



**Universidad
de La Laguna**

ESCUELA SUPERIOR DE INGENIERÍA Y TECNOLOGÍA

Grado en Ingeniería Electrónica Industrial y Automática

Trabajo Fin de Grado

**Diseño de máquina CNC: diseño
estructural, implementación de software y
hardware y puesta en marcha**

Autor: Miguel Perