interpretados por el por el firmware Marlin, es necesario que sean transformados a código G antes de ser enviados. Para esta transformación se utiliza el sistema embebido Arduino, al que será conectado el control y mediante programación los pulsos generados por el encoder y las señales enviadas por los selectores sean transformados a código G.

El primer paso antes de realizar el diseño de la HMI es determinar las funciones que debe cumplir, un proceso muy utilizado es el diagrama de flujo, también llamado Flujograma de Procesos o Diagrama de Procesos, representa la secuencia o los pasos lógicos (ordenados) para realizar una tarea mediante unos símbolos. Dentro de los símbolos se escriben los pasos a seguir. Un diagrama de flujo debe proporcionar una información clara, ordenada y concisa de todos los pasos a seguir. Por lo dicho anteriormente, podríamos decir que: "Un diagrama de flujo es una representación gráfica o simbólica de un proceso" [56].

### 2.3.1. Diagrama de flujo

En la Figura 2.19 se muestra el diagrama de flujo, en él se muestran las funciones que debe cumplir la HMI, las cuales se realizarán con los elementos elegidos mediante el método de criterios ponderados. A continuación, se procede a describir cada uno de los objetivos a cumplir por la HMI.

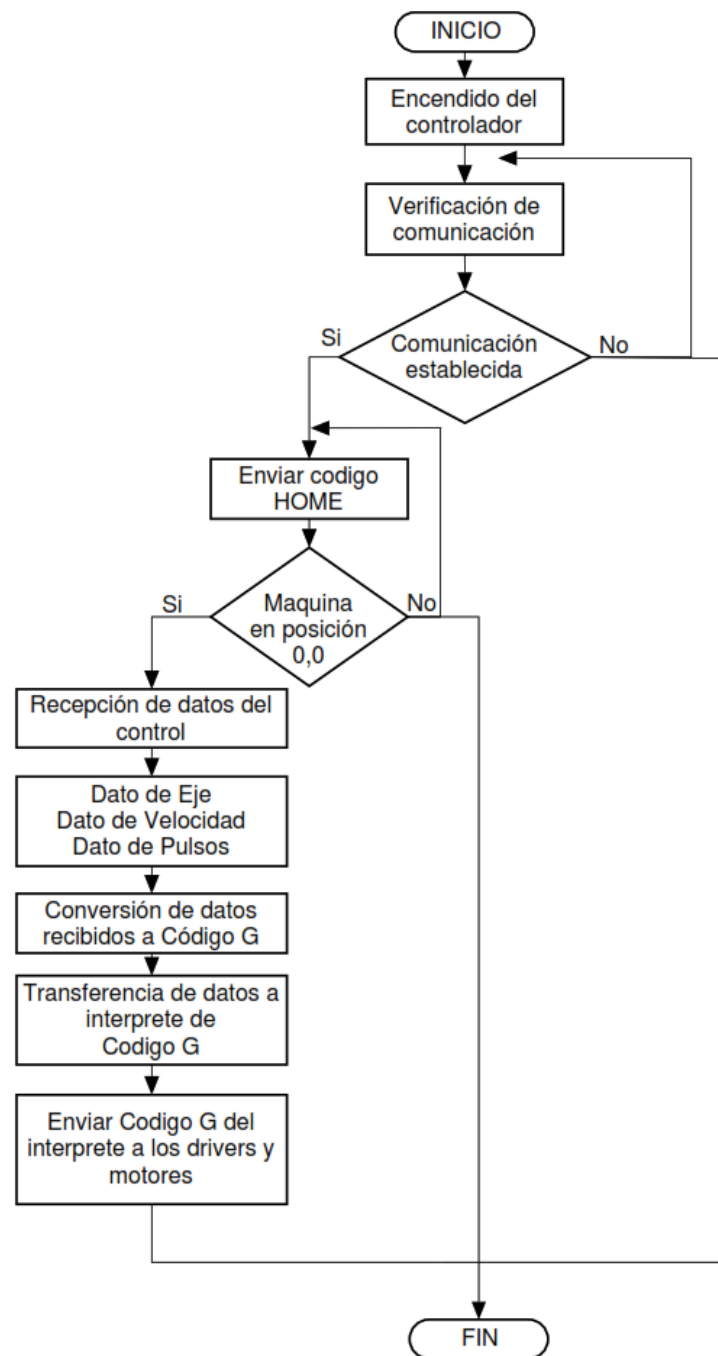


Figura 2.19. Diagrama de Flujo.

### **2.3.1.1. Encendido del controlador**

Esta función se cumple mediante el uso de un selector de dos posiciones, permitiendo el paso de la energía eléctrica a todos los componentes dentro del gabinete.

### **2.3.1.2. Verificación de la comunicación.**

Una vez que se encuentren encendidos todos los dispositivos de la máquina, se necesita verificar que existe comunicación entre los controladores. Para esto se utiliza la comunicación serial, disponible en las placas de Arduino, el sistema embebido encargado de la recepción de los datos proporcionados por el control espera el mensaje de inicialización que el firmware Marlin envía una vez que se inicializa correctamente. Después de ser recibido este mensaje, se puede dar por establecida la comunicación, caso contrario existe problemas en las placas.

### **2.3.1.3. Enviar comando HOME**

Una vez que la comunicación entre controladores se encuentre establecida se procede a enviar el comando HOME al controlador CNC para que la herramienta se traslade al cero máquina. Este comando debe ser enviado en código G para que sea entendido, G28 es el comando que debe enviarse para que el firmware pueda entenderlo como “HOME” y llevar la herramienta al cero máquina. Por seguridad se lo realiza de manera individual para cada eje, siendo el primero el eje Z para precautelar la integridad de la herramienta y evitar una posible colisión. Esta función será enviada cuando se presione el botón verde desde el tablero de control. Cuando la herramienta se encuentre en esta posición el controlador CNC envía un mensaje confirmando que la acción se realizó.

### **2.3.1.4. Recepción de datos del control**

Cuando la herramienta se encuentre en cero máquina se puede enviar datos del control al intérprete de código G, pero para ello primero hay que convertirlo para que sea entendido por Marlin. Para esto se toman los siguientes datos que son proporcionados por el hardware del control.



Figura 2.20.Elementos de Control alámbrico

- **Eje:** Este dato es tomado de un switch rotativo de 5 posiciones que posee el control, de estas posiciones, solo cuatro serán tomadas en cuenta: OFF, X, Y y Z, la primera posición indica que el control está en modo inactivo, es decir, que los datos que se generen en el control mientras se encuentre en este estado no serán tomados en cuenta, y las posiciones X, Y y Z representan a los ejes de la máquina, la quinta posición no se la usara pero su función es la de controlar un cuarto eje.
- **Multiplicador de Pulsos:** Al igual que el dato anterior este valor es tomado de un switch rotativo de 3 posiciones y son representados por las siguientes etiquetas x1, x10 y x100, que determina la cantidad de pulsos por unidad de rotación del encoder.
- **Velocidad:** Este dato es referenciado al multiplicador de pulsos. A mayor cantidad de pulsos, mayor será la velocidad empleada para el desplazamiento de la herramienta.

La variable **Eje** tiene el objetivo de determinar el eje en el cual se va a realizar el desplazamiento de la herramienta, puede tener los valores de tipo cadena X, Y y Z, correspondientes a las posiciones 2,3 y 4 del primer selector. La primera posición del



selector corresponde a la acción OFF, esto significa que cuando se seleccione esta posición ningún dato será enviado del controlador del control hacia el controlador CNC. Además de estas posiciones existe una quinta posición la cual puede ser utilizada para el manejo de un cuarto eje.

La variable **Multiplicador** tiene el objetivo de multiplicar o dividir la cantidad de pulsos leídos del encoder dependiendo de la posición en la que se encuentre el switch rotativo. Para la primera posición, correspondiente a la etiqueta X1 los pulsos serán divididos para que la herramienta se traslade una fracción de unidad de medida, para la segunda posición, correspondiente a la etiqueta X10, se realizará una división de menor magnitud que en la velocidad X1. Para X100 que corresponde a la última posición, los pulsos serán divididos en menor magnitud que las velocidades anteriores, para que la herramienta se desplace una mayor distancia por cada pulso del encoder.

La variable **Velocidad** está ligada a **Multiplicador** y es proporcional, esto quiere decir que si el multiplicador es X1 la velocidad a la que la herramienta se desplazará debe ser baja, esta velocidad es usada cuando la herramienta se encuentre cerca de la pieza a mecanizar, en cambio si el multiplicador es X10 o X100 la velocidad de desplazamiento puede ser alta, estas velocidades son usadas para el desplazamiento de la herramienta a lo largo del área de trabajo de la máquina, donde no exista el riesgo de colisiones.

Una vez obtenidos estos datos, deben ser transformados a código G, para lo cual se definen algunos parámetros de acuerdo con los distintos valores que puede entregar el control. El parámetro principal es que todos los datos sean de tipo string o cadena cuando se transmita desde el controlador del control hacia el controlador que contiene el firmware CNC.

Ahora se analiza los tipos de variables y las operaciones que se van a realizar antes de que sean transmitidas:

- **Eje:** esta variable según la posición en la que se encuentre el switch rotativo tendrá los valores tipo cadena "X" si se encuentra en la posición 2, "Y" si se encuentra en la posición 3 y "Z" si se encuentra en la posición 4.

- **Pulsos:** Esta variable es de tipo entero, debido a que se encarga de contar los pulsos que son entregados por el encoder rotativo.
- **Multiplicador:** Este variable tomará los valores numéricos “20” si el switch rotativo se encuentra en la primera posición, “4” si se encuentra en la posición 2 y “2” cuando se encuentre en la posición 3.
- **Velocidad:** Esta variable también depende del switch rotativo Multiplicador de pulsos, para la posición 1 la variable tomará el valor tipo cadena “G01 F200” y para la posición 2 tendrá el valor “G00 F400” y para la posición 3 tendrá el valor de “G00 F600, estos son códigos G que el firmware interpreta como velocidad de desplazamiento baja para G01 y alta para G00.

Estás variables son las que se encuentran ligadas a los datos entregados por el control, además de estas, existen otras las cuales permiten la interacción entre los valores entregados por el control y obtener los valores que serán reunidos para su transmisión.

#### 2.3.1.5. Conversión de datos recibidos a código G

Una vez que se obtengan todos los valores del control y realizar las operaciones necesarias es necesario unir todos los datos según el siguiente formato:

**G0.... F... X... Y... Z....**

Este es el formato en el cual se escribe el código G que puede ser interpretado por cualquier máquina CNC, las variables son: la velocidad, el avance y las coordenadas de cada eje.

El siguiente paso es llevar todo lo mencionado con anterioridad a un código de programación para que pueda ser entendido por el sistema embebido. El IDE usado es Arduino 1.8. El código se realiza con funciones que serán ejecutadas cuando sean requeridas.

### **2.3.1.6. Transferencia de datos a intérprete de código G**

El último paso es enviar la línea de código desde el controlador del control hacia el controlador CNC, esta acción es realizada mediante comunicación serial disponible en los dos controladores. El código de la placa Arduino Mega para procesar los datos entregados por el control manual se encuentra en el Anexo 2.

## **2.4. Construcción de estructura de la máquina CNC**

La construcción de la estructura de la máquina CNC se realiza con los planos del trabajo de grado “Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Diseño”.

La estructura de la máquina se construye con materiales existentes en entorno local. Existen partes que necesariamente necesitan ser realizadas mediante un proceso de manufactura, porque se diseñaron para cumplir objetivos específicos en la máquina.

### **2.4.1. Estructura externa**

La estructura externa o principal, que está encargada de sujetar toda la máquina, se construye con correa estructural tipo G de 12cm de alto por 3mm de espesor, según las especificaciones de los planos se construye un rectángulo de 190x110cm, unido mediante soldadura, además existe una estructura interna, cuyo objetivo es la sujeción de las guías. La estructura interna, encargada de sostener los elementos del eje Y, está formada por dos platinas de 100mm de alto y 6mm de espesor, separadas una distancia de 121.8 cm entre si. En la Figura 2.21 se muestra el diseño de la estructura externa con las respectivas dimensiones

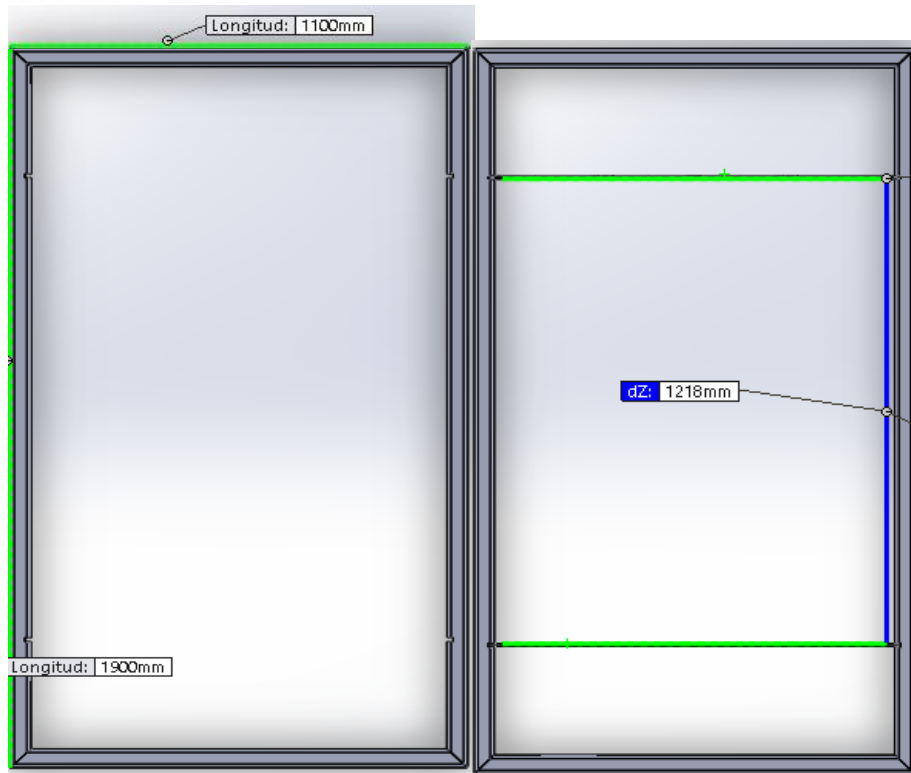


Figura 2.21. Estructura externa de la máquina CNC.

#### 2.4.2. Sistema de movimiento

El sistema de movimiento para todos los ejes es similar, cada uno está conformado por dos varillas lisas de 20mm de diámetro para la sujeción y soporte de la carga de los elementos. Este diámetro es sugerido después de haberse realizado los cálculos respectivos en el trabajo de grado “Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Diseño”. En la Figura 2.22 se muestra el acero plata de 20mm de diámetro, para cada eje la longitud de la guía cambia.



Figura 2.22. Varilla lisa 20mm diámetro (Acero plata) [57].

Para la sujeción de las guías a la estructura de cada uno de los ejes, se utilizan sujetadores de pared para evitar posibles desplazamientos de las guías. El sujetador se encuentra ilustrado en la Figura 2.23, este elemento es para todas las guías.



*Figura 2.23. Sujetador de pared para guías [58].*

Para la transformación de movimiento rotacional suministrada por el motor a lineal se utiliza una varilla roscada de 25.4mm de diámetro con dos tuercas para cada eje. La varilla es mostrada en la Figura 2.24. Las longitudes dependen de cada eje.



*Figura 2.24. Varilla roscada*

Para el acoplamiento del tornillo y motor se utiliza un acople flexible ranurado de aluminio, 10-16 mm, este permite absorber desalineaciones entre los ejes unidos [59].



*Figura 2.25. Acople flexible ranurado de aluminio [60].*

En la Figura 2.25 se muestra un acople flexible de aluminio con prisioneros para la sujeción entre ejes.

Para el desplazamiento a través de las guías se usan rodamientos lineales cerrados y tipo T de 20 mm de diámetro interior. Sobre los rodamientos cerrados se ubica la mesa, mientras que los tipo T son usados en los ejes X y Z. Estos elementos se pueden visualizar en la Figura 2.26. Los rodamientos cerrados se usan para la construcción del eje Y, mientras que los tipo T se usan en los ejes X y Z.



*Figura 2.26. Rodamiento lineal cerrado y tipo T [75].*

Para la sujeción del tornillo a la estructura de los ejes se utiliza una chumacera de pared de 20mm de diámetro interior, debido a que el tornillo tiene un cambio de sección en la parte final. A continuación, se muestra la chumacera en la Figura 2.27.



*Figura 2.27. Chumacera de pared 20mm [61].*

Con estos elementos se procede a realizar los sistemas de movimiento de cada uno de los ejes.

### 2.4.2.1. Estructura Eje X y Z

Se encuentra constituida por dos platinas de 58cm de alto, 20cm de ancho y 10mm de espesor acopladas a la estructura externa mediante pernos, separadas una distancia de 110cm, tienen el objetivo de sujetar las guías, el motor y el tornillo. Para la sujeción de las guías se utilizan los sujetadores de pared. El motor se acopla a la estructura mediante una base soporte y la sujeción del tornillo a la estructura se la realiza mediante la chumacera. En la se ilustra la estructura para el sistema de movimiento del eje X.

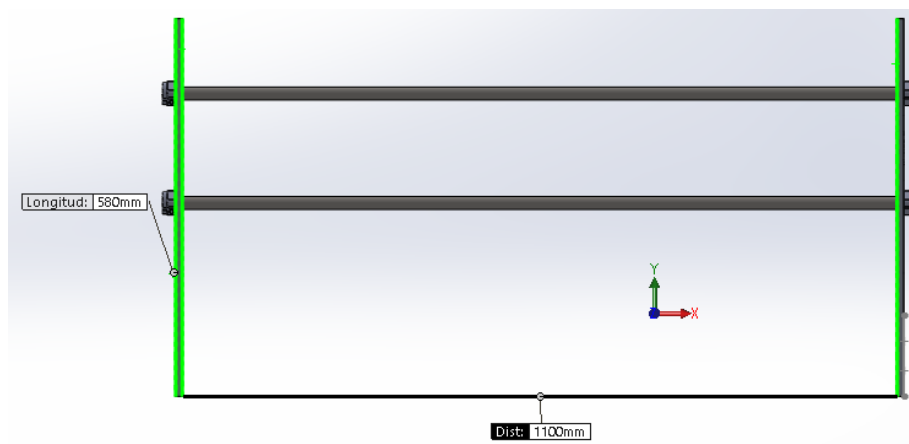


Figura 2.28. Estructura del eje X.

Para permitir el movimiento de la torre a lo largo del tornillo se construye una estructura en la cual se acoplan y sujetan las tuercas, esta estructura tiene piezas que no están disponibles en el mercado tal y como están diseñadas, necesitando un método de manufactura para obtenerlas. Teniendo en cuenta que el factor económico es un factor crucial, se analizan los métodos disponibles en el entorno local: maquinado y fundición en arena, de los cuales se escoge realizar las piezas diseñadas por fundición en arena, al ser más accesible económicamente. Para elaborar piezas por este método es necesario tener los moldes, los cuales se realizan de MDF, para aumentar la precisión en la fundición, los moldes son realizados mediante corte láser.

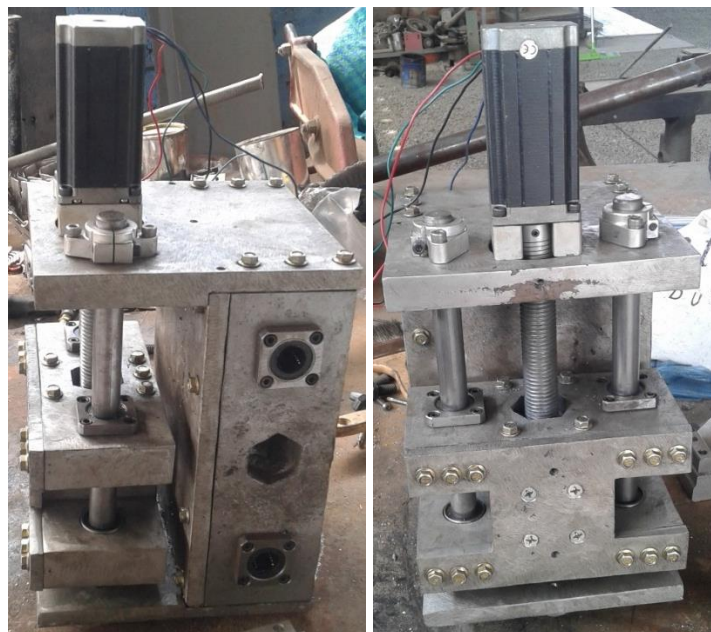
En la Figura 2.29 se muestran los moldes realizados en MDF que son utilizados para realizar la fundición en arena.



*Figura 2.29. Moldes en MDF*

Las piezas se realizan en aluminio, por sus grandes prestaciones, principalmente por qué es ligero, y su relación resistencia-peso es más alta que otros metales [62].

Después de la fundición de las piezas se realiza un proceso de pulido para mejorar el acabado. Por último, se procede a integrar todos los componentes entre sí, formando una estructura que se denomina torre, esta estructura se desplaza a lo largo del eje X. La estructura del eje Z forma parte de la torre y consta de los mismos elementos a excepción del soporte del motor PAP, para este eje se usa un soporte tipo mesa.



*Figura 2.30. Estructura eje X y Z.(Torre)*



En la Figura 2.30 se muestra el sistema de movimiento del eje X y Z ensamblado con las piezas realizadas mediante la fundición de aluminio, la unión de los elementos se las realiza mediante pernos.

#### 2.4.2.2.Estructura Eje Y

La estructura del eje Y está conformada por platinas de 10mm de ancho por 6mm de espesor, estas platinas se encuentran acopladas a las estructura externa de la máquina por medio de pernos, tiene la función de sostener las guías, el tornillo y el motor.

Para la construcción de la mesa se utiliza perfiles de aluminio estructural de 30x30mm. Las especificaciones del área de trabajo es que la mesa sea de 800x800mm, por lo que se realiza una mesa de 900x900mm, teniendo en consideración la forma de sujeción de la pieza. Para cumplir con este objetivo se agrupan 30 piezas de perfiles de aluminio de 900mm de largo. Para realizar un solo conjunto se las agrupa con la ayuda de una platina de 50x30mm de 90 cm de largo y tornillos autoperforantes que se encargan de fijar cada uno de los perfiles de aluminio a las platinas.

Para acoplar la mesa a las guías se realiza una estructura aparte en la cual se encuentran acoplados los rodamientos y dentro de ellos las guías, la mesa se encuentra fijada sobre esta estructura, permitiendo que se desplace a lo largo del eje Y.

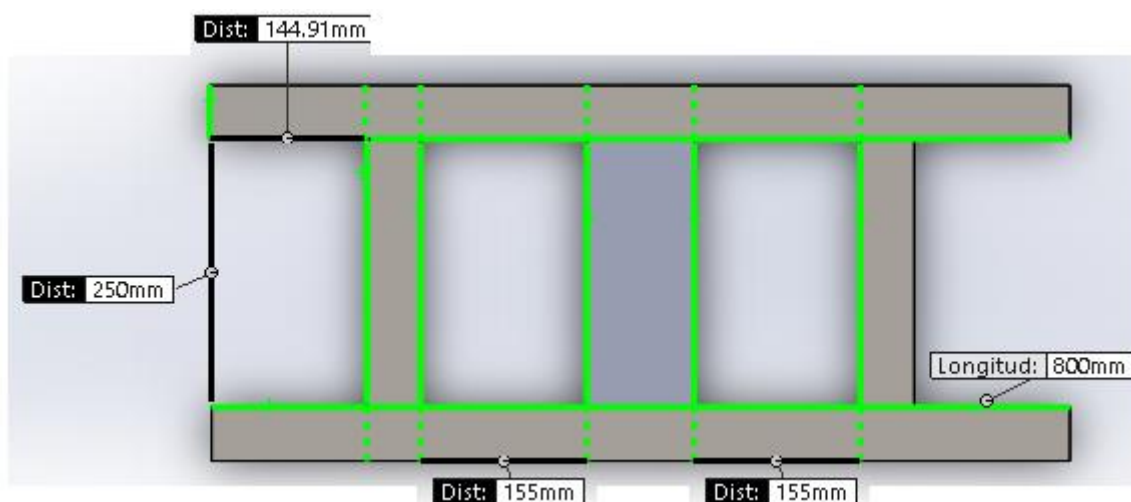
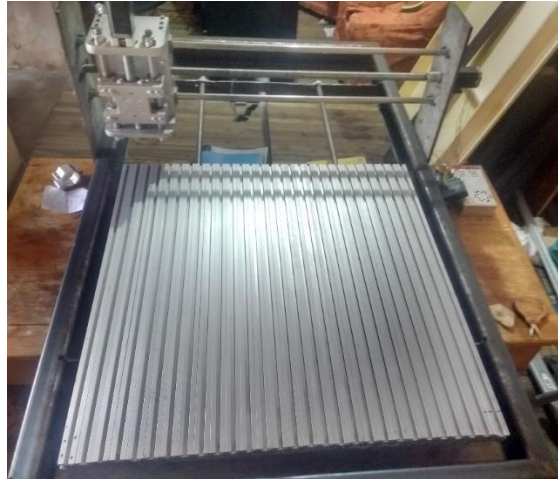


Figura 2.31. Estructura para acoplamiento de rodamientos, tornillo y guías.

En la Figura 2.31 se muestra la estructura encargada de acoplar la mesa de trabajo, los rodamientos lineales, la estructura para el movimiento a través del tornillo.

Una vez contruidos los sistemas de movimiento con sus respectivas estructuras, se ensamblan todas las partes para obtener la máquina completa. En la Figura 2.32 se muestra la máquina con todos sus elementos acoplados.



*Figura 2.32. Estructura y sistema de movimiento acoplados.*

## **2.5. Implementación de la HMI**

Para la implementación de la HMI se toma en cuenta los elementos electrónicos sugeridos en el trabajo de grado titulado “Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Diseño”, los cuales son:

- Motor paso a paso NEMA 23 HS45-4204S
- Controlador para motor paso a paso TBT6600 Y DM542T
- Potencia para motor husillo de 500W

Las hojas de datos del motor y los controladores se encuentran en el Anexo 1.

### 2.5.1. Selección de fuentes de alimentación.

Con estos elementos y los elementos elegidos en este trabajo se procede a realizar la selección de fuentes de alimentación, utilizando los consumos de corriente máximos de cada elemento:

- La corriente máxima que consume un motor paso a paso nema 23 es de 3 A RMS y 4.2 A pico. Está corriente también es la máxima que puede suministrar los controladores DM542. Las corrientes de los 3 motores sumadas indican un consumo de 12.6 A, el voltaje de alimentación de los motores y controladores es de 24 VDC. Tomando en cuenta que los motores consumen toda la corriente cuando se encuentran sin realizar ningún movimiento [63], la elección es una fuente de alimentación de 24 V y 15 A de corriente.
- Debido a que los controladores consumen corrientes altas, tienden a calentarse, por lo que se necesita un sistema de ventilación para que el aire circule por todo el sistema evitando sobrecalentamiento en los elementos, para cubrir esta necesidad se usa ventiladores de 120mm que se alimentan a 12VDC y consumen 250mA, esto implica que se necesita una fuente de este voltaje. Otro componente que usa el mismo voltaje es el convertidor de señal PWM a 0-10 V con un consumo de corriente promedio de 100mA.
- Para la etapa de control se usan dos sistemas embebidos Arduino Mega, necesitan de un voltaje de alimentación de 5 voltios llegando a consumir una corriente máxima de 500 mA cada uno alimentados por el puerto USB [64]. Para la alimentación de estos dispositivos se utiliza un conversor DC-DC reductor, capaz de convertir el voltaje de 12 a 5 V y entregar una corriente máxima de 3A, el control alámbrico manual también se alimenta a una tensión de 5V y su consumo de corriente promedio de 100 mA [65]. La corriente del conversor, el convertidor, ventiladores y control indican una corriente total de 4.2 A, se elige una fuente de alimentación de 12V y 5A.
- La potencia necesaria para realizar cortes de 5mm de profundidad es de 500 Watts, debido a este requerimiento se elige un husillo que satisfaga este parámetro. Posee

En la Figura 2.33 se muestra el kit de motor husillo que se adquiere para la implementación en la máquina CNC.



Figura 2.33. Kit husillo 500W con control de velocidad [66].

### 2.5.2. Cálculo de la protección eléctrica.

Es un dispositivo que salvaguarda la integridad de una instalación eléctrica residencial o comercial de sobrecargas o cortocircuitos, cortando el suministro eléctrico en caso de producirse una falla [67].

Para adquirir la protección eléctrica para salvaguardar la integridad de los elementos, es necesario sumar las corrientes máximas que van a consumir todos los elementos. A continuación, se detalla el consumo de todos los componentes a implementarse en un gabinete de control.

- Motor NEMA 23 eje X con controlador DM542T 4.2 A.
- Motor NEMA 23 eje Y con controlador DM542T 4.2 A.
- Motor NEMA 23 eje Z con controlador TBT6600 4.2 A.
- Arduino Mega Firmware CNC 0.5 A [64].
- Arduino Mega control manual 0.5 A.
- Control alámbrico manual 0.12 A.
- Luces de señalización 0.6 A.
- Controlador husillo 4.2 A.

- Conversor PWM a 0-10V 0.1 A [68].

Al sumar las corrientes de todos los elementos electrónicos se obtiene una total de 18,2A. El valor comercial más cercano para las protecciones es de 20 A. En la Figura 2.34 se muestra una protección eléctrica para una fase de 20A.



Figura 2.34. Protección eléctrica 20 A [67].

### 2.5.3. Implementación elementos electrónicos

Se selecciona un gabinete eléctrico para colocar todos los elementos electrónicos adquiridos, se estima un área aproximada de 60x60cm. Para diferenciar los componentes dentro del gabinete se establecen grupos o fases: alimentación, potencia, control y salida. En la Figura 2.35 se muestra el gabinete eléctrico utilizado.



Figura 2.35. Gabinete de Control.

Para la interconexión entre fases es necesario elaborar dos tarjetas auxiliares, estas son denominadas como TA1 y TA2. La TA1 se encargada de interconectar las salidas de la placa CNC RAMPS 1.4 hacia los controladores de los motores PAP, mientras que la TA2 se encarga de permitir la conexión entre el Arduino Mega con el control alámbrico manual y el módulo relé, este último esta encargado de la activación de las luces indicadoras ubicadas en la parte frontal del gabinete, además las dos tarjetas auxiliares permiten la comunicación entre los dos Arduino. Los diagramas PCB se encuentran en el anexo 3.

Después de colocar los componentes en el grupo correspondiente dentro del gabinete se procede a conectarlos entre sí, para realizar el cableado se usa canaleta ranurada que rodea los elementos, para tener una mejor organización. A los cables se les coloca terminales al final de cada extremo.

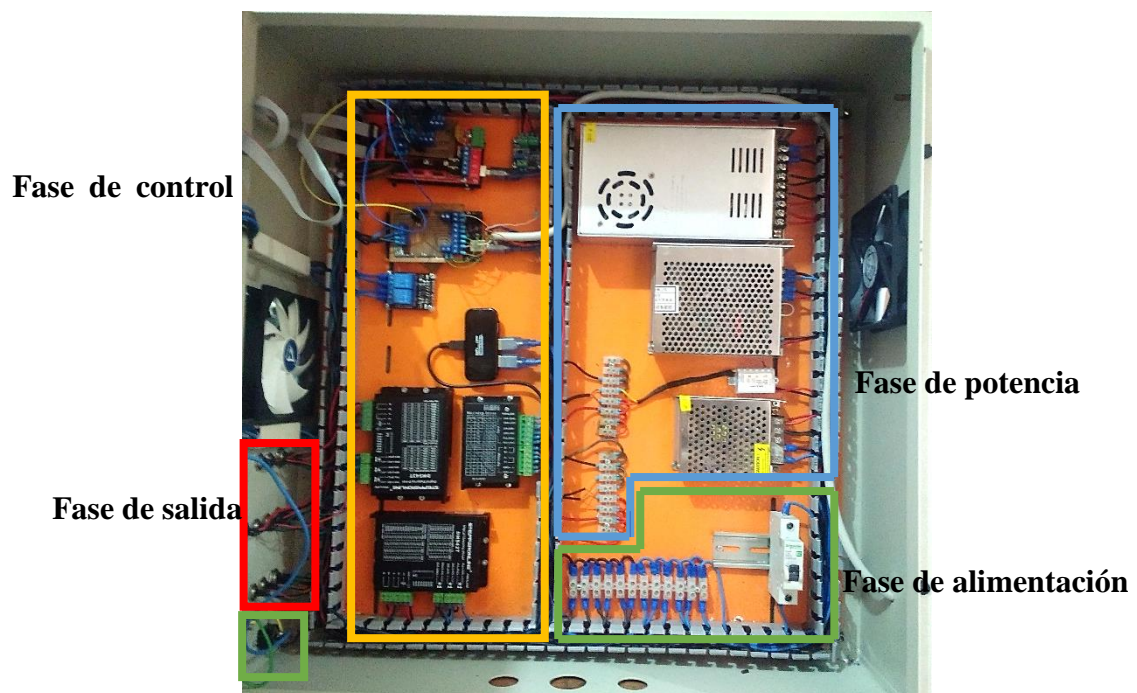


Figura 2.36. Interior del gabinete de control y sus fases.

En la Figura 2.36 se muestra el interior del gabinete de control con los dispositivos ubicados según las fases establecidas.

La fase de alimentación tiene como elemento principal el conector macho que permite la alimentación a la red eléctrica, después se encuentra la protección eléctrica para evitar daños por sobrecargas o cortocircuitos, posteriormente esta una línea de distribución que

está destinada a brindar el voltaje AC a las fuentes de alimentación. En la Figura 2.37 se muestra el conector para la alimentación de energía eléctrica.



*Figura 2.37. Conector 110V AC.*

La fase de potencia conformada por las fuentes de alimentación, encargadas de suministrar el voltaje DC para los elementos de control: la placa de control CNC, Arduino Mega, controladores de los motores PAP, módulo conversor PWM a 0-10V, modulo relé, ventiladores y husillo.

Por último, la fase de salida que permite la interconexión y control de los elementos que se encuentran fuera del gabinete. Esta fase está compuesta por terminales macho, permitiendo que los dispositivos que se encuentran fuera puedan desconectarse cuando sea necesario, facilitando su movilización. A continuación, en la se muestra los terminales que conforman la fase de salida.



*Figura 2.38. Fase de salida.*

## 2.5.4. Implementación algoritmo de programación para el control de la máquina CNC

El controlador que está encargado del control de los motores PAP necesita del firmware Marlin, el cual es capaz de interpretar código G, monitorear los sensores final de carrera, configurar el software y hardware de la máquina, recibir información de otros dispositivos a través del puerto serial, mostrar información de posición y estado de la máquina a través de una pantalla LCD, entre otras características adicionales. Al ser software libre, se tiene acceso a su código fuente en página oficial de Marlin <http://marlinfw.org/>.

La placa Arduino Mega esta encargado de receptrar los datos entregados por el control, realizar las operaciones necesarias y convertirlos a código G y enviar por el puerto serial a la placa de control CNC, también se encarga de la señalización del gabinete de control, mediante un módulo relé.

Marlin al ser un firmware con muchas prestaciones, necesita ciertas configuraciones, para que la máquina funcione correctamente. A continuación, se detallan las modificaciones realizadas antes la implementación.

### 2.5.4.1. Configuración firmware Marlin

Luego de descargar el firmware Marlin, es necesario disponer del IDE de Arduino para la configuración de los parámetros de la máquina. El primer paso es abrir el archivo llamado Marlin.ino, este archivo está ubicado dentro de la carpeta del mismo nombre como se muestra en la Figura 2.39.

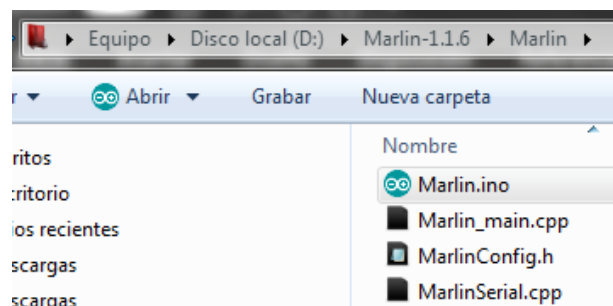


Figura 2.39. Archivo Marlin.ino



Una vez abierto el archivo en el IDE de Arduino se debe dirigir a la pestaña *Configuration.h*, en este archivo se encuentran las configuraciones generales de la máquina, a continuación, se procede a configurar algunos parámetros.

La primera sección para configurar es el inicio del programa. El firmware Marlin permite que se muestre una imagen de arranque personalizada, es necesario habilitar la opción: `#define SHOW_CUSTOM_BOOTSCREEN` como se ve en la Figura 2.40.

```
// *** VENDORS PLEASE READ *****
//
// Marlin now allow you to have a vendor boot image to be displayed
// start. When SHOW_CUSTOM_BOOTSCREEN is defined Marlin will first
// custom boot image and then the default Marlin boot image is shown
//
// We suggest for you to take advantage of this new feature and keep
// boot image unmodified. For an example have a look at the bq Hep
// example configuration folder.
//
#define SHOW_CUSTOM_BOOTSCREEN
```

Figura 2.40. Función Imagen de inicio personalizada.

Esta opción permite incluir una imagen de arranque para que sea mostrada en la pantalla LCD, antes de implementarse es necesario convertir la imagen a código hexadecimal, para lo cual el firmware nos ofrece 2 herramientas online para convertir una imagen en una matriz hexadecimal. Estas herramientas permiten cargar un archivo de imagen para convertirlo en una matriz hexadecimal que el firmware puede interpretar. En la Figura 2.41 se muestra una imagen convertida a matriz hexadecimal.

### Bitmap Converter

This tool converts a JPEG or PNG image file to a U8Glib-compatible C/C++ array that can be used in Marlin as a custom `_Bootscreen.h` file or as icons for Graphical Displays. Darker colors in the source image will produce "on" pixels, while lighter colors will be converted to transparent pixels.

If this tool doesn't work in your browser, use the server-side tool at [digole.com](http://digole.com).



Figura 2.41. Conversión de imagen con herramienta Bitmap Converter [69].

El código que muestra la herramienta se coloca en la pestaña *\_Bootscreen.h* para que sea mostrada al momento de encender la máquina. Además, se puede configurar el tiempo que permanezca la imagen. En la Figura 2.42 se muestra la configuración de la pestaña *\_Bootscreen.h*.

```

#include <avr/pgmspace.h>

#define CUSTOM_BOOTSCREEN_TIMEOUT 6500
#define CUSTOM_BOOTSCREEN_BMPWIDTH 128
#define CUSTOM_BOOTSCREEN_BMPHEIGHT 64

const unsigned char custom_start_bmp[1024] PROGMEM = {0x00,0x00,0x
0x7F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF
0x7C,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00
0x7C,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00
0x7F,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF,0xFF
0x7C,0x00,0x00,0x7C,0x00,0x00,0xFF,0x00,0x01,0xC0,0x00,0x1F,0xFC
0x7C,0x00,0x00,0x7C,0x00,0x00,0xFF,0x00,0x01,0xC0,0x00,0x0F,0xFC
0x7C,0x00,0x00,0x7C,0x00,0x00,0xFF,0x00,0x01,0xC0,0x00,0x07,0xFC
0x7C,0x00,0x00,0x7C,0x00,0x00,0xFF,0x00,0x01,0xC0,0x00,0x03,0xFC
0x7C,0x00,0x00,0x7C,0x00,0x00,0xFF,0x00,0x01,0xC0,0x00,0x01,0xFC
0x7C,0x00,0x00,0x7C,0x00,0x00,0xFF,0x00,0x01,0xC0,0x00,0x00,0xFC
0x7C,0x00,0x00,0x7C,0x00,0x00,0xFF,0x00,0x01,0xC0,0x00,0x00,0x7C
0x7C,0x00,0x00,0x7C,0x00,0x00,0xFF,0x00,0x01,0xC0,0x00,0x00,0x3C

```

Figura 2.42. Pestaña *\_Bootscreen.h*.

La siguiente función para configurar es el puerto de comunicación para transmitir y recibir información. El firmware tiene configurado la comunicación en el puerto 0 y un baudrate de 250000, este puerto y velocidad deben ser tomados en cuenta posteriormente para la comunicación con el controlador del control alámbrico manual.

```

// @section machine
/**
 * Select which serial port on the board will be used for communi
 * This allows the connection of wireless adapters (for instance)
 * Serial port 0 is always used by the Arduino bootloader regardl
 *
 * :[0, 1, 2, 3, 4, 5, 6, 7]
 */
#define SERIAL_PORT 0
/**
 * This setting determines the communication speed of the printer
 *
 * 250000 works in most cases, but you might try a lower speed if
 * you commonly experience drop-outs during host printing.
 * You may try up to 1000000 to speed up SD file transfer.
 *
 * :[2400, 9600, 19200, 38400, 57600, 115200, 250000, 500000, 100
 */
#define BAUDRATE 250000

```

Figura 2.43. Configuración de puerto y velocidad de comunicación.

La siguiente configuración a realizarse es establecer el tipo de placa de control CNC conectada al Arduino Mega, las placas que son compatibles con el firmware se encuentran en la pestaña *boards.h* como se muestra en la Figura 2.44.

```

#define BOARD_GEN7_14          13 // Gen7 v1.4
#define BOARD_CNCONTROLS_11    111 // Cartesio CN Controls V11
#define BOARD_CNCONTROLS_12    112 // Cartesio CN Controls V12
#define BOARD_CHEAPTRONIC      2 // Cheaptronic v1.0
#define BOARD_CHEAPTRONIC_V2   21 // Cheaptronic v2.0
#define BOARD_SETHI             20 // Sethi 3D_1
#define BOARD_MIGHTYBOARD_REVE 200 // Makerbot Mightyboard Revision E
#define BOARD_RAMPS_OLD         3 // MEGA/RAMPS up to 1.2
#define BOARD_RAMPS_13_EFB      33 // RAMPS 1.3 (Power outputs: Hotend, F
#define BOARD_RAMPS_13_EEB      34 // RAMPS 1.3 (Power outputs: Hotend0,
#define BOARD_RAMPS_13_EFF      35 // RAMPS 1.3 (Power outputs: Hotend, F
#define BOARD_RAMPS_13_EEF      36 // RAMPS 1.3 (Power outputs: Hotend0,
#define BOARD_RAMPS_13_SF       38 // RAMPS 1.3 (Power outputs: Spindle,
#define BOARD_FELIX2            37 // Felix 2.0+ Electronics Board (RAMPS
#define BOARD_RIGIDBOARD        42 // Invent-A-Part RigidBoard
#define BOARD_RIGIDBOARD_V2     52 // Invent-A-Part RigidBoard V2
#define BOARD_RAMPS_14_EFB      43 // RAMPS 1.4 (Power outputs: Hotend, F
#define BOARD_RAMPS_14_EEB      44 // RAMPS 1.4 (Power outputs: Hotend0,
#define BOARD_RAMPS_14_EFF      45 // RAMPS 1.4 (Power outputs: Hotend, F
#define BOARD_RAMPS_14_EEF      46 // RAMPS 1.4 (Power outputs: Hotend0,
#define BOARD_RAMPS_14_SF       48 // RAMPS 1.4 (Power outputs: Spindle,

```

Figura 2.44. Placas soportadas por Marlin.

La placa acoplada al Arduino Mega es una RAMPS 1.4, al tener una salida para el control de velocidad del motor husillo, la *BOARD RAMPS 14 SF* permite, esta placa es introducida en la opción *MOTHERBOARD* como se muestra en la Figura 2.45.

```

// The following define selects which electronics board you have.
// Please choose the name from boards.h that matches your setup
#ifndef MOTHERBOARD
#define MOTHERBOARD BOARD_RAMPS_14_SF
#endif

```

Figura 2.45. Configuración de placa.

Ahora se configura la fuente de alimentación, como los controladores tienen el suministro de voltaje de una fuente externa, se coloca 0 en esta sección, como se muestra en la Figura 2.46.

```

*
* :{ 0:'No power switch', 1:'ATX', 2:'X-Box 360' }
*/
#define POWER_SUPPLY 0

#if POWER_SUPPLY > 0
// Enable this option to leave the PSU off at startup.
// Power to steppers and heaters will need to be turned on with M80.
//#define PS_DEFAULT_OFF
#endif

```

Figura 2.46. Configuración de fuente de alimentación.

Los sensores mecánicos para el límite del área de trabajo a usar se configuran en la sección *endstops*, dependiendo el número de sensores se habilitan. En la Figura 2.47 se

muestra los finales de carrera declarados, además se activa las resistencias pullups internas del Arduino para evitar fallas en la lectura de los sensores.

```
// @section homing
// Specify here all the endstop connectors that are connected to any endstop
// Almost all printers will be using one per axis. Probes will use one or more
// extra connectors. Leave undefined any used for non-endstop and non-probe
#define USE_XMIN_PLUG
#define USE_YMIN_PLUG
#define USE_ZMIN_PLUG
#define USE_XMAX_PLUG
#define USE_YMAX_PLUG
#define USE_ZMAX_PLUG

// coarse Endstop Settings
#define ENDSTOPPULLUPS // Comment this out (using // at the start of the line)
```

Figura 2.47. Configuración de finales de carrera.

A continuación, se configura la sección de movimiento, en la cual se establece los pasos por unidad, la velocidad de avance y la aceleración, estos parámetros dependen de los sistemas de movimiento de cada eje de la máquina. Primero se establecen los pasos por cada mm. Para esto es necesario saber cuántos mm se recorre por cada vuelta del tornillo.

El tornillo tiene 8 hilos por pulgada [70], necesita de 8 vueltas para recorrer 25.4 mm, en cada vuelta recorre 3.175 mm, el controlador del motor PAP del eje X tiene configurado los micro pasos a 1/2, es decir, cada paso recibido es dividido en 2 pasos pequeños, el motor tiene una resolución de 1.8° por paso [71], quiere decir que necesita de 200 pasos para girar una vuelta, el controlador entrega 400 pasos para realizar una vuelta. Si realizamos una regla de tres para determinar el número de pasos necesarios para recorrer 1 mm, tenemos que:

$$\begin{array}{rcl} 400 \text{ pasos} & = & 3.175 \text{ mm} \\ X & = & 1 \text{ mm} \end{array}$$

Despejando X se encuentra los pasos necesarios para recorrer 1mm. Al dividir 400 entre 3.175 se tiene:

$$X = 125.98 \approx 126 \text{ pasos}$$

Para verificar si este dato es correcto es necesario realizar un movimiento por parte de la máquina, esta parte debe realizarse cuando todos los parámetros se encuentren configurados y el firmware cargado en el controlador. El controlador del eje Y tiene la misma configuración que el controlador del eje X por lo que los cálculos de los pasos son los mismos. El eje Z por su lado, posee una diferente configuración y el controlador entrega 200 pasos por revolución del motor. Esto indica que para el cálculo de los pasos/mm de este eje es la división de 200 entre 3.175:

$$X=Y=126 \text{ pasos}$$

$$Z=62.99 \approx 63 \text{ pasos}$$

Estos datos son establecidos en la opción de pasos por unidad, como se muestra en la Figura 2.48.

```
* Default Axis Steps Per Unit (steps/mm)
* Override with M92
*
*                               X, Y, Z, E0 [, E1[, E2[, E3[, E4]]]
*/
#define DEFAULT_AXIS_STEPS_PER_UNIT {126, 126, 63, 500 }
```

Figura 2.48. Configuración de pasos/mm.

La opción de avance permite establecer la velocidad por defecto a la cual se mueve la máquina, esta velocidad se mantiene vigente mientras no se especifique otro valor con el comando F, al enviar este código la velocidad de avance se reemplaza con este valor hasta que se establezca otro o se reinicie la máquina. En esta sección también se configura la aceleración por defecto para el movimiento de los motores. La velocidad de avance se encuentra en mm/s y la aceleración en mm/s<sup>2</sup>. En la Figura 2.49 se muestra los avances y aceleraciones seleccionados para cada uno de los ejes.

```

/**
 * Default Max Feed Rate (mm/s)
 * Override with M203
 *
 * X, Y, Z, E0 [, E1[, E2[, E3[, E4]]]
 */
#define DEFAULT_MAX_FEEDRATE      { 800, 800, 500, 25 }

/**
 * Default Max Acceleration (change/s) change = mm/s
 * (Maximum start speed for accelerated moves)
 * Override with M201
 *
 * X, Y, Z, E0 [, E1[, E2[, E3[, E4]]]
 */
#define DEFAULT_MAX_ACCELERATION  { 500, 500, 400, 10000 }

/**
 * Default Acceleration (change/s) change = mm/s
 * Override with M204
 *
 * M204 P Acceleration
 * M204 R Retract Acceleration
 * M204 T Travel Acceleration
 */
#define DEFAULT_ACCELERATION      500 // X, Y, Z and E acceleration
#define DEFAULT_RETRACT_ACCELERATION 600 // E acceleration for retracts
#define DEFAULT_TRAVEL_ACCELERATION 600 // X, Y, Z acceleration for travel

```

Figura 2.49. Configuración de Avance por defecto

Ahora, se procede a configurar el área de trabajo de la máquina, estos parámetros son establecidos en mm y por cada eje. El área de trabajo de la máquina es de 800x800mm, en la configuración se establece 800 para el eje X y Y. En la Figura 2.50 se muestra la configuración del área de trabajo.

```

// @section machine

// The size of the spacework (mm)
#define X_BED_SIZE 800
#define Y_BED_SIZE 800

```

Figura 2.50. Configuración del área de trabajo.

Se pueden establecer límites por software como protección adicional, al activar esta opción el firmware no permitirá que existan movimiento fuera de los límites establecidos.

```

// @section machine

// The size of the spacework (mm)
#define X_BED_SIZE 800
#define Y_BED_SIZE 800

// Travel limits (mm) after homing, corresponding to endstop posit
#define X_MIN_POS 0
#define Y_MIN_POS 0
#define Z_MIN_POS 0
#define X_MAX_POS X_BED_SIZE
#define Y_MAX_POS Y_BED_SIZE
#define Z_MAX_POS 200

// If enabled, axes won't move below MIN_POS in response to moveme
#define MIN_SOFTWARE_ENDSTOPS
// If enabled, axes won't move above MAX_POS in response to moveme
#define MAX_SOFTWARE_ENDSTOPS

```

Figura 2.51. Activación de límites por software

En la Figura 2.58 se muestra la activación de los límites por software, además se establecen los límites mínimos y máximos, se establece el límite mínimo de 0 en los ejes X, Y y Z, los límites máximos son establecidos por el área de trabajo

La siguiente sección para configurar es la pantalla LCD y sus características, el primer paso en esta sección es establecer el idioma en el cual se va a mostrar la información. Este campo configura con el idioma español, existen varios idiomas que son soportados por Marlin. En la Figura 2.52 se muestra la configuración del idioma.

```

/**
 * LCD LANGUAGE
 *
 * Select the language to display on the LCD. These languages are available
 *
 * en, an, bg, ca, cn, cz, cz_utf8, de, el, el-gr, es, eu, fi, fr, gl, h
 * it, kana, kana_utf8, nl, pl, pt, pt_utf8, pt-br, pt-br_utf8, ru, sk_u
 * tr, uk, zh_CN, zh_TW, test
 *
 * :{ 'en':'English', 'an':'Aragonese', 'bg':'Bulgarian', 'ca':'Catalan', '
 */
#define LCD_LANGUAGE es

```

Figura 2.52. Selección de idioma para LCD.

Si se tiene una LCD con soporte para gráficos se puede activar la opción Full Graphics Display, con esta opción se puede mostrar imágenes en la LCD. A continuación, se muestra en la Figura 2.53 la activación de esta opción.

```

/**
 * LCD TYPE
 *
 * Enable ULTRA_LCD for a 16x2, 16x4, 20x2, or 20x4 character-based LCD.
 * Enable DOGLCD for a 128x64 (ST7565R) Full Graphical Display.
 * (These options will be enabled automatically for most displays.)
 *
 * IMPORTANT: The U8glib library is required for Full Graphic Display!
 *             https://github.com/olikraus/U8glib\_Arduino
 */
#define ULTRA_LCD // Character based
#define DOGLCD // Full graphics display

```

Figura 2.53. Activación de opción Full Graphics Display.

A continuación, se configura los elementos adicionales que posee la LCD, estos elementos pueden ser, ranura para tarjeta SD, encoder para desplazarse por el menú, bocina, etc.

Si la LCD cuenta con una ranura SD para lectura de archivos, se puede activar la opción de *SDSUPPORT*, esta herramienta permite realizar lecturas de archivos de código G desde una tarjeta de memoria SD. En la Figura 2.54 se muestra cómo definir la opción de *SDSUPPORT*.

```

/**
 * SD CARD
 *
 * SD Card support is disabled by default. If your controller has an SD slot
 * you must uncomment the following option or it won't work.
 *
 */
#define SDSUPPORT

```

Figura 2.54. Activación de opción SDSUPPORT.

Gran parte de los controladores LCD para el control de una máquina CNC posee un encoder con pulsador el cual permite al usuario desplazarse por el menú y seleccionar la opción deseada. La configuración de este elemento esta dado por el giro del encoder, si es necesario que se desplace las opciones del menú de arriba a abajo cuando el encoder es girado en sentido horario, es necesario activar la opción *REVERSE\_ENCODER\_DIRECTION*, como se muestra en la Figura 2.55.



```

/**
 * Encoder Direction Options
 *
 * Test your encoder's behavior first with both options disabled.
 *
 * Reversed Value Edit and Menu Nav? Enable REVERSE_ENCODER_DIRECTION.
 * Reversed Menu Navigation only?   Enable REVERSE_MENU_DIRECTION.
 * Reversed Value Editing only?     Enable BOTH options.
 */
//
// This option reverses the encoder direction everywhere.
//
// Set this option if CLOCKWISE causes values to DECREASE
//|
#define REVERSE_ENCODER_DIRECTION

```

Figura 2.55. Configuración de encoder.

El siguiente elemento para configurar es la bocina, si el controlador LCD posee este dispositivo se activa la opción `SPEAKER`, como se muestra en la Figura 2.56.

```

// SPEAKER/BUZZER
//
// If you have a speaker that can produce tones, enable it here.
// By default Marlin assumes you have a buzzer with a fixed frequency.
//
#define SPEAKER

```

Figura 2.56. Activación de la opción `SPEAKER`.

Si se necesita realizar el “homing” de los ejes individualmente mediante el menú de la LCD se puede activar la opción de `INDIVIDUAL_AXIS_HOMING`, al hacerlo se mostrará en el menú LCD una opción de homing para cada eje. En la Figura 2.57 se muestra la activación de esta opción.

```

// Individual Axis Homing
//
// Add individual axis homing items (Home X, Home Y, and Home Z) to the LCD
//
#define INDIVIDUAL_AXIS_HOMING_MENU

```

Figura 2.57. Activación de la opción `INDIVIDUAL_AXIS_HOMING`

Ahora se procede a definir el tipo de controlador de la LCD, la pantalla adquirida es RepRapDiscount FULL GRAPHIC Smart Controller, por lo que se procede a activar la opción con este nombre, como se ilustra en la Figura 2.58.

```

// RepRapDiscount Smart Controller.
// http://reprap.org/wiki/RepRapDiscount\_Smart\_Controller
//
// Note: Usually sold with a white PCB.
//
//#define REPRAP_DISCOUNT_SMART_CONTROLLER

//
// GADGETS3D G3D LCD/SD Controller
// http://reprap.org/wiki/RAMPS\_1.3/1.4\_GADGETS3D\_Shield\_with\_Panel
//
// Note: Usually sold with a blue PCB.
//
//#define G3D_PANEL

//
// RepRapDiscount FULL GRAPHIC Smart Controller
// http://reprap.org/wiki/RepRapDiscount\_Full\_Graphic\_Smart\_Controller
//
#define REPRAP_DISCOUNT_FULL_GRAPHIC_SMART_CONTROLLER

```

Figura 2.58. Configuración del tipo de controlador de LCD.

Estas son las configuraciones generales del firmware Marlin, existen configuraciones avanzadas, las cuales pueden ser modificadas, pero se recomienda que las realice por personal capacitado. Algunas configuraciones avanzadas serán modificadas para establecer funciones específicas de la máquina CNC. Estas configuraciones se encuentran en la pestaña *Configuration\_adv.h*.

Una de las funciones más importantes de la máquina es el “homing”, cuando un eje realiza esta función, se acerca al final de carrera a una velocidad, existe una función adicional que permite un retroceso cuando el final de carrera sea activado, posteriormente se acerca nuevamente al sensor pero a una velocidad reducida, la distancia a retroceder como la velocidad de acercamiento puede establecerse mediante la opción HOME BUMP MM y HOMING BUMP DIVISOR respectivamente, los valores para estas opciones se establecen por cada eje como se muestra en la Figura 2.59.

```

// @section homing

//homing hits the endstop, then retracts by this distance, before
#define X_HOME_BUMP_MM 4
#define Y_HOME_BUMP_MM 4
#define Z_HOME_BUMP_MM 3
#define HOMING_BUMP_DIVISOR {8, 8, 2} // Re-Bump Speed Divisor (I
//#define QUICK_HOME //if this is defined, if both x and y are to

```

Figura 2.59. Definición de distancia de retroceso y divisor de velocidad.

Otra opción para configurar es la *SPINDLE LASER ENABLE*, permite que el firmware envíe señal PWM para el control de velocidad de un motor husillo o potencia de Láser. Esta opción tiene varios parámetros, los cuales se ven configurados en la Figura 2.60.

```

* Spindle & Laser control
*
* Add the M3, M4, and M5 commands to turn the spindle/laser on and off, and
* to set spindle speed, spindle direction, and laser power.
*
* SuperPid is a router/spindle speed controller used in the CNC milling community.
* Marlin can be used to turn the spindle on and off. It can also be used to set
* the spindle speed from 5,000 to 30,000 RPM.
*
* You'll need to select a pin for the ON/OFF function and optionally choose a 0-5V
* hardware PWM pin for the speed control and a pin for the rotation direction.
*
* See http://marlinfw.org/docs/configuration/laser\_spindle.html for more config details.
*/
#define SPINDLE_LASER_ENABLE
#if ENABLED(SPINDLE_LASER_ENABLE)

  #define SPINDLE_LASER_ENABLE_INVERT false // set to "true" if the on/off function is
  #define SPINDLE_LASER_PWM true // set to true if your controller supports
  #define SPINDLE_LASER_PWM_INVERT false // set to "true" if the speed/power goes
  #define SPINDLE_LASER_POWERUP_DELAY 3000 // delay in milliseconds to allow the spin
  #define SPINDLE_LASER_POWERDOWN_DELAY 3000 // delay in milliseconds to allow the spin
  #define SPINDLE_DIR_CHANGE true // set to true if your spindle controller
  #define SPINDLE_INVERT_DIR false
  #define SPINDLE_STOP_ON_DIR_CHANGE true // set to true if Marlin should stop the s

```

Figura 2.60. Configuración de sección Spindle/Láser.

Para relacionar las revoluciones máximas del husillo con la señal PWM que suministra el firmware, es necesario realizar un cálculo simple que se muestra en la Figura 2.61.

```

/**
 * The M3 & M4 commands use the following equation to convert PWM duty cycle to speed/po
 *
 * SPEED/POWER = PWM duty cycle * SPEED_POWER_SLOPE + SPEED_POWER_INTERCEPT
 * where PWM duty cycle varies from 0 to 255
 *
 * set the following for your controller (ALL MUST BE SET)
 */

```

Figura 2.61. Fórmula para el cálculo de PWM en relación con las revoluciones del husillo.

El motor husillo otorga un máximo de 12000 revoluciones, el PWM máximo es de 255, el valor de *SPEED\_POWER\_INTERCEPT* es 0, entonces se va a obtener *SPEED\_POWER\_SLOPE*, despejando esta variable se tiene:

$$SPEED\_POWER\_SLOPE = \frac{SPEED}{PWM \text{ duty cycle}} = \frac{12000}{255} = 47.06$$



```

/
 * User-defined menu items that execute custom GCode
 */
#define CUSTOM_USER_MENUS
#if ENABLED(CUSTOM_USER_MENUS)
  #define USER_SCRIPT_DONE "M117 Acción Realizada"
  #define USER_SCRIPT_AUDIBLE_FEEDBACK
  #define USER_SCRIPT_RETURN // Return to status screen after a script

  #define USER_DESC_1 "Encender Spindle 8000 RPMs"
  #define USER_GCODE_1 "M3 S8000"

  #define USER_DESC_2 "Apagar Spindle"
  #define USER_GCODE_2 "M3 S100"

  #define USER_DESC_3 "Set 0 pieza"
  #define USER_GCODE_3 "G92 X0 Y0 Z0"

  #define USER_DESC_4 "Quitar Limites Soft."
  #define USER_GCODE_4 "M211 S0"

  #define USER_DESC_5 "Act. Limites Soft"
  #define USER_GCODE_5 "M211 S1"
#endif

```

*Figura 2.63. Implementación de Funciones en Custom User Menu*

Con las configuraciones realizadas, se carga el firmware al controlador Arduino Mega, una vez subido es posible comprobar el correcto funcionamiento de la máquina, la calibración de los movimientos en cada eje, verificar que los parámetros establecidos cumplan con su propósito.

## CAPÍTULO III

### 3. PRUEBAS Y RESULTADOS

En este capítulo se realizan las pruebas necesarias para verificar el correcto funcionamiento de la máquina.

#### 3.1. Pruebas de desplazamiento

Las pruebas de desplazamiento son el primer paso para determinar el correcto funcionamiento de los sistemas de movimiento, se realizan de forma individual, es decir, un eje a la vez, se utiliza un programa desarrollado en el IDE de Arduino, la placa UNO de la misma plataforma y un controlador para motor PAP.

En la Figura 3.1 se muestra la estructura completa de la máquina CNC y se determina el nombre de cada uno de los ejes.

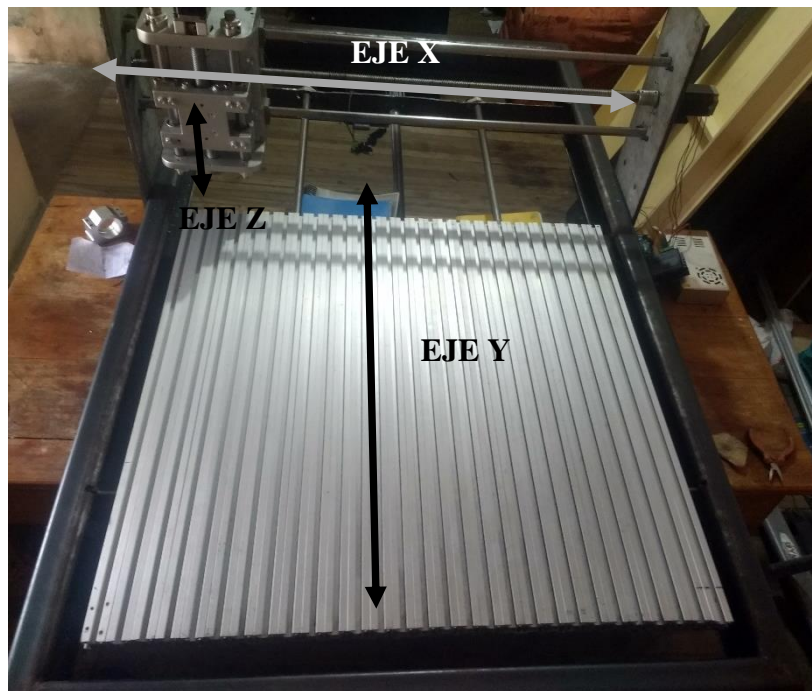


Figura 3.1. Estructura completa de la máquina CNC.

El programa consiste en la activación secuencial de las bobinas del motor, esta secuencia es enviada al controlador, el cual se encarga de interpretarlos y enviar los pulsos necesarios al motor. Para cada uno de los ejes el código cambia ligeramente, precisamente en la

cantidad de pasos, pues cada uno de los ejes tiene una distancia diferente a recorrer. A continuación, se muestra los códigos empleados para realizar las pruebas de desplazamiento. El código empleado se encuentra en el Anexo 2.

Para preservar la integridad de los elementos mecánicos inicialmente se realiza el desplazamiento a velocidad reducida durante cierto número de pasos luego se realiza un pequeño aumento de velocidad, este proceso se lo realiza 4 ocasiones, luego de esto la velocidad se mantiene durante la mayor parte del trayecto y al final se realiza una desaceleración igualmente en 4 fases hasta detenerse por completo. La velocidad de desplazamiento está determinada por el tiempo de espera entre los accionamientos de las bobinas, si se emplea un tiempo de espera alto, la velocidad será reducida.

Para evitar tener que cargar otro programa para realizar el desplazamiento de retroceso se implementó otro subprograma en el cual se realiza el proceso de retroceso, los procesos son controlados por una variable la cual tendrá el valor de 0 o 1, siendo 1 para avanzar y 0 para retroceder, esta variable es leída por un puerto digital, que a su vez está conectado un botón a 5 V. esto permite el control de los procesos desde hardware.

Después de realizar las pruebas de desplazamiento en cada eje, se corrobora que los sistemas de movimiento funcionan correctamente. Después se realiza la misma prueba con el firmware Marlin cargado en el controlador Arduino Mega, donde los sistemas de movimiento funcionan correctamente

### **3.2. Pruebas de tolerancia del material**

Las pruebas se realizan a diferentes velocidades de avance y revoluciones para determinar el comportamiento del material que se va a cortar. El material que se utiliza es tubo rectangular de aluminio de 2mm de espesor.

Primero se realiza corte por desbaste de material en varias pasadas, es decir, la fresa va ingresando poco a poco hasta cortar el material. A continuación, se realiza una tabla con los datos de cada prueba. Los valores de avance y revoluciones del husillo son tomados de los datos recomendados que se encuentran en el trabajo de grado “Máquina de Control

Numérico Computacional para corte de planchas de aluminio por desbaste de material: Diseño”.

Tabla 3-I. Prueba de tolerancia del material.

<b>Prueba</b>	Profundidad de inserción (mm)	Avance (mm/min)	Revoluciones del husillo (rpm)	Observación
<b>1</b>	0,3	40	9000	El material acepta bien este corte
<b>2</b>	0,5	35	9500	El material presenta imperfecciones por vibración moderada.
<b>3</b>	1,0	30	9500	El material presenta muchas imperfecciones por una vibración considerable.

Los datos proporcionados por la Tabla 3-I, muestran que el corte realizado en varias pasadas es más preciso que cortes profundos. A mayor profundidad de corte el material presenta más imperfecciones y las dimensiones esperadas fallan más que al realizar cortes ingresando menor profundidad por pasada.

### 3.3. Obtención de código G de un modelo

Para obtener el código G de un modelo existen varias maneras, una de ellas es realizarlo a mano, esto es recomendable cuando el diseño no es muy complejo, por ejemplo, para realizar figuras geométricas o elementos que o contengan muchos detalles. Para modelos que tengan un alto grado de complejidad, se puede obtener el código G con la ayuda de software CAD/CAM, permite diseñar el modelo y después obtener el código utilizando una herramienta CAM.

Se utiliza el programa llamado Inkscape, es un software de vectores gráficos de calidad profesional para Windows, Mac OS X y GNU/Linux. Es usado por diseñadores profesionales y aficionados de todo el mundo para crear una gran variedad de gráficos como ilustraciones, iconos, logos, diagramas, mapas y diseños web. Inkscape es un software libre y de código abierto, que utiliza SVG (Scalable Vector Graphic), el estándar abierto de W3C, como formato nativo [72].



Posee varios complementos que lo hacen muy versátil, las herramientas son desarrolladas por la comunidad del proyecto Inkscape. Una complemento de gran utilidad es Gcodetools, permite convertir trayectorias vectoriales a código G, utilizando interpolación circular o movimiento lineal según sea requerido. Permite que Inkscape sea usado como un programa CAM [73].

### 3.3.1. Pasos para obtener código G en Inkscape de una imagen.

Una ventaja importante que tiene Inkscape es poder realizar un diseño en la interfaz principal para su posterior conversión a código G, también permite cargar una imagen y con ayuda de una herramienta de vectorización automática, obtener el diseño para convertirlo a código G. El primer paso es ubicar la imagen en la interfaz principal, este procedimiento se lo puede realizar arrastrando la imagen hacia el área de trabajo del programa o mediante la opción **Importar** que se encuentra en el menú **Archivo**. En la Figura 3.2 se muestra una imagen insertada en la interfaz del software Inkscape.

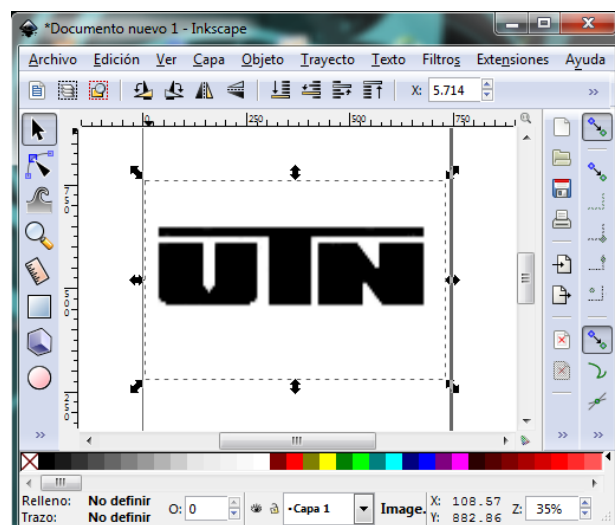


Figura 3.2. Inserción de imagen en el área de trabajo de Inkscape

Después de realizar la inserción de la imagen en el área de trabajo se puede cambiar el tamaño y orientación de la imagen según sea necesario, el siguiente paso es vectorizar la imagen, esto se realiza seleccionando la opción **Vectorizar mapa de bits** que se encuentra en el menú **Trayecto**. En la Figura 3.3 se muestra el cuadro de dialogo que se muestra al seleccionar esta opción.

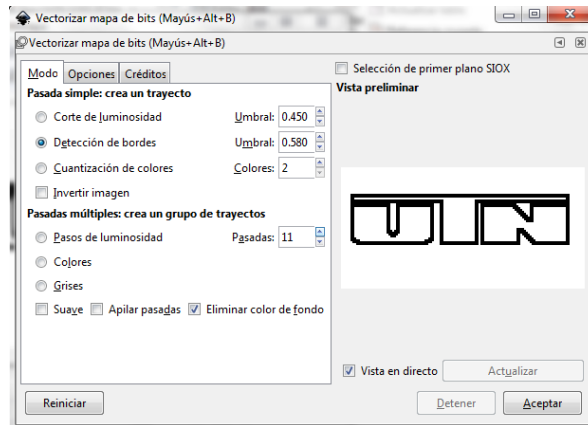


Figura 3.3. Cuadro de diálogo Vectorizar mapa de bits

En el cuadro de dialogo mostrado existen varias opciones para vectorizar la imagen, existe una vista preliminar, permitiendo visualizar el resultado. Una vez que se determina el tipo de vectorización y las opciones adicionales que ofrece la herramienta, se da clic en aceptar.

Una vez que se obtuvo la imagen vectorizada, es necesario establecer unos parámetros antes que se puede realizar la conversión a código G, el primer paso es determinar los puntos de orientación. Esta opción nos permite ubicar el 0 pieza y determinar la profundidad máxima de mecanizado. En la Figura 3.4 se muestra el cuadro de dialogo en el cual se introduce la superficie del eje Z, la profundidad total del mecanizado y las unidades en las que se va a trabajar.

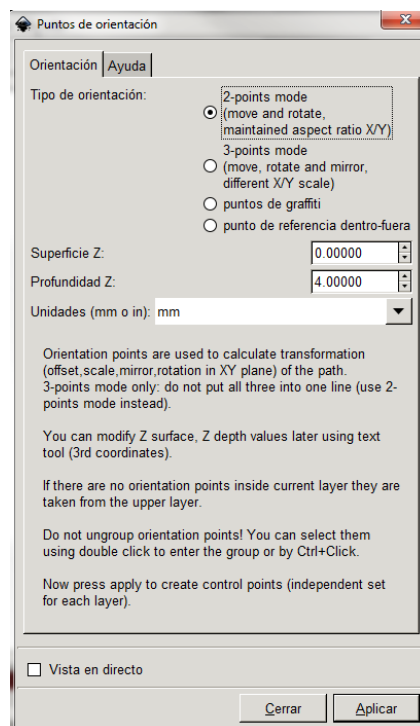


Figura 3.4. Definición de puntos de orientación y profundidad de mecanizado.

Ahora se configura las opciones de la herramienta, se selecciona la opción **Biblioteca de herramientas** del submenú **Gcodetools** que se encuentra en el menú **Extensiones**. Se muestra un cuadro de dialogo como se muestra en la Figura 3.5, aquí se establecen la geometría de la herramienta que se va a usar.

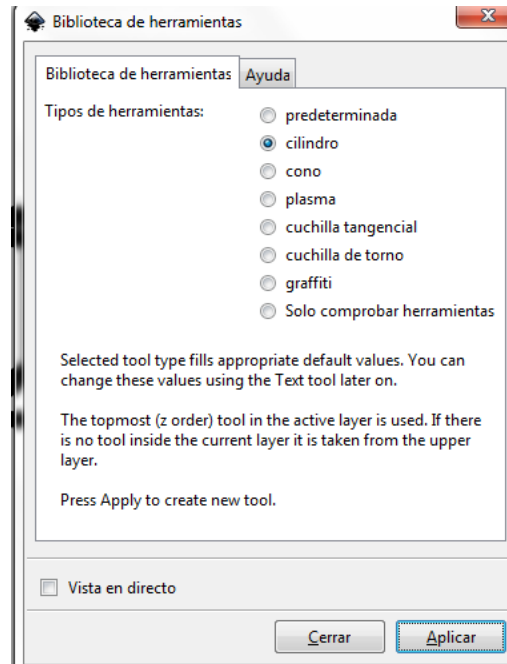


Figura 3.5. Cuadro de diálogo Biblioteca de Herramientas.

Se genera un cuadro de dialogo con la herramienta seleccionada como se muestra en la Figura 3.6.

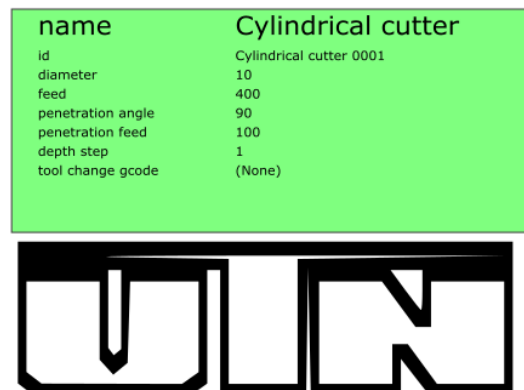


Figura 3.6. Generación de Herramienta

En el cuadro generado se muestran las características de la herramienta que pueden ser modificadas de acuerdo con los parámetros de funcionamiento de la máquina. Se establece el diámetro de la fresa, la velocidad de avance, la velocidad de inserción y la profundidad de inserción por pasada. Una vez realizada las modificaciones correspondientes se puede

generar el código G seleccionando la opción **Trayecto a Gcode** que se encuentra en la opción **Gcodetools**. En la Figura 3.7 se muestra el cuadro de dialogo mostrado al seleccionar esta opción.

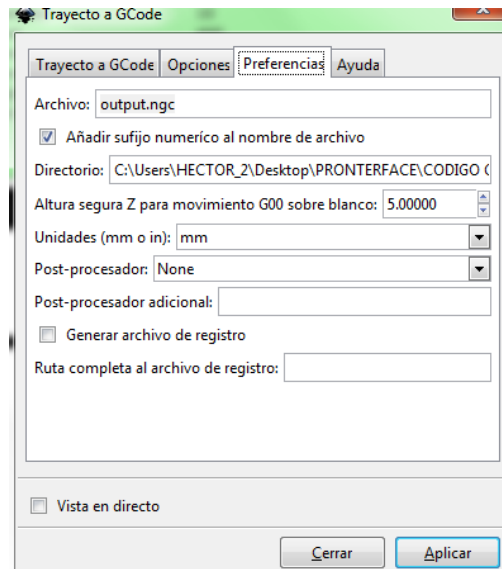


Figura 3.7. Opción Trayecto a GCode

En esta sección se puede seleccionar el nombre del archivo, la dirección en donde se desea guardar, las unidades a utilizar, entre otras opciones. Una vez modificadas las opciones necesarias se procede a dar clic en el botón **Aplicar** para generar el Archivo con el código G. En la Figura 3.8 se muestra el archivo generado con el Código G, si es necesario adicionar líneas adicionales para que sea compatible con la máquina, se las puede incluir abriendo el archivo en un editor de texto. El código generado se encuentra en el Anexo 2.

```

UTN.ngc: Bloc de notas
Archivo Edición Formato Ver Ayuda
%
(Header)
(Generated by gcodetools from Inkscape.)
(Using default header. To add your own header create file
"header" in the output dir.)
M3
(Header end.)
G21 (All units in mm)

(Start cutting path id: path10006)
(Change tool to Default tool)

G00 Z5.000000
G00 X344.256531 Y712.147234

G01 Z0.000000 F100.0(Penetrare)
G03 X225.105155 Y689.813355 Z0.000000 I-0.923881 J-324.074679
F400.000000
G03 X118.218038 Y622.530107 Z0.000000 I123.021938 J-313.977200
G03 X50.066497 Y535.230917 Z0.000000 I240.307775 J-257.851601
G03 X11.830182 Y430.117946 Z0.000000 I314.618167 J-173.957744
G03 X7.024656 Y396.588043 Z0.000000 I330.376782 J-64.459123
G03 X5.265760 Y358.979756 Z0.000000 I414.118483 J-38.213120
G03 X7.650708 Y314.698319 Z0.000000 I401.438927 J-0.583892
G03 X14.590117 Y273.829122 Z0.000000 I348.694350 J38.183022
G03 X40.064064 Y200.558393 Z0.000000 I352.913981 J81.633568
G03 X79.688092 Y136.223188 Z0.000000 I310.663426 J146.967672
G03 X187.302232 Y44.763723 Z0.000000 I274.641421 J214.111047

```

Figura 3.8. Código G generado.

Después de realizar las modificaciones respectivas, se guarda el archivo y se puede cargar el archivo en la máquina o en un simulador de código G para verificar que el resultado es el adecuado.

### 3.4. Pruebas de corte de material

Para las pruebas de corte de material se realiza el corte de un cuadrado, el material seleccionado es tubo rectangular de aluminio, debido a que el costo de adquisición de una plancha completa es muy elevado.

Al ser una figura geométrica de baja complejidad, el código G es generado a mano mediante el uso de un editor de texto, la primera prueba se efectúa mediante varias pasadas para visualizar el comportamiento de la máquina y la herramienta. Para verificar el código G generado, se utiliza un simulador online llamado NC Viewer, en el que también se puede escribir el código a mano y simularlo. En la Figura 3.9 se muestra la simulación del código G.

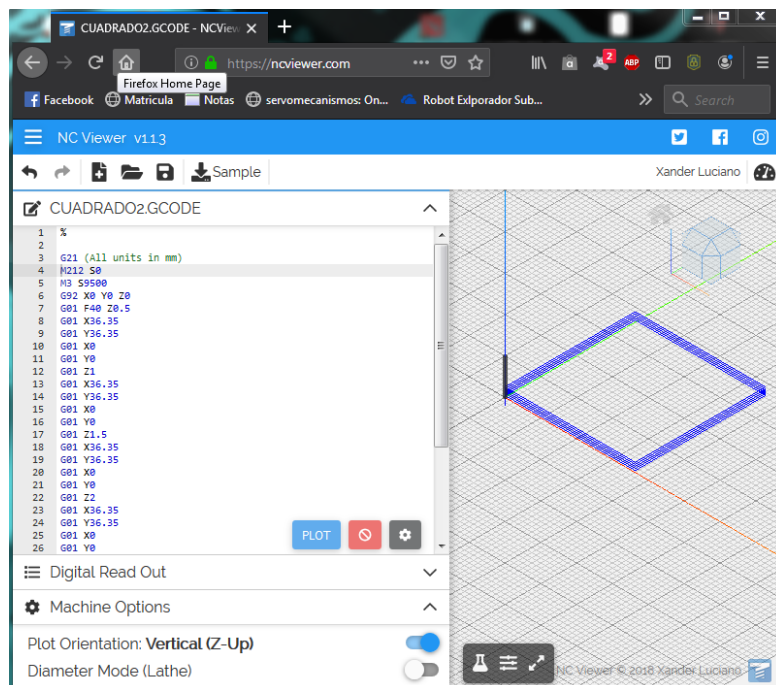


Figura 3.9. Simulación código G mediante NC Viewer [74].

Una vez realizada la simulación, se verifica que el código creado realiza la pieza esperada, después puede ser grabada para que sea realizada por la máquina. Los archivos se almacenan en una tarjeta SD porque el controlador LCD posee una ranura para este tipo de dispositivo de almacenamiento.

Antes de cargar el programa, es necesario tomar el cero pieza, esta acción se la realiza mediante el uso del control alámbrico manual, si en el código que se va a cargar se encuentra la línea “G92 X0 Y0 Z0” podemos ejecutar el archivo directamente después de llevar la herramienta al cero pieza, pero si no se encuentra en el archivo, se debe seleccionar el cero pieza antes de realizar el corte, para ello se ingresa al menú de la pantalla LCD, submenú Comandos Personalizados, y se selecciona la opción de Set 0 pieza.

Al momento de colocar la memoria SD en la ranura del controlador, se muestra el mensaje “Tarjeta Colocada”, este mensaje indica que se puede acceder a los archivos, ingresamos al menú principal y posteriormente al submenú Abrir de SD, se muestran los archivos almacenados ordenados por fecha de modificación. Seleccionar el archivo y la máquina empezara a realizar el trabajo de corte. En la Figura 3.10 se muestra los archivos almacenados en la memoria SD.



Figura 3.10. Selección de archivo desde SD.

Una vez seleccionado el archivo, la máquina empieza a realizar las instrucciones que se encuentren. En la Figura 3.11 se muestra a la máquina realizando el corte de la pieza.



Figura 3.11. Corte de pieza.

### 3.5. Tabla de resultados

Después de realizar las pruebas de corte se compara las dimensiones de la pieza cortada con las dimensiones esperadas, el cuadrado debe medir 30x30mm. A continuación, se elabora una tabla de resultados. Cabe destacar que el margen de error está establecido en  $\pm 0,5\text{mm}$  de tolerancia, es decir  $\pm 1,66\%$ . En la Tabla 3-II se muestra las piezas cortadas con el respectivo porcentaje de falla.

Tabla 3-II. Tabla de resultados

Pieza	Valor medido (mm)	Porcentaje de falla (X,Y)	Observaciones
1	33,97x30,66	13% - 2,2%	El valor medido en el eje X se aleja mucho del esperado y el valor en el eje Y aun no cubre los requerimiento, por lo que es necesario efectuar una recalibración del desplazamiento de los ejes.
2	28,73x29,48	4,23% - 17,3%	Después de la calibración se logra una mejoría, pero no satisface los requerimientos, se realiza otra calibración más minuciosa.
3	30,28x30,16	0,93% - 0,53%	Esta pieza cumple con los parámetros establecidos.

En el Anexo 4 se muestran imágenes de las piezas cortadas con las mediciones respectivas.

## 4. CONCLUSIONES Y RECOMENDACIONES

### 4.1. Conclusiones

- La estructura de la máquina CNC se construyó en base a los planos establecidos en el trabajo de grado “Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Diseño” con un área de trabajo de 80x80cm.
- Los sistemas de movimiento de cada eje se construyeron con elementos adquiridos en el país y extranjero, las piezas diseñadas se elaboraron mediante el método más accesible desde el punto de vista económico, uno de los factores más importantes.
- Con la realización de moldes en MDF por corte laser para las piezas diseñadas, se obtuvo una precisión aceptable con una tolerancia de  $\pm 0.3\text{mm}$ .
- El uso del controlador Arduino Mega ayudo a no depender de un computador para la configuración del cero pieza.
- Las configuraciones realizadas en el firmware Marlin permitieron el control de la velocidad del husillo mediante código G y la personalización de funciones.
- Las pruebas de desplazamiento y corte ayudaron a realizar una correcta calibración de los pasos por mm de cada uno de los ejes y determinar una tolerancia de corte mínima de  $\pm 0.16\text{mm}$  y una máxima de  $\pm 0.30\text{mm}$ , estos rangos están dentro del parámetro establecido en la fase de diseño.



## 4.2. Recomendaciones

- Construir las piezas diseñadas de los sistemas de movimiento mediante procesos de mecanizado para asegurar un alto grado de precisión y evitar fallas al momento de implementarlos.
- Adquirir una mesa de trabajo que sea un solo conjunto para evitar problemas de desnivelaciones en el área de trabajo.
- Si se desea una velocidad de avance elevado al momento de realizar un corte, se recomienda adquirir un husillo de mayor potencia para que abastezca las revoluciones necesarias.
- Utilizar las medidas de seguridad necesarias y equipo de protección personal para evitar accidentes durante el proceso de corte.
- Si se trabaja a altas velocidades de corte se recomienda el uso de un sistema de refrigeración para evitar el rápido deterioro de la herramienta por sobrecalentamiento.

## Bibliografía

- [1] «Departamento de Ingeniería eléctrica, electrónica y control,» [En línea]. Available: [www.ieec.uned.es/.../Informacion\\_de\\_referencia\\_ISE6\\_2\\_1.pdf](http://www.ieec.uned.es/.../Informacion_de_referencia_ISE6_2_1.pdf). [Último acceso: 17 Agosto 2018].
- [2] «ElCosmonauta,» [En línea]. Available: <https://elcosmonauta.es/tecnologia-mejoras-proceso-produccion/>. [Último acceso: 14 Agosto 2018].
- [3] «El Comercio,» 3 Marzo 2018. [En línea]. Available: <https://elcomercio.pe/especial/zona-ejecutiva/tendencias/10-empresas-que-destacan-materia-investigacion-noticia-1993115>. [Último acceso: 21 Agosto 2018].
- [4] R. E. Haber Guerra, J. R. Alique López y A. Alique Palomar, «Control inteligente de los procesos de mecanizado,» Madrid, 2002.
- [5] «TCI Cutting,» [En línea]. Available: <https://www.tci-cutting.com/tci-cutting-incorporara-la-vision-artificial-avanzada-en-sus-maquinas-corte-por-laser/>. [Último acceso: 25 Agosto 2018].
- [6] ABAX, «ABAX,» 5 Abril 2018. [En línea]. Available: <https://www.abax3dtech.com/single-post/2018/04/05/La-revolucion-40-se-hace-con-tecnologia-CNC>. [Último acceso: 2018 Junio 02].
- [7] M. P. Groover, Fundamentos de Manufactura Moderna, Tercera ed., Mexico: McGraw-Hill, 2007, p. 4.
- [8] E. J. Carletti, «Robots Argentina,» 2017. [En línea]. Available: [http://robots-argentina.com.ar/MotorPP\\_basico.htm](http://robots-argentina.com.ar/MotorPP_basico.htm). [Último acceso: 2018 Junio 20].
- [9] D. Bueno, «Wordpress,» [En línea]. Available: <https://1mecanizadoelarenal.files.wordpress.com/2014/11/motor-electrico.pdf>. [Último acceso: 12 Junio 2018].
- [10] A. Ramírez, «Academia,» 09 Julio 2015. [En línea]. Available: [https://www.academia.edu/13851213/Clasificacion\\_de\\_los\\_motores](https://www.academia.edu/13851213/Clasificacion_de_los_motores). [Último acceso: 15 Junio 2018].
- [11] J. M. Camacho Mejia, «Wordpress,» 11 Abril 2014. [En línea]. Available: <https://elmasolicitadopayan.wordpress.com/2014/04/11/motor-paso-a-paso/>. [Último acceso: 27 Agosto 2018].

- [12] «ClubSe,» [En línea]. Available: <http://www.clubse.com.ar/download/pdf/notasrevistas20/nota04.htm>. [Último acceso: 02 Diciembre 2018].
- [13] L. F. Obando, «dademuchconnection,» [En línea]. Available: <https://dademuchconnection.wordpress.com/2018/04/26/driver-de-motor-dc-electronica-de-potencia/>. [Último acceso: 03 Diciembre 2018].
- [14] ECG-SAVEBASE , *User Manual of 5Axis Breakout Board*.
- [15] J. Ramos, «Rootear,» 13 05 2014. [En línea]. Available: <https://rootear.com/desarrollo/tipos-placas-arduino>. [Último acceso: 19 Mayo 2018].
- [16] Arduino.cl, «Arduino.cl,» [En línea]. Available: <http://arduino.cl/que-es-arduino/>. [Último acceso: 13 Mayo 2018].
- [17] Arduino.cl, «Arduino.cl,» [En línea]. Available: <http://arduino.cl/que-es-un-shield/>. [Último acceso: 12 Mayo 2018].
- [18] BricoGeek, «BricoGeek,» [En línea]. Available: <https://tienda.bricogeek.com/shields-arduino/837-arduino-cnc-shield-v3.html>. [Último acceso: 2018 Abril 20].
- [19] I. Uriarte, «PROMETEC,» 12 Marzo 2019. [En línea]. Available: <https://www.prometec.net/blog-ramps-1-4/>. [Último acceso: 2019 Abril 19].
- [20] REPRAP, «REPRAP,» 5 Octubre 2014. [En línea]. Available: [https://reprap.org/wiki/File:RAMPS1-3\\_fin.JPG](https://reprap.org/wiki/File:RAMPS1-3_fin.JPG). [Último acceso: 2018 Agosto 18].
- [21] Staticboards, «Staticboards,» [En línea]. Available: <https://www.staticboards.es/productos/ramps14sb/>. [Último acceso: 2018 Agosto 29].
- [22] E. Rodríguez de Luis, «Xakata,» 17 Julio 2018. [En línea]. Available: <https://www.xataka.com/makers/cero-maker-todo-necesario-para-empezar-raspberry-pi>. [Último acceso: 2018 Diciembre 20].
- [23] PROTONEER, «PROTONEER,» 22 Agosto 2018. [En línea]. Available: [https://wiki.protoneer.co.nz/Raspberry\\_Pi\\_CNC](https://wiki.protoneer.co.nz/Raspberry_Pi_CNC). [Último acceso: 06 Noviembre 2018].
- [24] Ebay, «Ebay,» 24 Enero 2019. [En línea]. Available: [https://www.ebay.com/itm/Raspberry-Pi-CNC-Hat-V2-60-GRBL-v1-1-Compatable-Uses-Pololu-Stepper-Drivers-/271901344091?\\_ul=EC](https://www.ebay.com/itm/Raspberry-Pi-CNC-Hat-V2-60-GRBL-v1-1-Compatable-Uses-Pololu-Stepper-Drivers-/271901344091?_ul=EC). [Último acceso: 19 Febrero 2019].

- [25] E. T. d. Dédalo, «El Taller de Dédalo,» [En línea]. Available: <http://tallerdedalo.es/web/MundoCNC/ControladorasCNC>. [Último acceso: 14 Noviembre 2018].
- [26] EcuRed, «EcuRed,» [En línea]. Available: [https://www.ecured.cu/Software\\_propietario](https://www.ecured.cu/Software_propietario). [Último acceso: 03 Marzo 2018].
- [27] «NEWFANGLED SOLUTIONS,» [En línea]. Available: <http://www.machsupport.com/software/mach3/>. [Último acceso: 2018 Septiembre 15].
- [28] L. Arteaga, «GNU,» 21 Marzo 2019. [En línea]. Available: <https://www.gnu.org/philosophy/free-sw.es.html>. [Último acceso: 2019 Abril 14].
- [29] «DAK Engineering,» 2005. [En línea]. Available: <http://www.dakeng.com/man/turbocnc.html>. [Último acceso: 2018 Agosto 13].
- [30] «LinuxCNC,» 3 Diciembre 2016. [En línea]. Available: <http://linuxcnc.org/docs/2.7/html/getting-started/about-linuxcnc-es.html>. [Último acceso: 2018 Agosto 02].
- [31] «LinuxCNC,» [En línea]. Available: <http://linuxcnc.org/docs/2.7/html/gui/axis.html>. [Último acceso: 2018 Septiembre 18].
- [32] «CNC Control,» 2015. [En línea]. Available: <http://www.cnccontrolapp.com/index.html>. [Último acceso: 2018 Agosto 01].
- [33] Printrun, «Printrun,» [En línea]. Available: <https://www.pronterface.com/>. [Último acceso: 01 Septiembre 2018].
- [34] «Plastic Scribbler,» 2016. [En línea]. Available: <https://www.plasticscribbler.com/tutorial/getting-started/item/21-getting-started-with-pronterface#.W66pqfIRfIU>. [Último acceso: 2018 Agosto 03].
- [35] V. Ventura, «Polaridad,» 03 Diciembre 2014. [En línea]. Available: <https://polaridad.es/que-es-g-code/>. [Último acceso: 23 Noviembre 2018].
- [36] V. Ventura, «Polaridad,» 3 Diciembre 2014. [En línea]. Available: <https://polaridad.es/que-es-g-code/>. [Último acceso: 2018 Agosto 04].
- [37] L. R, «R-Luis CNC,» [En línea]. Available: <http://r-luis.xbot.es/cnc/codes02.html>. [Último acceso: 14 Noviembre 2018].
- [38] C. Riba, «Diseño Concurrente,» de *Diseño Concurrente*, 2002, p. 176.

- [39] D. Corzo y M. Cuatín, *Diseño de una máquina semiautomática para la fabricación de adoquines y bloques*, Quito, Pichincha, 2011.
- [40] «jaibalaji,» [En línea]. Available: <http://www.jaibalaji.firm.in/pushbutton.html>. [Último acceso: 15 Noviembre 2018].
- [41] Jorge, «Paletos de la Electrónica,» 27 Octubre 2014. [En línea]. Available: <https://paletosdelaelectronica.wordpress.com/2014/10/27/arduino-comunicacion-serial/>. [Último acceso: 15 Noviembre 2018].
- [42] «TutorialsPoints,» [En línea]. Available: [https://www.tutorialspoint.com/arduino/arduino\\_communication.htm](https://www.tutorialspoint.com/arduino/arduino_communication.htm). [Último acceso: 15 Noviembre 2018].
- [43] «Novelec,» 21 Septiembre 2018. [En línea]. Available: <https://blog.gruponovelec.com/electricidad/finales-de-carrera-que-son-y-caracteristicas-principales/>. [Último acceso: 15 Noviembre 2018].
- [44] «Cetronic,» [En línea]. Available: <https://www.cetronic.es/sqlcommerce/disenos/plantilla1/seccion/producto/DetalleProducto.jsp?idIdioma=&cPath=1160&codProducto=354253011&idTienda=93>. [Último acceso: 15 Noviembre 2018].
- [45] RECURSOSTIC, «RECURSOSTIC,» [En línea]. Available: [http://recursostic.educacion.es/secundaria/edad/4esotecnologia/quincena11/4quincena11\\_1\\_contenidos\\_3f.htm](http://recursostic.educacion.es/secundaria/edad/4esotecnologia/quincena11/4quincena11_1_contenidos_3f.htm). [Último acceso: 2018 Agosto 17].
- [46] Keyence, «Keyence,» [En línea]. Available: <http://www.keyence.com.mx/ss/products/sensor/sensorbasics/proximity/info/>. [Último acceso: 2018 Agosto 17].
- [47] «Computación Aplicada al Desarrollo S.A.,» [En línea]. Available: [http://www.cad.com.mx/que\\_es\\_una\\_computadora.htm](http://www.cad.com.mx/que_es_una_computadora.htm). [Último acceso: 2018 Agosto 20].
- [48] Sherlin, «Sherlin,» [En línea]. Available: <http://sherlin.xbot.es/microcontroladores/introduccion-a-los-microcontroladores/que-es-un-microcontrolador>. [Último acceso: 2018 Noviembre 18].

- [49] «Semanticwebbuilder,» [En línea]. Available: [http://www.semanticwebbuilder.org.mx/es\\_mx/swb/Sistemas\\_Embebidos\\_Innovando\\_hacia\\_los\\_Sistemas\\_Inteligentes\\_](http://www.semanticwebbuilder.org.mx/es_mx/swb/Sistemas_Embebidos_Innovando_hacia_los_Sistemas_Inteligentes_). [Último acceso: 16 Noviembre 2018].
- [50] «WINCNC,» [En línea]. Available: [http://www.wincnc.net/mapp\\_knife.php](http://www.wincnc.net/mapp_knife.php). [Último acceso: 17 Noviembre 2018].
- [51] «HNC Electric,» [En línea]. Available: [http://hncelectric.com/sp\\_product\\_list.aspx?category\\_id=103](http://hncelectric.com/sp_product_list.aspx?category_id=103). [Último acceso: 17 Noviembre 2018].
- [52] «HNC Electric,» [En línea]. Available: [http://hncelectric.com/sp\\_product\\_show.aspx?id=452](http://hncelectric.com/sp_product_show.aspx?id=452). [Último acceso: 17 Noviembre 2018].
- [53] «Nova Labs,» 27 Julio 2015. [En línea]. Available: [https://www.nova-labs.org/wiki/3d\\_printer\\_build\\_group\\_7\\_instructions/uploading\\_firmware](https://www.nova-labs.org/wiki/3d_printer_build_group_7_instructions/uploading_firmware). [Último acceso: 15 Diciembre 2018].
- [54] «XYZCNC,» [En línea]. Available: <http://www.xyzcnc.com.mx/electr%C3%B3nica-cnc/controladoras-cnc/>. [Último acceso: 15 Diciembre 2018 ].
- [55] «Aliexpress,» [En línea]. Available: <https://es.aliexpress.com/item/32704620352.html?spm=a219c.search0304.4.15.rvs8PS>. [Último acceso: 17 Diciembre 2019].
- [56] Tecnología , «Área Tecnología,» [En línea]. Available: <https://www.areatecnologia.com/diagramas-de-flujo.htm>. [Último acceso: 02 Septiembre 2018].
- [57] «Mercado Libre,» [En línea]. Available: [https://articulo.mercadolibre.com.co/MCO-454341571-varilla-lisa-acero-plata-eje-20mm-x-1-m-cnc-\\_JM?quantity=1](https://articulo.mercadolibre.com.co/MCO-454341571-varilla-lisa-acero-plata-eje-20mm-x-1-m-cnc-_JM?quantity=1). [Último acceso: 17 Diciembre 2018].
- [58] «Amazon,» [En línea]. Available: <https://www.amazon.es/SODIAL-SHF20-lineal-Rodamiento-soporte/dp/B00NLYGA7O>. [Último acceso: 17 Diciembre 2018].
- [59] Disai, «Disai,» [En línea]. Available: [https://www.disai.net/wp-content/uploads/catalogos\\_pdf/Acoplamientos.pdf](https://www.disai.net/wp-content/uploads/catalogos_pdf/Acoplamientos.pdf). [Último acceso: 02 Septiembre 2018].

- [60] «Amazon,» [En línea]. Available: <https://www.amazon.es/Aluminio-Eje-Flexible-acoplamiento-acoplador/dp/B072QVXW1J>. [Último acceso: 17 Diciembre 2018].
- [61] «Rollmecatronic,» [En línea]. Available: <http://rollmecatronic.com/producto/soporte-rodamiento-8mm-kfl08/>. [Último acceso: 18 Diciembre 2018].
- [62] Jwwinco, «Jwwinco,» [En línea]. Available: <https://www.jwwinco.com/es-mx/technical/engineering-tips/pros-and-cons-of-aluminum>. [Último acceso: 04 Septiembre 2018].
- [63] «Staticboards,» [En línea]. Available: <https://www.staticboards.es/blog/motores-paso-paso/>. [Último acceso: 18 Diciembre 2018].
- [64] «PracticarArduino,» [En línea]. Available: [http://www.practicararduino.com/manualrapido/alimentacion\\_elctrica\\_de\\_arduino.html](http://www.practicararduino.com/manualrapido/alimentacion_elctrica_de_arduino.html). [Último acceso: 18 Diciembre 2018].
- [65] «YUMO,» [En línea]. Available: <http://www.yumoelectric.com/ISMM1468-002-100B-5L-Manual-Encoder-for-CNC-Machine-Tool-Hand-Wheel-Encoder-pd6527652.html>. [Último acceso: 17 Diciembre 2018].
- [66] «Amazon,» [En línea]. Available: <https://www.amazon.com/Spindle-Cooled-Milling-Converter-Engraving/dp/B01LNBOCDA>. [Último acceso: 18 Diciembre 2018].
- [67] «Sodimac,» [En línea]. Available: <https://www.sodimac.cl/sodimac-cl/product/375756/interruptor-automatico-20-a>. [Último acceso: 19 Diciembre 2018].
- [68] «Amazon,» [En línea]. Available: <https://www.amazon.es/Digital-Anal%C3%B3gico-M%C3%B3dulo-Convertidor-Tranformer/dp/B01L1Y0TQ8>. [Último acceso: 17 Diciembre 2018].
- [69] «Marlin,» [En línea]. Available: <http://marlinfw.org/tools/u8glib/converter.html>. [Último acceso: 20 Diciembre 2018].
- [70] «Famic,» [En línea]. Available: <https://www.famiq.com.ar/producto/67565-varilla-roscada-unc-astm-193-304-1-x-1000-mm>. [Último acceso: 19 Diciembre 2018].
- [71] «Stepperbest,» 08 Agosto 2018. [En línea]. Available: <https://www.stepperbest.com/download/23HS45-4204S.pdf>. [Último acceso: 17 Diciembre 2018].
- [72] inkscape, «inkscape,» [En línea]. Available: <https://inkscape.org/es/acerca-de/>. [Último acceso: 18 09 2018].

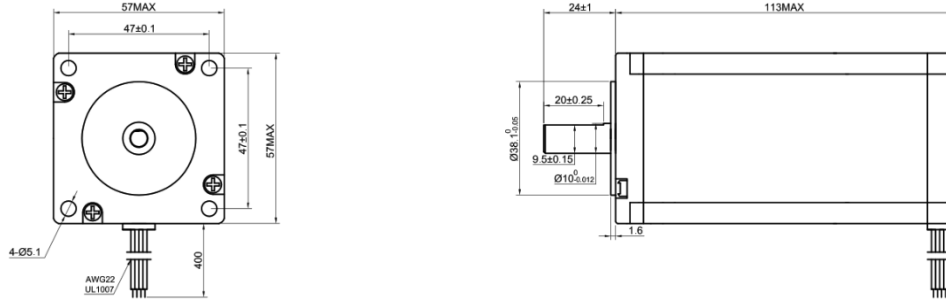
- [73] Gcodetools, «CNC-Club,» [En línea]. Available: <http://www.cnc-club.ru/gcodetools>. [Último acceso: 16 08 2018].
- [74] L. Xander, «NC Viever,» [En línea]. Available: <https://ncviewer.com/>. [Último acceso: 06 Enero 2019].
- [75] «MercadoLibre,» [En línea]. Available: [https://articulo.mercadolibre.com.co/MCO-472628899-rodamiento-lineal-20mm-cerrado-lm20uu-scs20uu-\\_JM](https://articulo.mercadolibre.com.co/MCO-472628899-rodamiento-lineal-20mm-cerrado-lm20uu-scs20uu-_JM). [Último acceso: 18 Diciembre 2018].



# **ANEXOS**

Anexo 1. Hojas de datos

Nema 23 23HS45-4204S

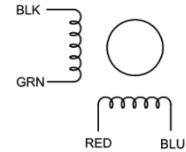


SPECIFICATION	CONNECTION	BIPOLAR
VOTAGE(VOC)		3.78
AMPS/PHASE		4.20
RESISTANCE/PHASE(Ohms)@25°C		0.90±10%
INDUCTANCE/PHASE(mH)@1KHz		3.80±20%
HOLDING TORQUE(Nm)[lb-in]		3.00[26.55]
STEP ANGLE(°)		1.80
STEP ACCURACY(NON-ACCUM)		±5.00%
ROTOR INERTIA(g-cm²)		800.00
WEIGHT(Kg)[lb]		1.60[3.53]
TEMPERATURE RISE:MAX.80°C ( MOTOR STANDSTILL;FOR 2PHASE ENERGIZED )		
AMBIENT TEMPERATURE -10°C-50°C[14°F-122°F]		
INSULATION RESISTANCE 100 Mohm ( UNDER NORMAL TEMPERATURE AND HUMIDITY )		
INSULATION CLASS B 130°C[266°F]		
DIELECTRIC STRENGTH 500VAC FOR 1MIN.(BETWEEN THE MOTOR COILS AND THE MOTOR CASE )		
AMBIENT HUMIDITY MAX.85%(NO CONDENSATION)		

TYPE OF CONNECTION (EXTERN)		MOTOR	
PIN NO	BIPOLAR	LEADS	WINDING
1	A —	BLK	A
2	A  —	GRN	A
3	B —	RED	B
4	B  —	BLU	B

FULL STEP 2 PHASE-Ex. .  
WHEN FACING MOUNTING END (X)

STEP	A	B	A	B	
1	+	+	-	-	CCW
2	-	+	+	-	↓
3	-	-	+	+	↑
4	+	-	-	+	CW



APVD		
CHKD		
1:1.5	DRN	
SCALE	SIGNATURE	DATE

STEPPER MOTOR

23HS45-4204S

## DRIVER TB6600



**DRIVER**  
Model: WD-TB6600

### FEATURES

- 1 - Average output current control;
- 2 - Bipolar sinusoidal micro-step stepping driver;
- 3 - Wide supply range from 12 to 48VDC;
- 4 - Digital Inputs optically isolated;
- 5 - Under-voltage protection;
- 6 - Over-current detection circuit;
- 7 - Five selectable micro-stepping possibilities (1/1, 1/2, 1/4, 1/8 and 1/16);
- 8 - Eight selectable output phase current settings;
- 9 - High starting speed;
- 10 - High-speed torque.

### TECHNICAL DATA

Supply voltage 12 ~ 48VDC

Input current of 1 to 5A depending on the selected  
Output current settings.

Stepper Motor output current of 0.2A ~ 5A

Operating Temperature -10 to 45 °C;

Storage temperature -40 °C to 70 °C

Weight 230 grams

### CONTROL SIGNAL INTERFACE

#### 1- Control signals description:

**PUL +** : step pulse signal positive input;

**PUL -** : step pulse signal negative input;

**DIR +** : stepping direction signal positive input;

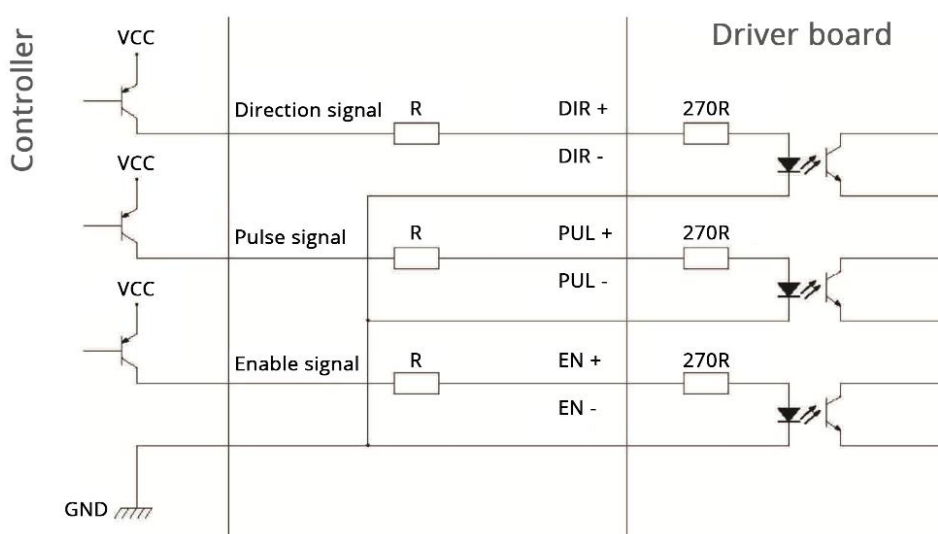
**DIR -** : stepping direction signal negative input;

**EN +** : offline enable signal positive input;

**EN -** : offline enable signal negative input;

## 2- Control Signal Connections

The control signals can be low-level active or high-level active. For low-level active, the positive inputs must be connected to VCC (+5V) and for high-level active the negative inputs must be connected to reference GND (0V). For open-collector and PNP output interface the connections should be as follows:



### Note:

For VCC = 5V, R is not necessary;

For VCC = 12V, R should be 1K - 1/4W resistor;

For VCC = 24V, R should be 2K - 1/4W resistor;

R resistor must be connected to the controller output terminals.

## FUNCTION SELECTION (DIP-switch on the drive panel)

Microstep resolutions and output current are programmable, the steps can be set from full-step (1/1) to (1/16) 3200 steps/rev and the latter can be set from 0.2A to 5A.

### 1- Microstep Resolution Selection

The microstep resolution is set by DIP-switches SW4, SW5 and SW6 as shown in the following table.

SUB	4	5	6
NC	ON	ON	ON
1	OFF	ON	ON
1/2	ON	OFF	ON
1/2	OFF	OFF	ON
1/4	ON	ON	OFF
1/8	OFF	ON	OFF
1/16	ON	OFF	OFF
NG	OFF	OFF	OFF

### 2- Output phase current settings

The dynamic current limitation for the motor coils is set by the DIP-switches SW1, SW2 and SW3 as shown in the following table.

Cur	1	2	3
0.2A	ON	ON	ON
0.6A	OFF	ON	ON
1.2A	ON	OFF	ON
1.8A	OFF	OFF	ON
2.5A	ON	ON	OFF
3.3A	OFF	ON	OFF
4.2A	ON	OFF	OFF
5A	OFF	OFF	OFF

#### Note:

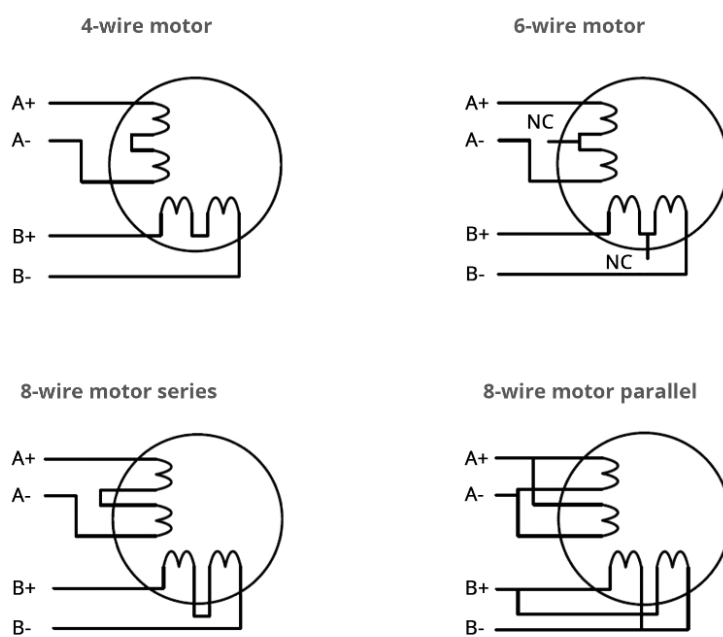
- Select a setting closest to your motor's required current.
- Due to motor inductance, the actual current in the coil may be smaller than the dynamic current setting, particularly under high speed condition.

### 3- POWER INTERFACE

**DC +, DC-:** Connection for power supply voltage (12 to 48VDC);

**A+, A-, B+, B-:** Connection for the two-phase hybrid stepping motor;

The stepper-motor driver WD-TB6600 supports any 2-phase or 4-phase hybrid stepping motors of 4, 6 and 8 wires. The following motor connections are possible:



## DRIVER DM542T

### User's Manual

For

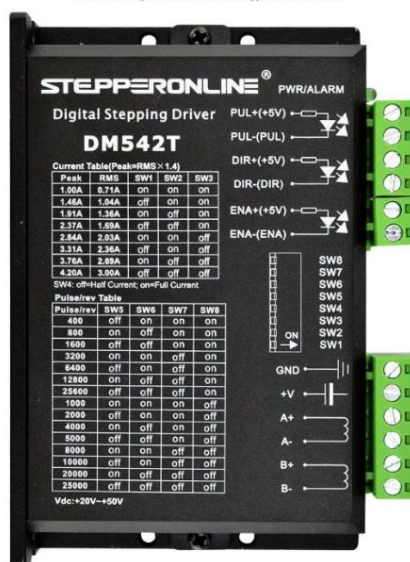
## DM542T

### Full Digital Stepper Drive

Designed by StepperOnline®

Manufactured by Leadshine®

©2017 All Rights Reserved Attention: Please read this manual carefully before using the drive!



# STEPPERONLINE®

#7 Zhongke Road, Jiangning, Nanjing, China

Tel: 0086-2587156578

Web site: [www.omc-stepperonline.com](http://www.omc-stepperonline.com)

E-Mail: [sales@stepperonline.com](mailto:sales@stepperonline.com)

## Table of Contents

1. Introductions, Features and Applications.....	1
Introductions.....	1
Features.....	1
Applications.....	1
2. Specifications.....	1
Electrical Specifications ( $T_j = 25^\circ\text{C} / 77^\circ\text{F}$ ).....	1
Operating Environment and other Specifications.....	2
Mechanical Specifications (unit: mm [1inch=24.5mm]).....	2
Elimination of Heat.....	2
3. Pin Assignment and Description.....	3
Connector P1 Configurations.....	3
Connector P2 Configurations.....	3
4. Control Signal Connector (P1) Interface.....	3
Connections of 4-lead Motor.....	4
Connections of 6-lead Motor.....	4
Half Coil Configurations.....	4
Full Coil Configurations.....	5
Connections of 8-lead Motor.....	5
Series Connections.....	5
Parallel Connections.....	5
5. Power Supply Selection.....	6
Regulated or Unregulated Power Supply.....	6
Multiple Drives.....	6
Selecting Supply Voltage.....	6
6. Selecting Microstep Resolution and drive Output Current.....	6
Microstep Resolution Selection.....	6
Current Settings.....	7
Dynamic Current Setting.....	7
Standstill Current Setting.....	8
7. Wiring Notes.....	8
8. Typical Connection.....	8
9. Sequence Chart of Control Signals.....	9
10. Protection Functions.....	9
11. Frequently Asked Questions.....	10
Problem Symptoms and Possible Causes.....	10



## 1. Introductions, Features and Applications

### Introductions

The DM542T is a fully digital stepper drive developed with advanced DSP control algorithm based on the latest motion control technology. It has achieved a unique level of system smoothness, providing optimal torque and nulls mid-range instability. Its motor auto-identification and parameter auto-configuration feature offers quick setup to optimal modes with different motors. Compared with traditional analog drives, DM542T can drive a stepper motor at much lower noise, lower heating, and smoother movement. Its unique features make DM542T an ideal choice for high requirement applications.

### Features

- Anti-Resonance provides optimal torque and nulls mid-range instability
- Motor auto-identification and parameter auto-configuration when power on, offer optimal responses with different motors
- Multi-Stepping allows a low resolution step input to produce a higher microstep output, thus offers smoother motor movement
- 15 selectable microstep resolutions including 400, 800, 1600, 3200, 6400, 12800, 25600, 1000, 2000, 4000, 5000, 8000, 10000, 20000, 25000
- Soft-start with no “jump” when powered on
- Input voltage 20-50VDC
- 8 selectable peak current including 1.00A, 1.46A, 1.91A, 2.37A, 2.84A, 3.31A, 3.76A, 4.20A
- Pulse input frequency up to 200 KHz, TTL compatible and optically isolated input
- Automatic idle-current reduction
- Suitable for 2-phase and 4-phase motors
- Support PUL/DIR mode
- Over-voltage and over-current protections

### Applications

Suitable for a wide range of stepper motors, size from NEMA17 to 24. It can be used in various kinds of machines, such as X-Y tables, engraving machines, labeling machines, laser cutters, pick-place devices, and so on. Particularly adapt to the applications with low noise, low heating, high speed and high precision.

## 2. Specifications

### Electrical Specifications ( $T_j = 25^{\circ}\text{C}/77^{\circ}\text{F}$ )

Parameters	DM542T			Unit
	Min	Typical	Max	
Output Peak Current	1.0	-	4.2 (3.0 RMS)	A
Input Voltage Logic	+20	+36	+50	VDC
Signal Current Pulse	7	10	16	mA
input frequency Pulse	0	-	200	kHz
Width	2.5	-	-	uS
Isolation resistance	500	-	-	MΩ

**Operating Environment and other Specifications**

Cooling	Natural Cooling or Forced cooling	
	Environment	Avoid dust, oil fog and corrosive gases
Operating Environment	Ambient Temperature	0°C – 65°C
	Humidity	40%RH – 90%RH
	Operating Temperature	-10°C – 45°C
Storage Temperature	-20°C – 65°C	
Weight	Approx. 210g (7.4oz)	

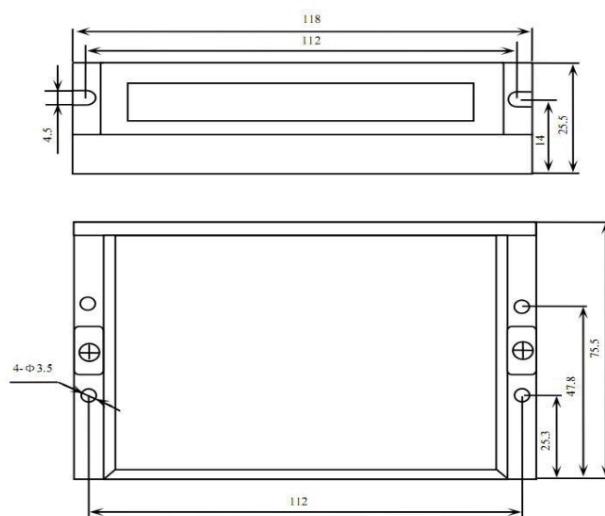
**Mechanical Specifications (unit: mm [1inch=25.4mm])**


Figure 1: Mechanical specifications

**\*Recommend use side mounting for better heat dissipation**

**Elimination of Heat**

- Drive's reliable working temperature should be <math><45^{\circ}\text{C}</math> (113 °F), and motor working temperature should be <math><80^{\circ}\text{C}</math> (176 °F);
- It is recommended to use automatic idle-current mode, namely current automatically reduce to 50% when motor stops, so as to reduce drive heating and motor heating;
- It is recommended to mount the drive vertically to maximize heat sink area. Use forced cooling method to cool the system if necessary.

### 3. Pin Assignment and Description

The DM542T has two connectors P1&P2, P1 is for control signals connections, and P2 is for power and motor connections. The following tables are brief descriptions of the two connectors. More detailed descriptions of the pins and related issues are presented in section 4, 5, 9.

#### Connector P1 Configurations

Pin Function	Details
PUL+	<u>Pulse signal:</u> In single pulse (pulse/direction) mode, this input represents pulse signal, each rising edge active; 4-5V when PUL-HIGH, 0-0.5V when PUL-LOW. The DM542T drive has no double pulse mode (pulse/pulse). For reliable response, pulse width should be longer than 2.5 $\mu$ s. Series connect resistors for current-limiting when +12V or +24V used (1K for +12V, 2k for +24V). The same as DIR and ENA signals.
PUL-	
DIR+	<u>DIR signal:</u> In single-pulse mode, this signal has low/high voltage levels, representing two directions of motor rotation; The DM542T drive has no double pulse mode (pulse/pulse). 4-5V when DIR-HIGH, 0-0.5V when DIR-LOW. Please note that rotation direction is also related to motor-drive wiring match. Exchanging the connection of two wires for a coil to the drive will reverse motor direction.
DIR-	
ENA+	<u>Enablesignal:</u> This signal is used for enabling/disabling the drive. High level (NPN control signal, PNP and Differential control signals are on the contrary, namely Low level for enabling.) for enabling the drive and low level for disabling the drive. Usually left
ENA-	

#### Connector P2 Configurations

Pin Function	Details
GND	Power Ground.
+V	Power supply, 20~50 VDC, Including voltage fluctuation and EMF voltage.
A+, A-	Motor Phase A
B+, B-	Motor Phase B

### 4. Control Signal Connector (P1) Interface

The DM542T can accept differential and single-ended inputs (including open-collector and PNP output). The DM542T has 3 optically isolated logic inputs which are located on connector P1 to accept line drive control signals. These inputs are isolated to minimize or eliminate electrical noises coupled with the drive control signals. Recommend using line drive control signals to increase noise immunity for the drive in interference environments. In the following figures, connections to open-collector and PNP signals are illustrated.

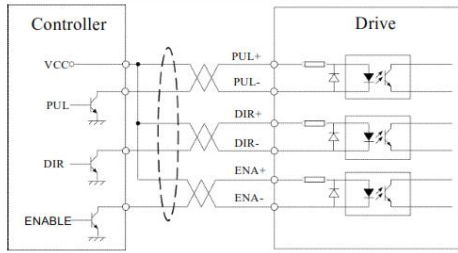


Figure 2: Connections to open-collector signal (common-anode)

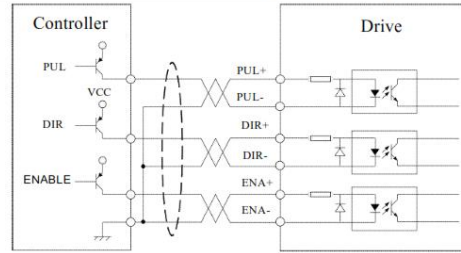


Figure 3: Connections to PNP signal (common-cathode)

**Connections of 4-lead Motor**

The 4 lead motors are the least flexible and easy to connect. And the Speed – torque of motor depends on winding inductance. The output current from drive that is multiply the specified phase current by 1.4 to determine the peak output current.

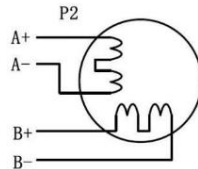


Figure 4: 4-lead Motor Connections

**Connections of 6-lead Motor**

Like 8 lead stepping motors, 6 lead motors have two configurations available for high speed or high torque operations. The higher speed configuration, or half coil, is described, because it uses one half of the motor’s inductor windings. The higher torque configuration, or full coil, uses the full coil windings.

**Half Coil Configurations**

As previously stated, the half coil configuration uses 50% of the motor phase windings. This gives lower inductance, hence, lower torque output. Like the parallel connection of 8 lead motor, the torque output will be more stable at higher speeds. This configuration is also referred to as half chopper. In setting the drive output current multiply the specified per phase (or unipolar) current rating by 1.4 to determine the peak output current.

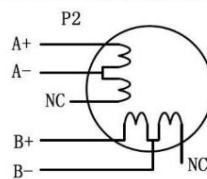


Figure 5: 6-lead motor half coil (higher speed) connections

### Full Coil Configurations

The full coil configuration on a six lead motor should be used in applications where higher torque at lower speed is desired. This configuration is also referred to as full copper. In full coil mode, the motors should be run at only 70% of their rated current to prevent overheating.

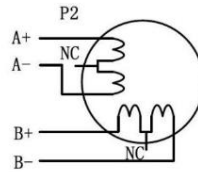


Figure 6: 6-lead motor full coil (higher torque) connections

### Connections of 8-lead Motor

8 lead motors offer a high degree of flexibility to the system designer in that they may be connected in series or parallel, thus satisfying a wide range of applications.

#### Series Connections

A series motor configuration would typically be used in applications where a higher torque at lower speed is required. Because this configuration has the most inductance, the performance will start to degrade at higher speed. In series mode, the motors should also be run at only 70% of their rated current to prevent overheating.

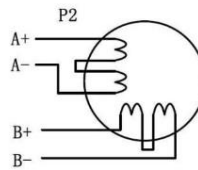


Figure 7: 8-lead motor series connections

#### Parallel Connections

An 8 lead motor in a parallel configuration offers a more stable, but lower torque at lower speeds. But because of the lower inductance, there will be higher torque at higher speeds. Multiply the per phase (or unipolar) current rating by 1.96, or the bipolar current rating by 1.4, to determine the peak output current.

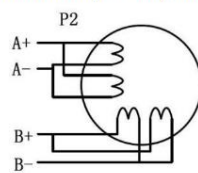


Figure 8: 8-lead motor parallel connections

## 5. Power Supply Selection

The DM542T can match medium and small size stepping motors (frame size from NEMA17 to 34). To achieve good driving performances, it is important to select supply voltage and output current properly. Generally speaking, supply voltage determines the high speed performance of the motor, while output current determines the output torque of the driven motor (particularly at lower speed). Higher supply voltage will allow higher motor speed to be achieved, at the price of more noise and heating. If the motion speed requirement is low, it's better to use lower supply voltage to decrease noise, heating and improve reliability.

### Regulated or Unregulated Power Supply

Both regulated and unregulated power supplies can be used to supply the drive. However, unregulated power supplies are preferred due to their ability to withstand current surge. If regulated power supplies (such as most switching supplies.) are indeed used, it is important to have large current output rating to avoid problems like current clamp, for example using 4A supply for 3A motor-drive operation. On the other hand, if unregulated supply is used, one may use a power supply of lower current rating than that of motor (typically 50% ~ 70% of motor current). The reason is that the drive draws current from the power supply capacitor of the unregulated supply only during the ON duration of the PWM cycle, but not during the OFF duration. Therefore, the average current withdrawn from power supply is considerably less than motor current. For example, two 3A motors can be well supplied by one power supply of 4A rating.

### Multiple Drives

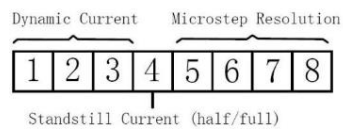
It is recommended to have multiple drives to share one power supply to reduce cost, if the supply has enough capacity. To avoid cross interference, DO NOT daisy-chain the power supply input pins of the drives. (Instead, please connect them to power supply separately.)

### Selecting Supply Voltage

The power MOSFETS inside the DM542T can actually operate within +20 ~ +50VDC, including power input fluctuation and back EMF voltage generated by motor coils during motor shaft deceleration. Higher supply voltage can increase motor torque at higher speeds, thus helpful for avoiding losing steps. However, higher voltage may cause bigger motor vibration at lower speed, and it may also cause over-voltage protection or even drive damage. Therefore, it is suggested to choose only sufficiently high supply voltage for intended applications, and it is suggested to use power supplies with theoretical output voltage of +20 ~ +45VDC, leaving room for power fluctuation and back-EMF.

## 6. Selecting Microstep Resolution and drive Output Current

This drive uses an 8-bit DIP switch to set microstep resolution, and motor operating current, as shown below:



### Microstep Resolution Selection

Microstep resolution is set by SW5, 6, 7, 8 of the DIP switches as shown in the following table:

Microstep	Steps/rev.(for 1.8°motor)	SW5	SW6	SW7	SW8
2	400	OFF	ON	ON	ON
4	800	ON	OFF	ON	ON
8	1600	OFF	OFF	ON	ON
16	3200	ON	ON	OFF	ON
32	6400	OFF	ON	OFF	ON
64	12800	ON	OFF	OFF	ON
128	25600	OFF	OFF	OFF	ON
5	1000	ON	ON	ON	OFF
10	2000	OFF	ON	ON	OFF
20	4000	ON	OFF	ON	OFF
25	5000	OFF	OFF	ON	OFF
40	8000	ON	ON	OFF	OFF
50	10000	OFF	ON	OFF	OFF
100	20000	ON	OFF	OFF	OFF
125	25000	OFF	OFF	OFF	OFF

#### Current Settings

For a given motor, higher drive current will make the motor to output more torque, but at the same time causes more heating in the motor and drive. Therefore, output current is generally set to be such that the motor will not overheat for long time operation. Since parallel and serial connections of motor coils will significantly change resulting inductance and resistance, it is therefore important to set drive output current depending on motor phase current, motor leads and connection methods. Phase current rating supplied by motor manufacturer is important in selecting drive current, however the selection also depends on leads and connections.

The first three bits (SW1, 2, 3) of the DIP switch are used to set the dynamic current. Select a setting closest to your motor's required current.

#### Dynamic Current Setting

Peak Current	RMS Current	SW1	SW2	SW3
1.00A	0.71A	ON	ON	ON
1.46A	1.04A	OFF	ON	ON
1.91A	1.36A	ON	OFF	ON
2.37A	1.69A	OFF	OFF	ON
2.84A	2.03A	ON	ON	OFF
3.31A	2.36A	OFF	ON	OFF
3.76A	2.69A	ON	OFF	OFF
4.20A	3.00A	OFF	OFF	OFF

**Notes:** Due to motor inductance, the actual current in the coil may be smaller than the dynamic current setting, particularly under high speed condition.

### Standstill Current Setting

SW4 is used for this purpose. OFF meaning that the standstill current is set to be half of the selected dynamic current, and ON meaning that standstill current is set to be the same as the selected dynamic current.

The current automatically reduced to 50% of the selected dynamic current 0.4 second after the last pulse.

### Motor auto-identification and parameter auto-configuration

The drive will operate the function of motor auto-identification and parameter auto-configuration when power on, and calculate the optimal parameter using for current control after this processing, then the stepper motor can output optimal torque.

### 7. Wiring Notes

- In order to improve anti-interference performance of the drive, it is recommended to use twisted pair shield cable.
- To prevent noise incurred in PUL/DIR signal, pulse/direction signal wires and motor wires should not be tied up together. It is better to separate them by at least 10 cm, otherwise the disturbing signals generated by motor will easily disturb pulse direction signals, causing motor position error, system instability and other failures.
- If a power supply serves several drives, separately connecting the drives is recommended instead of daisy-chaining.
- It is prohibited to pull and plug connector P2 while the drive is powered ON, because there is high current flowing through motor coils (even when motor is at standstill). Pulling or plugging connector P2 with power on will cause extremely high back-EMF voltage surge, which may damage the drive.

### 8. Typical Connection

A complete stepping system should include stepping motor, stepping drive, power supply and controller (pulse generator). A typical connection is shown as figure 9.

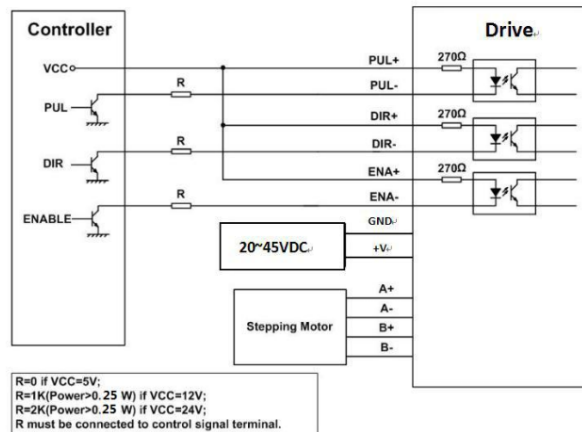


Figure 9: Typical connection



### 9. Sequence Chart of Control Signals

In order to avoid some fault operations and deviations, PUL, DIR and ENA should abide by some rules, shown as following diagram:

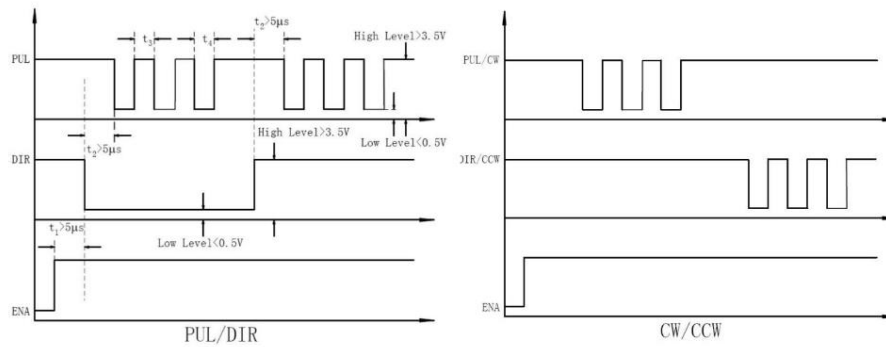


Figure 10: Sequence chart of control signals

#### Remark:

- t1: ENA must be ahead of DIR by at least  $5\mu\text{s}$ . Usually, ENA+ and ENA- are NC (not connected). See “Connector P1 Configurations” for more information.
- t2: DIR must be ahead of PUL effective edge by  $5\mu\text{s}$  to ensure correct direction;
- t3: Pulse width not less than  $2.5\mu\text{s}$ ;
- t4: Low level width not less than  $2.5\mu\text{s}$ .

### 10. Protection Functions

To improve reliability, the drive incorporates some built-in protections features.

Priority	Time(s) of Blink	Sequence wave of red LED	Description
1st	1		Over-current protection activated when peak current exceeds the limit.
2nd	2		Over-voltage protection activated when drive working voltage is greater than 60VDC

When above protections are active, the motor shaft will be free or the red LED blinks. Reset the drive by repowering it to make it function properly after removing above problems.

## 11. Frequently Asked Questions

In the event that your drive doesn't operate properly, the first step is to identify whether the problem is electrical or mechanical in nature. The next step is to isolate the system component that is causing the problem. As part of this process you may have to disconnect the individual components that make up your system and verify that they operate independently. It is important to document each step in the troubleshooting process. You may need this documentation to refer back to at a later date, and these details will greatly assist our Technical Support staff in determining the problem should you need assistance.

Many of the problems that affect motion control systems can be traced to electrical noise, controller software errors, or mistake in wiring.

### Problem Symptoms and Possible Causes

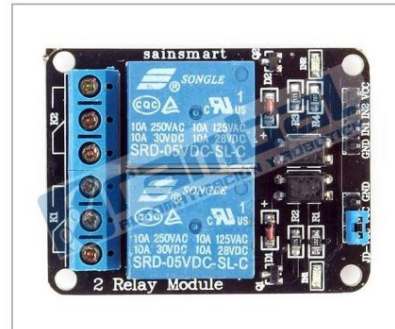
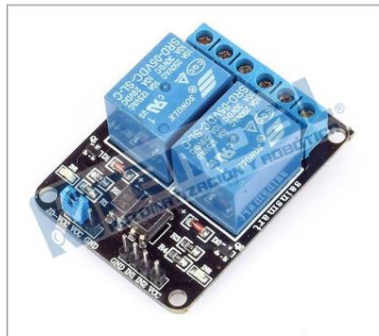
Symptoms	Possible Problems
<b>Motor is not rotating</b>	No power
	Microstep resolution setting is wrong
	DIP switch current setting is wrong
<b>Motor rotates in the wrong direction</b>	Fault condition exists
	The drive is disabled
	Motor phases may be connected in reverse
<b>The drive in fault</b>	DIP switch current setting is wrong
	Something wrong with motor coil
	Control signal is too weak
<b>Erratic motor motion</b>	Control signal is interfered
	Wrong motor connection
	Something wrong with motor coil
<b>Motor stalls during acceleration</b>	Current setting is too small, losing steps
	Current setting is too small
	Motor is undersized for the application
<b>Excessive motor and drive heating</b>	Acceleration is set too high
	Power supply voltage too low
	Inadequate heat sinking / cooling
<b>Excessive motor and drive heating</b>	Automatic current reduction function not being utilized
	Current is set too high

## MÓDULO RELÉ



www.rambal.com

### Módulo relé de 2 canales (Traducido por Rambal Ltda.)



Versión de mayor inmunidad a ruido electrónico. Full Optoacoplada Versión 4.

Esta placa puede ser controlada directamente desde cualquier controlador Home-SX, Arduino, PIC, ARM, etc., usando TTL. Solo se requiere que el microcontrolador proporcione 5 Voltios y 20 mA por cada uno de los dos pines y fácilmente estará comandando artefactos eléctricos de hasta 10 Amperes y 220 voltios en corriente alterna o artefactos de 10 Amperes y 30 Voltios en corriente continua. Hemos verificado la calidad de este componente de nuestro proveedor de Ucrania junto a nuestros clientes y no ha presentado fallas. El único inconveniente que encontramos fue la falta de imágenes de conexión, pero este inconveniente se solucionó al contactarnos con el fabricante.

#### Especificaciones:

- Voltaje de operación de 5 Vcc.
- Corriente de activación de 15 a 20 mA.
- 2 canales independientes, protegidos con opto acopladores.
- Voltaje máximo de carga en los relés de 250Vac x 10A o 30Vdc x 10A.
- Modo de funcionamiento enclave.
- Los 2 relés cuentan, cada uno con salidas NC y NA.
- Distancia de alcance de 15 a 30 metros.
- Indicación led para el estado de cada relé.
- Puede ser controlado directamente por circuitos lógicos.
- Terminales de entrada de señal lógica y alimentación con headers macho.

#### Sistema extra de protección:

Con jumper en la posición JD-Vcc / Vcc: La bobina del relé es excitado con la misma fuente de poder que es alimentado el modulo.

Sin jumper: Se requiere alimentar independientemente la bobina del relé en los pines VCC Y GND, con esto se asegura una máxima protección en posibles cortocircuitos.

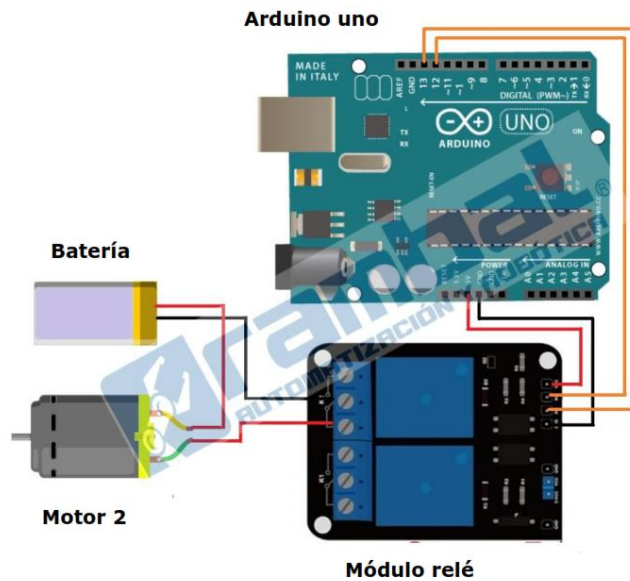
**Importante: No conectar el Jumper entre VCC y GND.**

**Características:**

Control por variados microcontroladores como: Home-SX , Arduino, PIC, ARM , B. Stamps, Raspberry Pi , usando TTL .

Aconsejamos el uso de Basic Stamp, especialmente el modelo Home-SX.

Nota: Nuestros clientes del área educación han probado los equipos Ukranianos y no han encontrado diferencias en calidad / funcionamiento.

**Conexión:**

## CONVERTOR PWM A 0-10V

Converter PWM/0-10VDC

  
www.solarcontrols.cz

### CONVERTER PWM/0-10VDC

#### USER MANUAL

Manufacturer: SOLAR controls s.r.o.

#### PURPOSE

The converter can be used wherever it is necessary to connect the output modulated by PWM signal to 0-10V input. Within the product range of the manufacturer it is used to connect controllers such as WATTrouter M or Heating Control to the 0-10V control input of an outdoor heat pump unit. Then you can directly control the output power of this unit.

#### CONNECTION

Connect the PWM converter to the PWM output of the control device according to the specifications of the relevant product (e.g. EXT1 on Heating Control). Observe PWM signal polarity. Connect an external 12 VDC power supply to terminals GND (negative) and + 12V (positive). Connect the analog signal 0-10V to the input with the same name which is available on controlled product (e.g. heat pump control unit).

#### SPECIFICATION

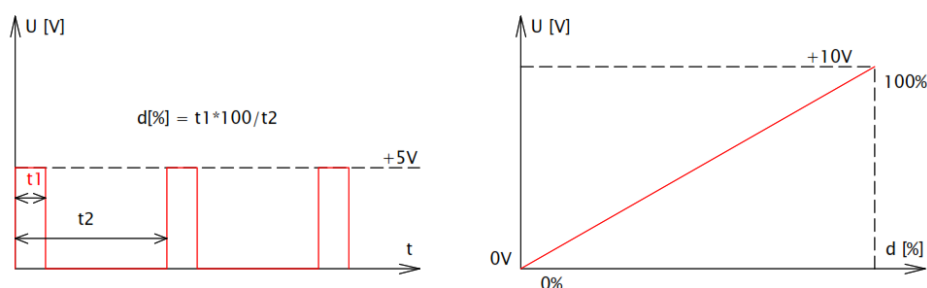


Figure 1: Output voltage  $U$  vs. PWM input signal duty cycle  $d$ .

Parameter	Value, note
Power supply	12 VDC / 10 mA
Current consumption	Max. 10 mA
Power supply – reverse polarity protection	Yes
Input – PWM base frequency	50 Hz – 500 Hz
Input – voltage	Max. 6 VDC (between PWM+ and PWM-)
Input – reverse polarity protection	Yes
Output – voltage	0-10 VDC, load max. 5 mA
Adjustable conversion linearity	Yes, after lifting the lid appropriate converter range can be adjusted with a small screwdriver
Galvanic isolation input (terminals PWM + and PWM -) /	Yes

---

**Converter PWM/0-10VDC**

<b>output (terminals GND, 0-10V, +12V)</b>	
<b>Working position</b>	Any
<b>Mounting</b>	DIN rail 35mm
<b>Pollution degree</b>	2
<b>Operational temperature range</b>	-20°C to +40°C
<b>Storage temperature range</b>	-40 °C to +80 °C
<b>IP code</b>	IP 20
<b>Dimensions (WxHxD)</b>	17,6x100x64mm (1 module)
<b>Weight</b>	50g
<b>Warranty period</b>	24 months

Note: DC supply power may be in the range 6-12 V. Then, the converter operates roughly in the range of supply voltage minus 1 V.

## Anexo 2. Códigos de programación

### Código para procesamiento de datos del control manual

```

#define ENCODER_OPTIMIZE_INTERRUPTS
#include <Encoder.h>
// Cambia los números de estos pines con los pines en los cuales esta conectado tu encoder.
// El mejor rendimiento: Ambos pines tienen capacidad de captar interrupción.
// Buen rendimiento: Solo el primer pin tiene capacidad de captar interrupción.
// Bajo rendimiento: Ninguno de los pines tiene capacidad de captar interrupción.
Encoder encoder(2, 3);
/////////////////////////////////////////////////////////////////
////////// Declaración de variables del Control //////////
int X = 4, Y = 5, Z = 6, amarillo=20,rojo=21; //Pines asignados para los ejes
int com = 10; //Pin común para switches rotativos de ejes y velocidad
int paro = 11;
int eje;
int vel1 = 7, vel10 = 8, vel100 = 9; //Pines asignados para las velocidades
String cod = "", linea = ""; //Variable para almacenar la generación de código G
String velocidad = "";
double posX = 0.0, posY = 0.0, posZ = 0.0, posAux; //Variables para almacenar la posición
                                                    de cada eje.
double newpos, sum, divisor, divi; //Variables auxiliares para el cálculo de posiciones.
double valx1 = 0.0, valx10 = 0.0, valx100 = 0.0; //Variables para almacenar los valores de
cada velocidad
bool aux = false; //Variable auxiliar

/////////////////////////////////////////////////////////////////
////////// Declaración de Botones Tablero ////
int com_b=19;
int b_blanco=17;
int b_verde=16;
//int b_emer=17;
int v_aux=0;
bool btn=true;
bool btn_aux=true;
bool btn_p=true;
bool btn_paux=false;
bool btn_r=true;

/////////////////////////////////////////////////////////////////
////////// Configuración de Pines y Comunicación //////////
void setup() {
  Serial.begin(250000); //Para el monitoreo desde un computador
  Serial3.begin(250000); //Para el envío de datos del control al interprete
  Serial.println("Test Control");
  pinMode(X, INPUT_PULLUP);

```

```

pinMode(Y, INPUT_PULLUP);
pinMode(Z, INPUT_PULLUP);
pinMode(vel1, INPUT_PULLUP);
pinMode(vel10, INPUT_PULLUP);
pinMode(vel100, INPUT_PULLUP);
pinMode(com, OUTPUT);
pinMode(amarillo, OUTPUT);
pinMode(rojo, OUTPUT);
digitalWrite(com, LOW);
pinMode(b_verde, INPUT_PULLUP);
pinMode(b_blanco, INPUT_PULLUP);
pinMode(com_b,OUTPUT);
digitalWrite(com_b,LOW);
}

/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////PROGRAMA PRINCIPAL/////////////////////////////////////////////////////////////////
void loop() {
  btnHome();
  btnPause();
  btnReanudar();
  divi = detectVel();
  detectEje();
  posAux = detectPos();
  leerEncoder(posAux, divi);
  enviarCod();
  delay(40);
}

/////////////////////////////////////////////////////////////////
/////////////////////////////////////////////////////////////////FUNCIONES/////////////////////////////////////////////////////////////////

void leerEncoder(double posActual, double divisor)
{
  sum = (valx1 * 16.0) + (valx10 * 4.0) + (valx100*2);
  newpos = encoder.read();
  if (newpos != sum)
  {
    double auxVel = (newpos - sum) / divisor;
    posActual = posActual + auxVel;
    Serial.println(sum);
    Serial.print("Posicion = ");
    Serial.println(newpos);
    escribirPos(posActual, auxVel, divisor);
    aux = true;
  }
}

void escribirPos(double newPos, double valVel, double divisor)
{
  if (divisor == 16)

```



```

    valx1 += valVel;
else if (divisor == 4)
    valx10 += valVel;
else if (divisor == 2)
    valx100 += valVel;

switch (eje)
{
    case 1:
    {
        posX = newPos;
        cod += (String)newPos + velocidad + "\n";
        break;
    }
    case 2:
    {
        posY = newPos;
        cod += (String) newPos + velocidad + "\n";
        break;
    }
    case 3:
    {
        posZ = newPos;
        cod += (String)newPos + velocidad + "\n";
        break;
    }
    default:
    {
        cod = "";
        break;
    }
}
}
double detectVel()
{
    if ((digitalRead(vel1)) == LOW)
    {
        cod += "G01 ";
        velocidad = " F200";
        divisor = 16;
    }
    else if ((digitalRead(vel10)) == LOW)
    {
        cod += "G00 ";
        velocidad = " F450";
        divisor = 4;
    }
    else if ((digitalRead(vel100)) == LOW)
    {
        cod += "G00 ";

```

```
    velocidad = " F550";
    divisor = 2;
}
return divisor;
}
void detectEje()
{
    if (digitalRead(X) == LOW)
    {
        cod += 'X';
        eje = 1;
    }
    else if (digitalRead(Y) == LOW)
    {
        cod += 'Y';
        eje = 2;
    }
    else if (digitalRead(Z) == LOW)
    {
        cod += 'Z';
        eje = 3;
    }
    else
        eje = 0;
}
double detectPos()
{
    switch (eje)
    {
        case 1:
            {
                return posX;
                break;
            }
        case 2:
            {
                return posY;
                break;
            }
        case 3:
            {
                return posZ;
                break;
            }
        default:
            break;
    }
}
void enviarCod()
{
```

```

if (aux)
{
  Serial.println(cod);
  Serial3.print(cod);
  cod = "";
  aux = false;
}
else
  cod = "";
}

////////////////////////////////////
//////////////////////////////////////FUNCIONES BOTONES TABLERO DE CONTROL////////////////////////////////////
void btnPause()
{
  if((digitalRead(b_blanco))!=LOW)
  {
    while(btn_p)
    {
      if((digitalRead(b_blanco))==LOW)
      {
        btn_p=false;
        btn_paux=true;
        Serial3.println("M25");
        Serial.println("M25");
      }
    }
  }
  btn_p=true;
}
void btnHome()
{
  if(btn & btn_paux==false)
  {
    if((digitalRead(b_verde))==LOW)
    {
      while(btn_aux)
      {
        if((digitalRead(b_verde))!=LOW)
        {
          btn=false;
          btn_aux=false;
          Serial3.println("G28 Z");
          Serial.println("G28 Z");
          delay(3000);
          Serial.println("G28 X Y");
          Serial3.println("G28 X Y");
          encoder.write(0);
          posX = 0.0;
          posY = 0.0;
        }
      }
    }
  }
}

```

```
    posZ = 0.0;
    posAux=0.0;
    sum=0;
    newpos=0;
  }
}
}
}
}
void btnReanudar()
{
  if(btn_paux)
  {
    if((digitalRead(b_verde))==LOW)
    {
      while(btn_r)
      {
        if((digitalRead(b_verde))!=LOW)
        {
          btn_r=false;
          btn_paux=false;
          Serial3.println("M24");
          Serial.println("M24");
        }
      }
    }
  }
  btn_r=true;
}
```

### Programación usada en las pruebas de desplazamiento

```

int PUL=7; //define Pulse pin
int DIR=6; //define Direction pin
int ENA=5; //define Enable Pin
int X = 9;
int cont = 0;
int cont1 = 0;
long pasos= 50000;
int del;
void setup() {
  pinMode (PUL, OUTPUT);
  pinMode (DIR, OUTPUT);
  pinMode (ENA, OUTPUT);
  digitalWrite(8, HIGH);
}
void loop() {
  if(digitalRead(X) == true & cont == 0){
    cont1 = 0;
    cont =1;
    for (long i=0; i<pasos; i++)
    {
      if (i<1000)
        del = 900;
      else if(i<2000)
        del = 800;
      else if(i<3000)
        del = 700;
      else if(i>4000 & i<44000)
        del = 600;
      else if(i>44000 & i<46000)
        del = 700;
      else if(i>46000 & i<48000)
        del = 800;
      else if(i>48000)
        del = 900;
      digitalWrite(DIR,LOW);
      digitalWrite(ENA,HIGH);
      digitalWrite(PUL,HIGH);
      delayMicroseconds(del);
      digitalWrite(PUL,LOW);
      delayMicroseconds(del);
    }
    delay(2000);
  }

  if(digitalRead(X) == false& cont1 == 0){
    cont =0;
    cont1 = 1;
  }
}

```

```
for (long i=0; i<pasos; i++)
{
  if (i<2000)
    del = 900;
  else if(i<4000)
    del = 800;
  else if(i<6000)
    del = 700;
  else if(i>8000 & i<44000)
    del = 600;
  else if(i>44000 & i<46000)
    del = 700;
  else if(i>46000 & i<48000)
    del = 800;
  else if(i>48000)
    del = 900;
  digitalWrite(DIR,HIGH);
  digitalWrite(ENA,HIGH);
  digitalWrite(PUL,HIGH);
  delayMicroseconds(del);
  digitalWrite(PUL,LOW);
  delayMicroseconds(del);
}
delay(2000);
}
```

## Código G obtenido de un modelo mediante el uso del software Inkscape

```

%
(Header)
(Generated by gcodetools from Inkscape.)
(Using default header. To add your own header create file "header" in the output dir.)
M3
(Header end.)
G21 (All units in mm)

(Start cutting path id: path14556)
(Change tool to Default tool)
M211 S0
G92 X0 Y0 Z0
G00 Z-5.000000
G00 X100.039164 Y99.584974

G01 Z0.500000 F100.0(Penetrare)
G01 X100.039164 Y15.363976 Z0.500000 F400.000000
G01 X74.037046 Y15.363976 Z0.500000
G01 X74.037046 Y18.174598 Z0.500000
G03 X83.115552 Y21.178202 Z0.500000 I-11.033691 J48.571616
G03 X87.720874 Y24.229916 Z0.500000 I-6.609583 J14.975270
G03 X91.110759 Y28.584370 Z0.500000 I-12.093093 J12.911031
G03 X93.818380 Y35.244516 Z0.500000 I-26.559649 J14.678033
G03 X94.559911 Y39.455385 Z0.500000 I-20.692356 J5.814632
G03 X94.984777 Y50.827734 Z0.500000 I-151.988623 J11.372349
G01 X94.984777 Y61.212928 Z0.500000 0
G01 X59.186821 Y61.212928 Z0.500000 0
G01 X59.186821 Y59.216193 Z0.500000 0
G03 X60.799645 Y50.408880 Z0.500000 0 I24.853912 J0.000000
G03 X64.440349 Y45.336711 Z0.500000 0 I11.341207 J4.297800
G03 X70.148926 Y42.117936 Z0.500000 0 I12.083582 J14.758992
G03 X79.935411 Y39.845689 Z0.500000 0 I14.937652 J42.124778
G01 X79.935411 Y37.024215 Z0.500000 0
G01 X33.905403 Y37.024215 Z0.500000 0
G01 X33.905403 Y39.845689 Z0.500000 0
G03 X41.200448 Y41.376569 Z0.500000 0 I-5.007779 J42.010223
G03 X46.223693 Y43.589568 Z0.500000 0 I-6.671706 J21.951567
G03 X50.464245 Y46.943898 Z0.500000 0 I-10.653594 J17.825919
G03 X52.672067 Y50.230883 Z0.500000 0 I-7.951413 J7.725826
G03 X53.792843 Y54.131754 Z0.500000 0 I-14.291888 J6.217707
G03 X54.322092 Y61.212928 Z0.500000 0 I-47.107243 J7.081174
G01 X29.552751 Y61.212928 Z0.500000 0
G03 X22.356863 Y60.933992 Z0.500000 0 I0.000000 J-92.957738
G03 X20.676754 Y60.518412 Z0.500000 0 I0.407845 J-5.252799
G03 X19.273588 Y59.563344 Z0.500000 0 I1.897465 J-4.296002
G03 X18.012059 Y57.881419 Z0.500000 0 I5.012470 J-5.073676
G03 X17.324529 Y55.860325 Z0.500000 0 I5.755883 J-3.085506

```

G03 X16.978422 Y51.641620 Z0.50000 0 I25.537885 J-4.218705  
 G01 X16.978422 Y45.629710 Z0.50000 0  
 G03 X18.664933 Y32.049509 Z0.50000 0 I55.518829 J0.000000  
 G03 X22.687130 Y23.036215 Z0.50000 0 I27.858640 J7.027859  
 G03 X29.527712 Y16.050346 Z0.50000 0 I22.221133 J14.916902  
 G03 X40.059807 Y10.838766 Z0.50000 0 I22.196610 J31.609226  
 G01 X40.059807 Y8.093255 Z0.50000 0  
 G01 X12.113694 Y12.618465 Z0.50000 0  
 G01 X12.113694 Y99.584974 Z0.50000 0  
 G01 X14.512868 Y99.584974 Z0.50000 0  
 G01 X14.512868 Y96.242613 Z0.50000 0  
 G03 X14.923334 Y91.959570 Z0.50000 0 I22.551105 J0.000000  
 G03 X15.878406 Y89.188928 Z0.50000 0 I10.246532 J1.982162  
 G03 X17.059746 Y87.547584 Z0.50000 0 I4.920399 J2.295607  
 G03 X18.988798 Y86.226380 Z0.50000 0 I5.081422 J5.350359  
 G03 X20.619338 Y85.737363 Z0.50000 0 I2.072926 J3.948925  
 G03 X27.153577 Y85.412493 Z0.50000 0 I6.534238 J65.550415  
 G01 X84.999281 Y85.412493 Z0.50000 0  
 G03 X90.946325 Y85.559878 Z0.50000 0 I-0.000000 J120.056576  
 G03 X92.320082 Y85.781455 Z0.50000 0 I-0.312926 J6.309481  
 G03 X94.332996 Y86.852241 Z0.50000 0 I-1.978004 J6.145726  
 G03 X95.828756 Y88.526967 Z0.50000 0 I-3.977229 J5.057533  
 G03 X97.089853 Y91.503478 Z0.50000 0 I-8.059124 J5.169923  
 G03 X97.639990 Y96.242613 Z0.50000 0 I-20.137483 J4.739135  
 G01 X97.639990 Y99.584974 Z0.50000 0  
 G01 X100.039164 Y99.584974 Z0.50000 0  
 G01 X100.039164 Y99.584974 Z0.50000 0  
 G00 Z-5.000000

(End cutting path id: path14556)

(Start cutting path id: path14556)  
 (Change tool to Default tool)

G00 Z-5.000000  
 G00 X100.039164 Y302.079991

G01 Z0.50000 0 F100.0(Penetrate)  
 G01 X100.039164 Y249.470353 Z0.50000 0 F400.000000  
 G01 X97.639990 Y249.470353 Z0.50000 0  
 G01 X97.639990 Y252.801863 Z0.50000 0  
 G03 X97.230777 Y257.086798 Z0.50000 0 I-22.638739 J-0.000000  
 G03 X96.283935 Y259.855548 Z0.50000 0 I-10.362652 J-1.997492  
 G03 X95.098939 Y261.515011 Z0.50000 0 I-5.217286 J-2.472756  
 G03 X93.164060 Y262.894058 Z0.50000 0 I-5.411161 J-5.545261  
 G03 X91.533520 Y263.383075 Z0.50000 0 I-2.072926 J-3.948925  
 G03 X84.999281 Y263.707945 Z0.50000 0 I-6.534238 J-65.550415  
 G01 X27.153577 Y263.707945 Z0.50000 0  
 G03 X20.493716 Y263.374259 Z0.50000 0 I-0.000000 J-66.627240



```
G03 X18.723276 Y262.861503 Z0.50000 0 I0.509559 J-5.072251
G03 X17.294721 Y261.752457 Z0.50000 0 I2.242822 J-4.363545
G03 X15.774094 Y259.519141 Z0.50000 0 I7.950346 J-7.047601
G03 X14.896731 Y256.938158 Z0.50000 0 I8.549480 J-4.345871
G03 X14.512868 Y252.801863 Z0.50000 0 I22.093302 J-4.136295
G01 X14.512868 Y249.470353 Z0.50000 0
G01 X12.113694 Y249.470353 Z0.50000 0
G01 X12.113694 Y302.079991 Z0.50000 0
G01 X14.512868 Y302.079991 Z0.50000 0
G01 X14.512868 Y298.737629 Z0.50000 0
G03 X14.923334 Y294.454587 Z0.50000 0 I22.551105 J-0.000000
G03 X15.878406 Y291.683944 Z0.50000 0 I10.246532 J1.982162
G03 X17.059746 Y290.042601 Z0.50000 0 I4.920399 J2.295607
G03 X18.988798 Y288.721397 Z0.50000 0 I5.081422 J5.350359
G03 X20.619338 Y288.232380 Z0.50000 0 I2.072926 J3.948925
G03 X27.153577 Y287.907510 Z0.50000 0 I6.534238 J65.550415
G01 X84.999281 Y287.907510 Z0.50000 0
G03 X91.664712 Y288.228691 Z0.50000 0 I-0.000000 J69.324090
G03 X93.429581 Y288.721397 Z0.50000 0 I-0.503256 J5.209888
G03 X94.853686 Y289.794239 Z0.50000 0 I-2.016119 J4.157830
G03 X96.378764 Y292.020351 Z0.50000 0 I-7.727039 J6.929147
G03 X97.256127 Y294.601334 Z0.50000 0 I-8.549480 J4.345871
G03 X97.639990 Y298.737629 Z0.50000 0 I-22.093302 J4.136295
G01 X97.639990 Y302.079991 Z0.50000 0
G01 X100.039164 Y302.079991 Z0.50000 0
G01 X100.039164 Y302.079991 Z0.50000 0
G00 Z-5.000000
```

(End cutting path id: path14556)

(Footer)

M5

G00 X0.0000 Y0.0000

M2

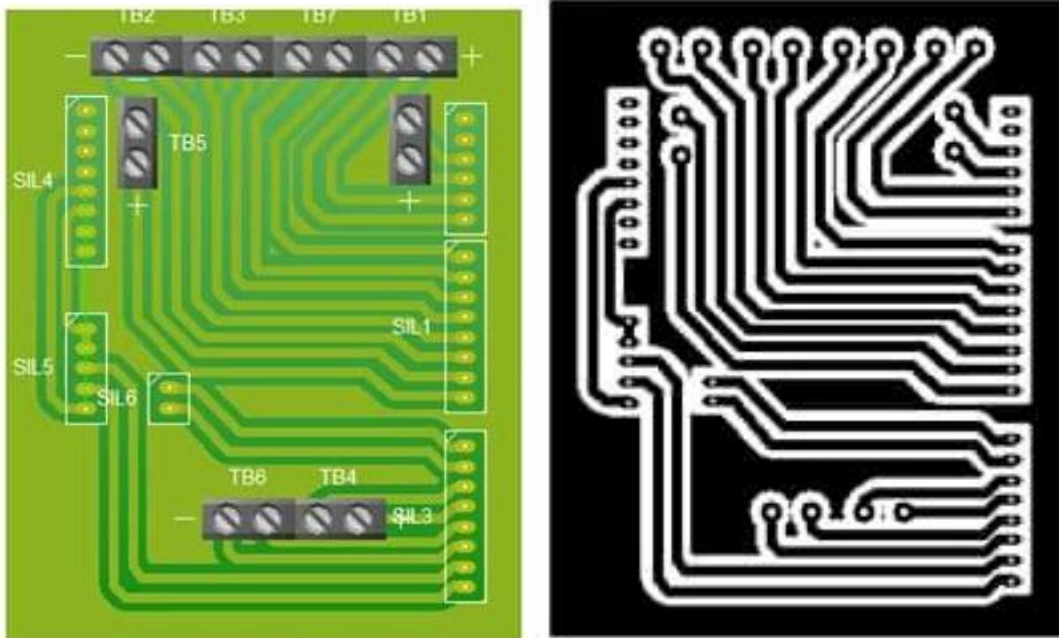
(Using default footer. To add your own footer create file "footer" in the output dir.)

(end)

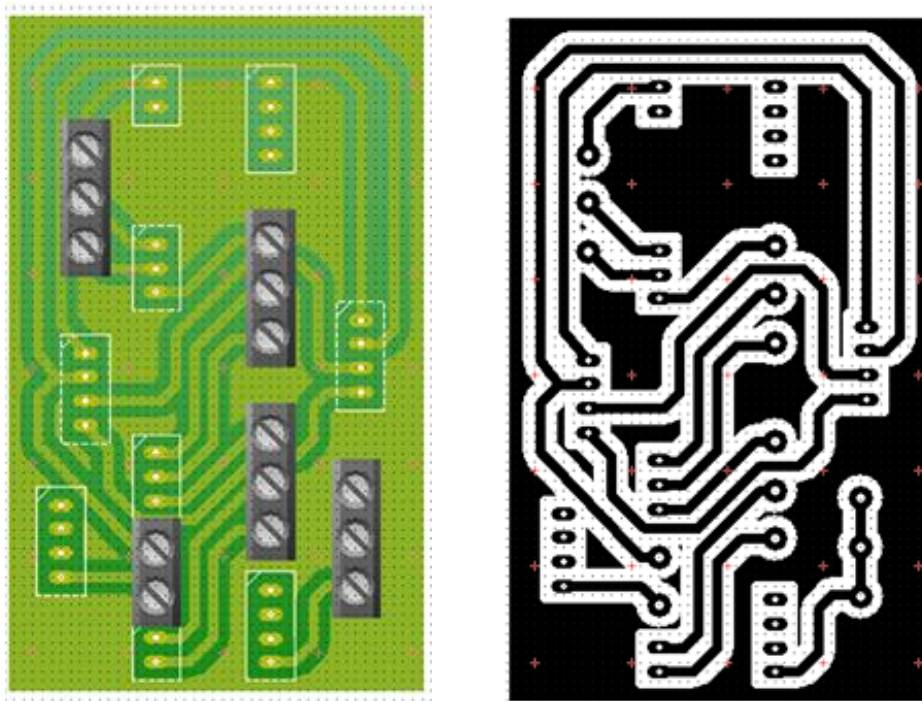
%

### Anexo 3. PCB de las tarjetas auxiliares

#### Tarjeta Auxiliar 1



#### Tarjeta auxiliar 2

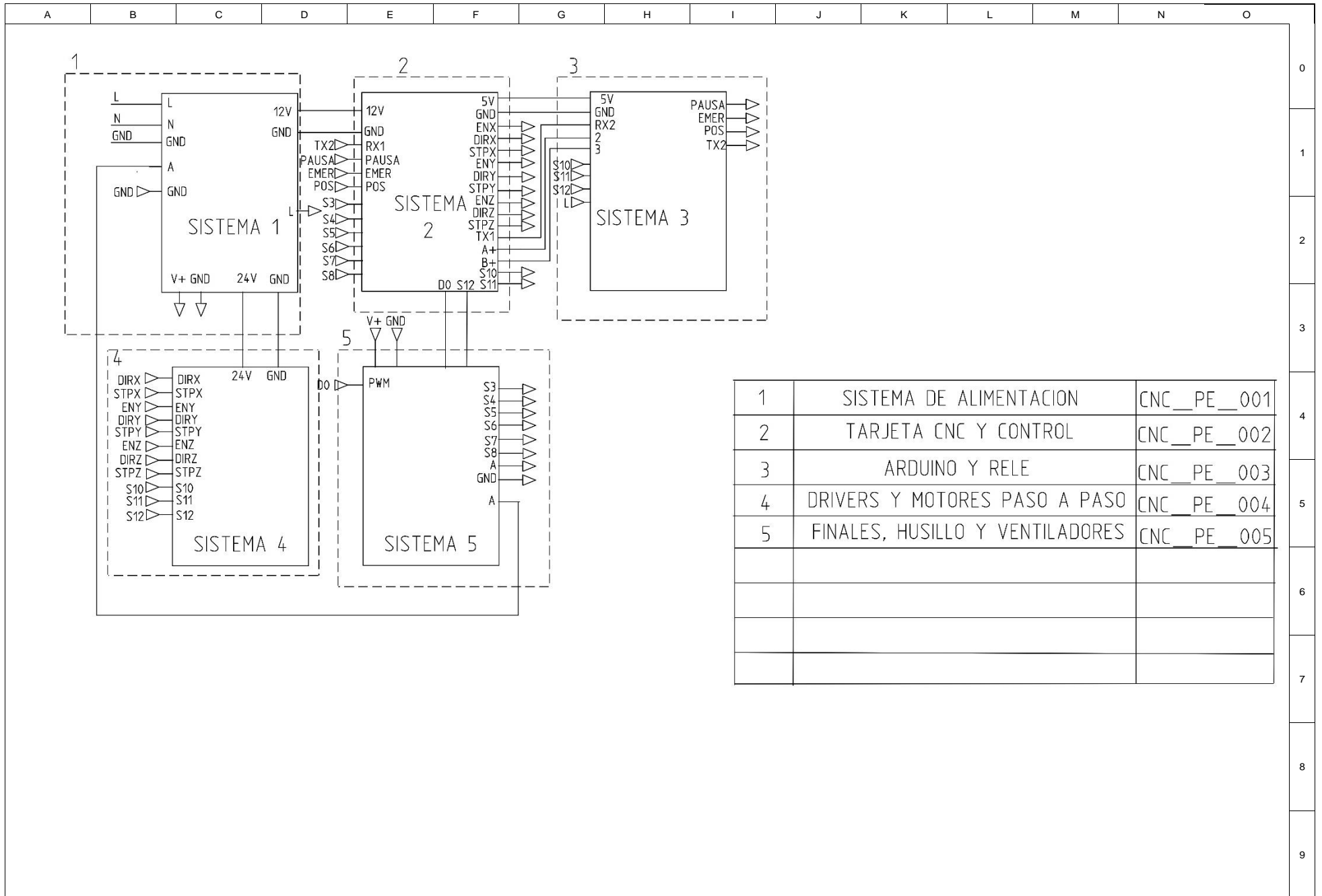


**Anexo 4**  
**Planos Eléctricos**

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
ETIQUETA	COMPONENTE		MARCA		MODELO		FUNCIÓN							
BRK	BREAKER		SCHNEIDER ELECTRIC		C20		PROTECCIÓN CIRCUITO ELÉCTRICO							
S1	SELECTOR 2 POSICIONES						ENCENDIDO DE LA MÁQUINA							
S2	BOTÓN NA		CAMSCO				LLEVAR CERO MÁQUINA/REANUDAR							
S3	BOTÓN NC		CAMSCO				PAUSE							
PWR1	FUENTE DE ALIMENTACIÓN				24 V		FUENTE DE VOLTAJE DC							
PWR2	FUENTE DE ALIMENTACIÓN				12 V		FUENTE DE VOLTAJE DC							
ARD	ARDUINO MEGA		ARDUINO		MEGA		CONTROLADOR CONTROL ALÁMBRICO							
RAMP	RAMPS 1.4		REPRAP		1.4		TARJETA DE CONTROL CNC							
M4	VENTILADOR		ARTIC		F12		EXTRAER CALOR DEL GABINETE							
M5	VENTILADOR						INTRODUCIR AIRE AL GABINETE							
TA1	TARJETA AUXILIAR 1						CONEXIÓN ENTRE PLACA CONTROLADORA CNC Y DRIVERS							
TA2	TARJETA AUXILIAR 2						CONEXIÓN ENTRE PLACA ARDUINO Y CONTROL ALÁMBRICO MANUAL							
S3,S4	FINALES DE CARRERA						PROTECCIONES MECÁNICAS EJE X							
S5,S6	FINALES DE CARRERA						PROTECCIONES MECÁNICAS EJE Y							
S7,S8	FINALES DE CARRERA						PROTECCIONES MECÁNICAS EJE Z							
S9,S10	PARO DE EMERGENCIA						PARO DE EMERGENCIA							
DRV1	DRIVER MOTOR PASO A PASO		STEPPER ONLINE		DM542T		CONTROL MOTOR EJE X							
DRV2	DRIVER MOTOR PASO A PASO		STEPPER ONLINE		DM542T		CONTROL MOTOR EJE Y							
DRV3	DRIVER MOTOR PASO A PASO				TB6600		CONTROL MOTOR EJE Z							
M1	MOTOR PASO A PASO		NEMA 23		HS45-4204S		MOVIMIENTO EJE X							
M2	MOTOR PASO A PASO		NEMA 23		HS45-4204S		MOVIMIENTO EJE Y							
M3	MOTOR PASO A PASO		NEMA 23		HS45-4204S		MOVIMIENTO EJE Z							
			Fecha:		Firma:		Proyecto: Máquina de control numérico computacional para corte de planchas de aluminio por desbaste de material: implementación			Escala: No aplica				
Elaborado por: Héctor Andrés Inbacuán T.							Dibujo			LISTA DE ELEMENTOS			Dibujo 1 / 9	
Revisado por: Ing. Cosme Mejía														

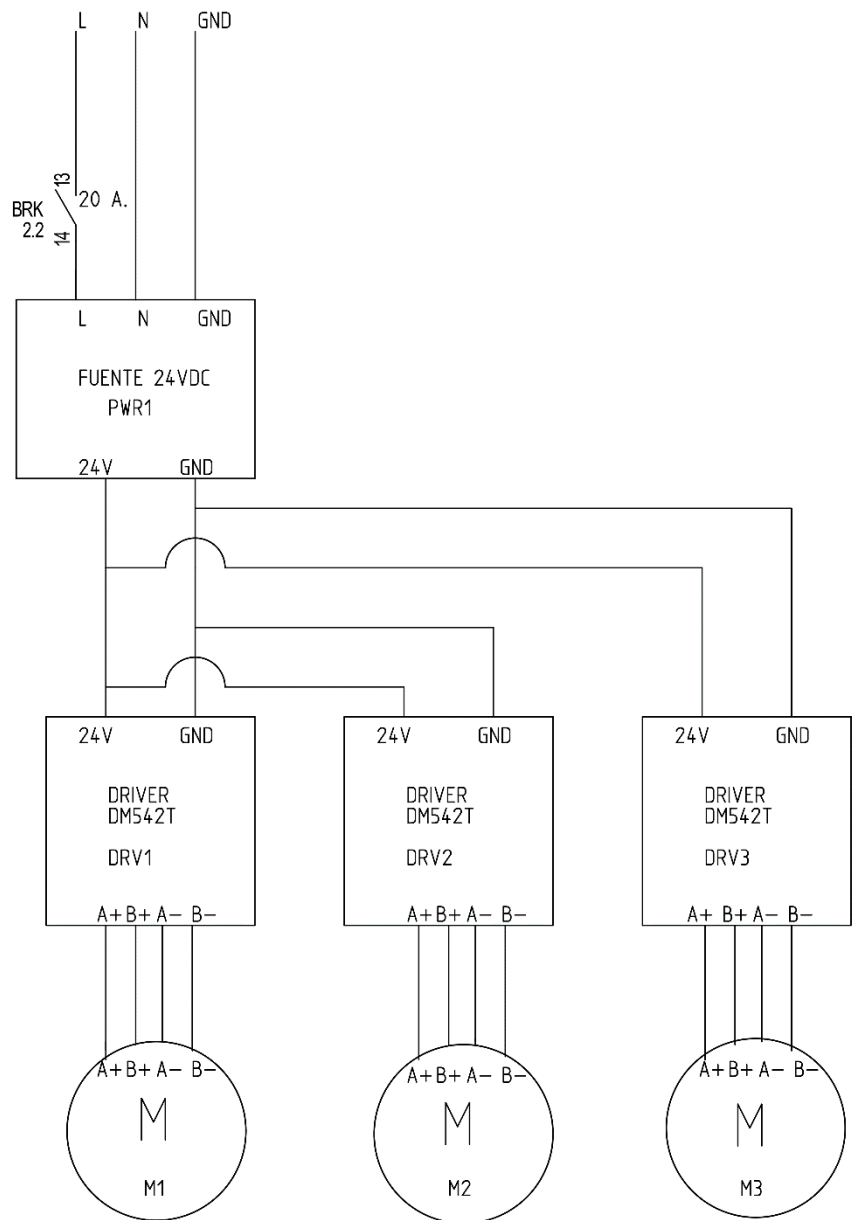
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
ETIQUETA	COMPONENTE		MARCA			MODELO		FUNCIÓN						
S11	SELECTOR 3 POSICIONES							CONTROL VELOCIDAD DE DESPLAZAMIENTO					0	
S12	SELECTOR 4 POSICIONES							SELECCIÓN DE EJE						
ENC	ENCODER							DESPLAZAMIENTO DE EJES					1	
REL	MÓDULO RELÉ							ACCIONAMIENTO DE LUCES PILOTO						
CONV1	CONVERSOR PWM A 0-10V							TRANSFORMAR SEÑAL PWM A 0-10 V					2	
PWR3	FUENTE ALIMENTACIÓN HUSILLO							CONTROL DE VELOCIDAD HUSILLO						
H1	LUZ PILOTO VERDE							SEÑALIZACIÓN DE ENCENDIDO					3	
H2	LUZ PILOTO AMARILLO							SEÑALIZACIÓN DE PAUSA						
H3	LUZ PILOTO ROJO							SEÑALIZACIÓN PARO DE EMERGENCIA						
CONV2	CONVERSOR 12V-5V							CONVERTIR VOLTAJE PARA ALIMENTACIÓN DE CONTROL Y PLACAS ELECTRÓNICAS					4	
LED	LED ROJO							SEÑALIZACIÓN DE CONTROL ENCENDIDO						
M6	MOTOR HUSILLO							ENTREGA DE POTENCIA PARA CORTE					5	
													6	
													7	
													8	
													9	

Elaborado por:		Fecha:	Firma:	Proyecto: Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Implementación	Escala: No aplica
Revisado por:				Dibujo	Dibujo
Héctor Andrés Inbacuán T.				LISTA DE ELEMENTOS	2 / 9
Ing. Cosme Mejía					

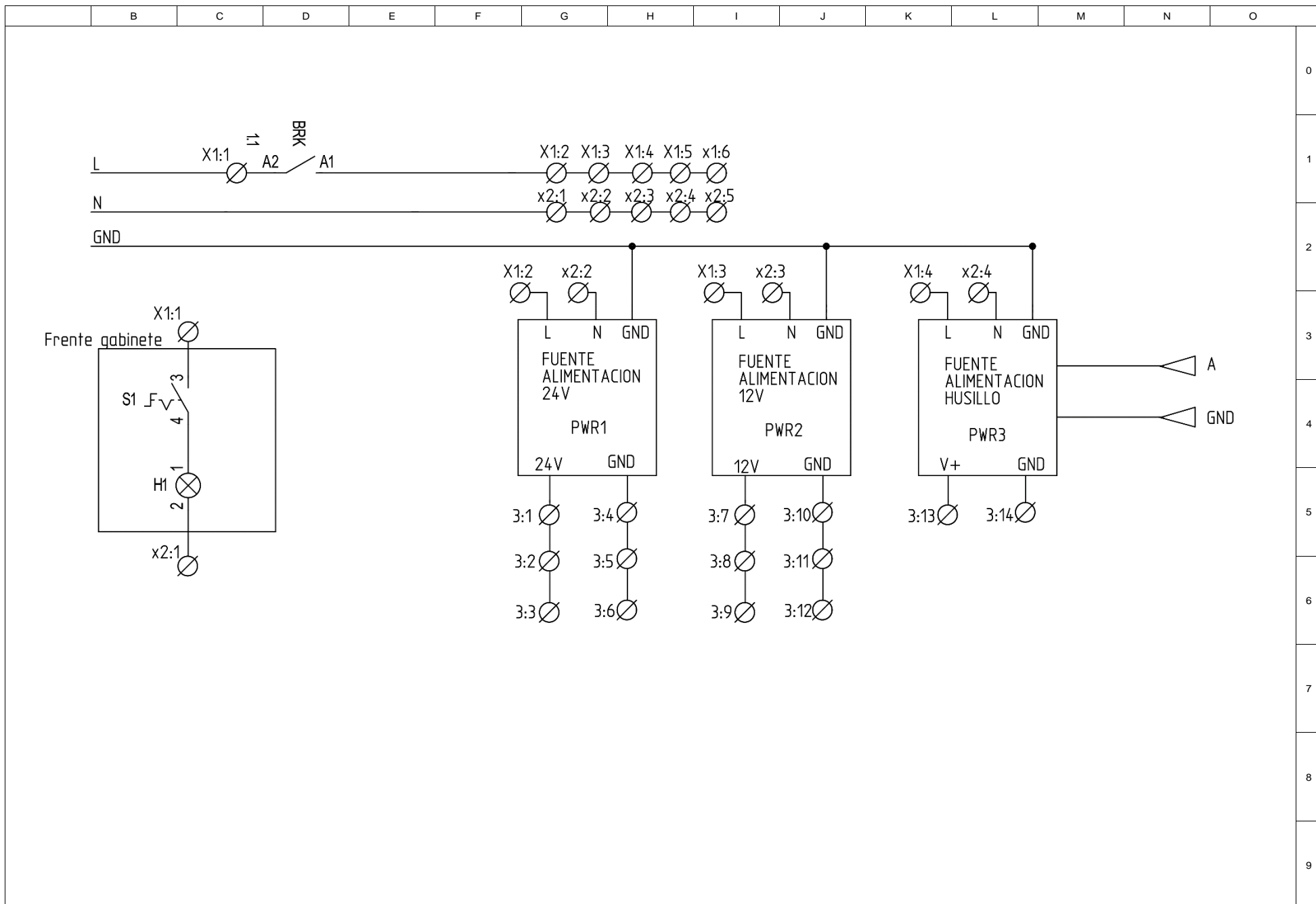


1	SISTEMA DE ALIMENTACION	CNC_PE_001
2	TARJETA CNC Y CONTROL	CNC_PE_002
3	ARDUINO Y RELE	CNC_PE_003
4	DRIVERS Y MOTORES PASO A PASO	CNC_PE_004
5	FINALES, HUSILLO Y VENTILADORES	CNC_PE_005

Elaborado por: Héctor Andrés Inbacuán T.		Fecha:	Firma:	Proyecto: Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Implementación	Escala: No aplica
Revisado por: Ing. Cosme Mejía				Dibujo Plano General	Dibujo 3 / 9

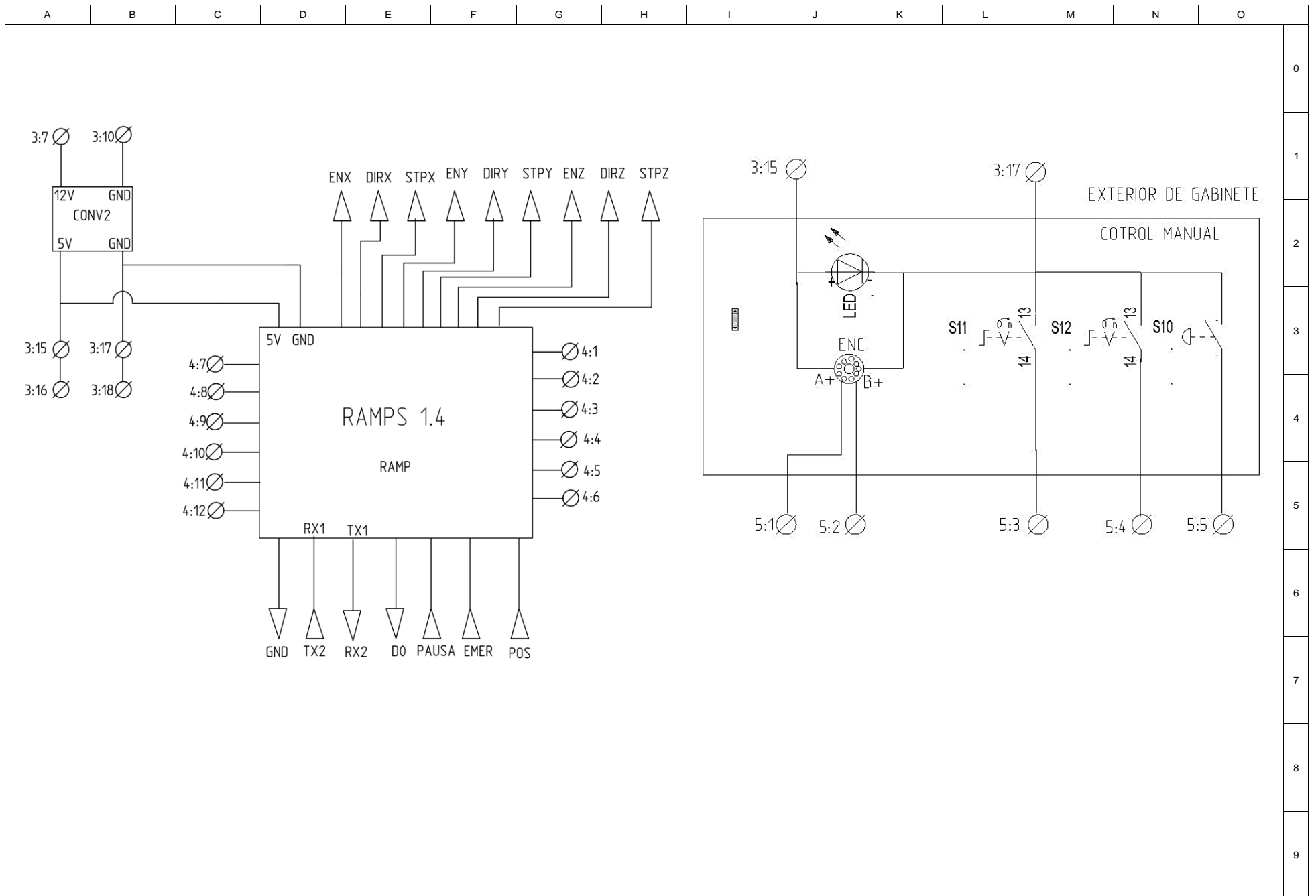


		Fecha:	Firma:	Proyecto: Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Implementación	Escala: No aplica
Elaborado por:	Héctor Andrés Inbacuán T.			Dibujo CNC_DF_001 Diagrama de fuerza	Dibujo 4 / 9
Revisado por:	Ing. Cosme Mejía				

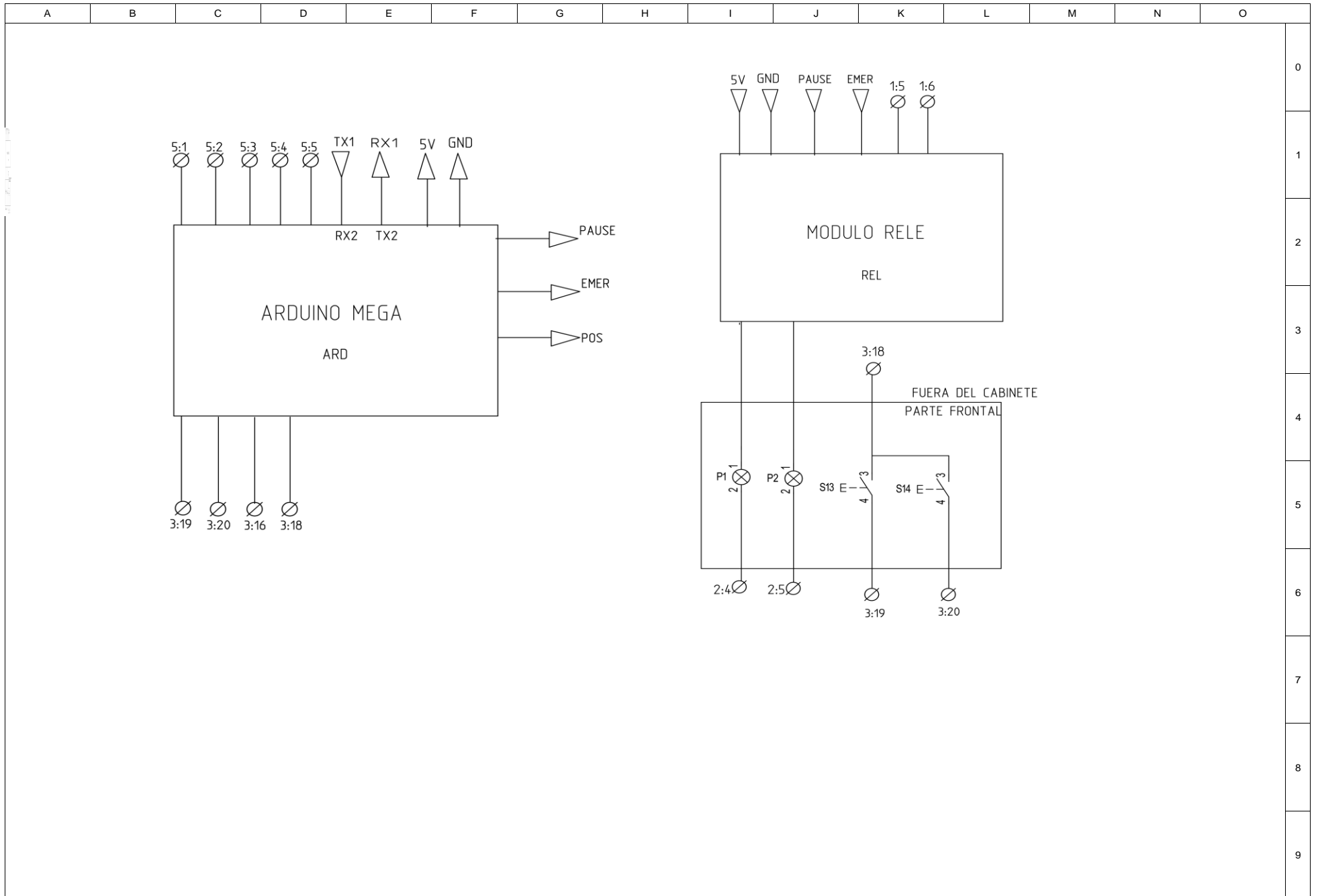


	Fecha:	Firma:	Proyecto: Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Implementación	Escala: No aplica
Elaborado por: Héctor Andrés Inbacuán T.			Dibujo CNC_PE_001 Alimentación del Sistema	Dibujo 5 / 9
Revisado por: Ing. Cosme Mejía				

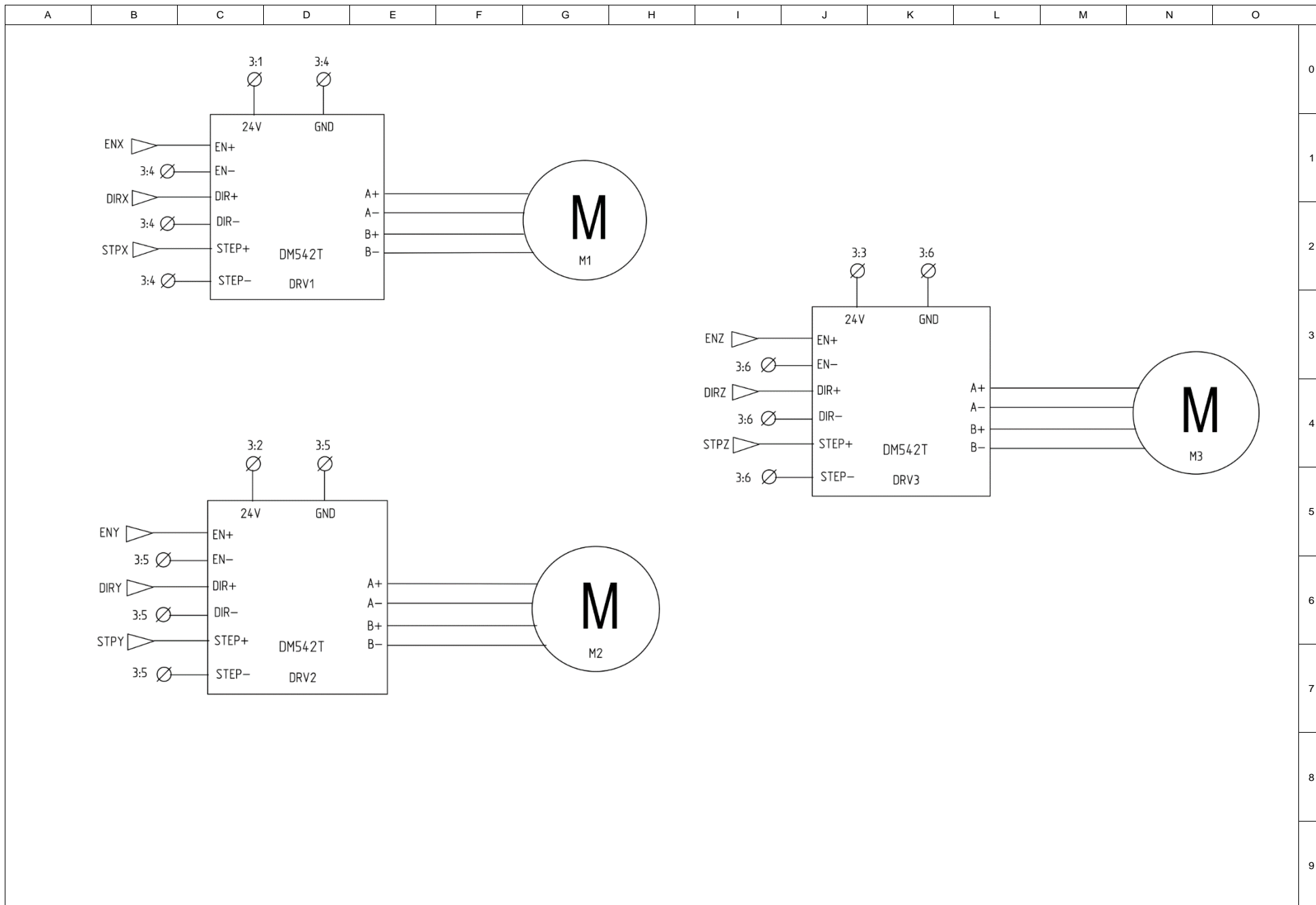




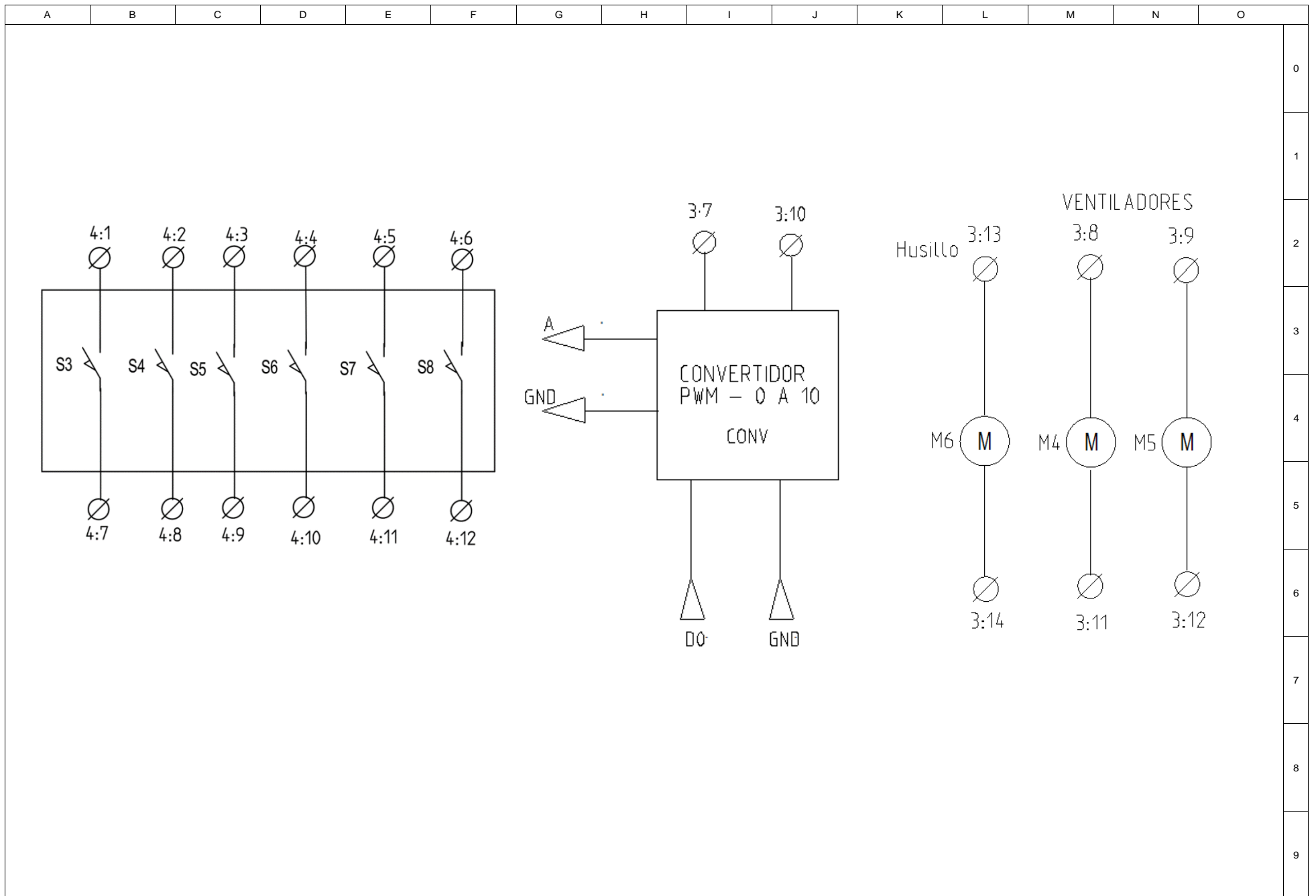
	Fecha:	Firma:	Proyecto: Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Implementación	Escala: No aplica
Elaborado por: Héctor Andrés Inbacuán T.			Dibujo CNC_PE_002	Dibujo 6 / 9
Revisado por: Ing. Cosme Mejía			Tarjeta de Control Numérico Computacional y control manual.	



	Fecha:	Firma:	Proyecto: Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Implementación	Escala: No aplica
Elaborado por: Héctor Andrés Inbacuán T.			Dibujo CNC_PE_003	Dibujo 7 / 9
Revisado por: Ing. Cosme Mejía			Tarjeta Arduino y Módulo relé	



	Fecha:	Firma:	Proyecto: Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Implementación	Escala: No aplica
Elaborado por: Héctor Andrés Inbacuán T.			Dibujo CNC_PE_004 Drivers y motores Paso a Paso	Dibujo 8 / 9
Revisado por: Ing. Cosme Mejía				



	Fecha:	Firma:	Proyecto: Máquina de Control Numérico Computacional para corte de planchas de aluminio por desbaste de material: Implementación	Escala: No aplica
Elaborado por: Héctor Andrés Inbacuán T.			Dibujo CNC_PE_005	Dibujo 9 / 9
Revisado por: Ing. Cosme Mejía			Finales de carrera, conversor PWM, ventiladores y husillo	

# **MANUAL DE FUNCIONAMIENTO DE LA MÁQUINA DE CNC PARA CORTE DE PLANCHAS DE ALUMINIO POR DESBASTE DE MATERIAL**

El presente documento tiene como finalidad brindar la información necesaria para el correcto uso de la maquina CNC por parte del operario. Todos los procesos descritos en este manual son de vital importancia para garantizar que el corte de piezas tenga el mayor grado de calidad.

la tecnología CNC tiene un grado de dificultad elevado, por lo que se recomienda que la persona encargada del uso de la máquina tenga conocimientos sobre sistemas CNC.

## **1. Requerimientos para el uso de la máquina**

- Energía eléctrica 110VAC 60Hz.
- Equipo de protección personal.
- Fresa para corte de aluminio.

Con los requisitos mencionados anteriormente se puede proceder al uso de la máquina teniendo en cuenta los siguientes procesos a seguir:

1. Verificar que el selector se encuentre en la posición OFF.
2. Colocar la herramienta a utilizarse.
3. Sujetar el material a cortar mediante las mordazas con firmeza para evitar movimiento en el transcurso del proceso de corte.
4. Conectar el enchufe de la maquina a una conexión de energía eléctrica.
5. Encender la máquina mediante el posicionamiento del selector en ON. Cerciorarse que la luz verde este encendida.
6. Verificar que no exista ningún elemento que obstaculice el desplazamiento de la herramienta.
7. Tomar el control manual y ubicar el selector de eje y velocidad a las posiciones deseadas.
8. Desplazarse a lo largo del área de trabajo a una velocidad alta siempre y cuando la herramienta se encuentra lejos del material a cortarse, si la herramienta se encuentra cerca, se recomienda la utilización de las velocidades X1 y X10.

9. Tomar el cero pieza, es decir, ubicarse lo mas cerca del elemento a cortar sin que la herramienta se choque, para este proceso se puede utilizar un papel y colocarlo entre la herramienta y el material e irse acercando poco a poco, se recomienda el uso de la velocidad X1 para realizar esta operación, mientras el papel no se presione entre la herramienta y el papel irse acercando paulatinamente, una vez que el papel se presione y no se mueva retiramos el papel y ese punto será considerado nuestro cero pieza.
10. Una vez ubicado el cero pieza, es necesario establecer esa posición en nuestra máquina, para lo cual existen dos métodos; el primer método es que tengamos la línea de posicionamiento de cero pieza dentro de nuestro código G y así no necesitamos realizar ninguna configuración en la máquina. El segundo método es utilizar la pantalla LCD entrando al menú y seleccionando la opción Comandos Personalizados, se nos mostrará algunas opciones, seleccionamos la opción “Set 0 pieza”. Una vez hecho este proceso tenemos establecido el cero pieza dentro de nuestra máquina.
11. Cargar el archivo con el código G en nuestra máquina, este proceso se lo realiza mediante una memoria SD en el cual debe encontrarse nuestro archivo con una extensión .GCODE, necesario para que sea reconocido por el intérprete de código G. para cargar el archivo nos dirigimos al menú de la pantalla SD, seleccionamos “MENÚ SD” y se mostrara los archivos que se encuentran dentro de nuestra memoria, ubicar el archivo que vamos a cortar, al seleccionarlo empezara a realizar las acciones que se encuentren dentro del código.
12. Revisar que el proceso de corte se realice de manera adecuada, evitar introducir las manos durante el corte, de existir algún inconveniente presionar el botón de paro de emergencia.
13. Una vez terminado el corte y si la herramienta se encuentra cerca de la pieza realizar un desplazamiento hasta que se encuentre en un punto donde no represente un riesgo para el operario.
14. Desmontar la pieza cortada, limpiarla con una brocha y de ser necesario realizar un proceso de pulido. Limpiar el área de trabajo.
15. Si se va a realizar otro proceso de corte realizar los mismo pasos establecidos anteriormente, si ya no se va a realizar un corte apagar la máquina llevando el selector que se encuentra en el gabinete hacia la posición OFF.

## MANUAL DE MANTENIMIENTO

Para el correcto funcionamiento de la máquina es necesario realizar un mantenimiento periódico de los elementos electrónicos y mecánicos.

### Mantenimiento mecánico:

<b>Mantenimiento</b>	<b>Frecuencia</b>	<b>Actividad</b>
Guías de desplazamiento ejes X,Y,Z	Después de cada uso	Limpiar el material residual del proceso de corte (viruta) de los elementos de los sistemas de movimiento: guías, tornillo, con una tela limpia.
Mesa de trabajo	Después de cada uso	Limpiar la mesa de trabajo con una escobilla, o aspiradora antes de limpiar las guías del eje Y.
Rodamientos lineales	Antes de cada uso	Revisar que estén libres de viruta, en caso de presentarse limpiar con una tela limpia.
Pernos y tornillos	Antes de cada uso	Revisar que estén sujetos adecuadamente.
Elementos del sistema de movimiento	Cada 6 meses	Extraer o desarticular todos los elementos de los sistemas de movimiento, para un mantenimiento preventivo, limpiando en su totalidad, rodamientos, guías, tuercas, tornillos de transmisión de potencia y engrasándolos según sea necesario, revisar los rodamientos lineales y cambiarlos si se necesita.
Elementos de sujeción	Cada 6 meses	Extraer cada elemento, acoples de motor a tornillo, chumaceras, sujetadores de pared, revisarlos y cambiarlos si están defectuosos.
Fresas	Antes de cada uso	Verificar que la fresa tenga un filo adecuado, cambiarlas si es necesario.

## Mantenimiento Eléctrico-Electrónico:

<b>Mantenimiento</b>	<b>Frecuencia</b>	<b>Actividad</b>
Cable de poder	Antes de cada uso	Revisar que el cable no presente discontinuidades en su recubrimiento exterior y cambiarlo si realiza falsos contactos.
Gabinete de control	Mensual	Verificar la continuidad entre las conexiones de los elementos dentro del gabinete, si no se establece continuidad entre alguna conexión, verificar la continuidad entre terminales o cambiar completamente el cable.
Protecciones	Mensual	Desconectar los elementos de la etapa de alimentación y realizar un cortocircuito controlado después del breaker. Si la protección no responde a este proceso, cambiarlas.
Tarjetas electrónicas	Cada 6 meses	Desconectar las tarjetas electrónicas y realizar una limpieza exhaustiva, removiendo el polvo con una escobilla y limpiar contactos.
Finales de carrera	Antes de cada uso	Verificar la señal de cada final de carrera manualmente cuando se encuentre encendida la máquina, presionándolos y verificando que la luz indicadora se encuentre encendida, cambiarlos si es necesario.



**Cuadro de posibles fallas:**

<b>Fallas</b>	<b>Posibles razones</b>	<b>Soluciones</b>
La máquina no enciende	<ul style="list-style-type: none"><li>- Cable de alimentación desconectado, mal conectado o defectuoso.</li><li>- Breaker con contacto abierto.</li></ul>	<ul style="list-style-type: none"><li>- Conectar el cable correctamente, si esta defectuoso, cambiarlo.</li><li>- Cambiar la configuración del breaker a cerrado, si el problema persiste cambiar de protección.</li></ul>
Que los motores no enciendan	<ul style="list-style-type: none"><li>- Mal conexión entre terminales</li><li>- Drivers defectuosos</li><li>- Tarjeta auxiliar no entrega la señal a los drivers</li></ul>	<ul style="list-style-type: none"><li>- Revisar la conexión entre terminales, si el problema persiste revisar la continuidad y reemplazar los cables defectuosos.</li><li>- Cambiar los drivers con mal funcionamiento</li><li>- Intercambiar la tarjeta electrónica.</li></ul>
Que los ejes no se muevan	<ul style="list-style-type: none"><li>- Contacto entre la torre y un final de carrera.</li><li>- Que el eje este sujeto o mal acoplado.</li><li>- Que el tornillo esté deteriorado.</li><li>- Que el acople flexible se rompa.</li></ul>	<ul style="list-style-type: none"><li>- Apagar la máquina y mover manualmente el eje, posteriormente realizar homing.</li><li>- Verificar que los prisioneros del acople o la chumacera este sujetos adecuadamente, verificar que las tuercas estén sujetas con el prisionero.</li><li>- Cambiar el tornillo apenas los dientes de la rosca presenten un deterioro o los sistemas de movimiento presenten un movimiento forzado.</li><li>- Cambiarlo por otro similar, tomando en consideración la relación del eje de 10mm a 16mm</li></ul>

Que no realice homing	<ul style="list-style-type: none"> <li>- Que el botón de homing no se encuentre conectado.</li> <li>- Que el botón no funcione.</li> </ul>	<ul style="list-style-type: none"> <li>- Verificar la conexión del botón a la placa electrónica, medir continuidad presionando el botón.</li> <li>- Cambiarlo inmediatamente.</li> </ul>
Que no cargue el archivo con el código G	<ul style="list-style-type: none"> <li>- Incompatibilidad de extensión.</li> </ul>	<ul style="list-style-type: none"> <li>- Revisar la extensión del archivo que sea en .GCODE y volver a cargarlo.</li> </ul>
Pantalla LCD no enciende	<ul style="list-style-type: none"> <li>- Mal conexión.</li> <li>- Daños en el controlador LCD.</li> </ul>	<ul style="list-style-type: none"> <li>- Revisar la correcta conexión entre la pantalla y la placa de control CNC.</li> <li>- Cambiar la LCD.</li> </ul>
Que el control manual para desplazamiento de ejes no funcione	<ul style="list-style-type: none"> <li>- Conexión inadecuada de conector VGA.</li> <li>- Cable UTP de 16 hilos dañado.</li> <li>- Desconexión de terminales hacia la placa.</li> </ul>	<ul style="list-style-type: none"> <li>- Revisar que el conector VGA se encuentre debidamente conectado y en buen estado.</li> <li>- Revisar la continuidad de cada hilo, desmontando el control manual de movimiento, cambiar todo el cable si es necesario.</li> <li>- Revisar la continuidad entre el conector VGA y la placa electrónica.</li> </ul>
Que el husillo no encienda	<ul style="list-style-type: none"> <li>- Mala conexión con el driver.</li> <li>- Mal funcionamiento del conversor de PWM a 0V-10V.</li> <li>- Que no exista el comando de revoluciones dentro del archivo de código G</li> </ul>	<ul style="list-style-type: none"> <li>- Revisar la continuidad entre terminales y cambiarlos si es necesario.</li> <li>- Realizar la medición de prueba del conversor enviando varias velocidad por código G y verificando el voltaje.</li> <li>- Insertar el comando M3 seguido de las revoluciones teniendo en cuenta el rango permitido de 0-12000 rpms</li> </ul>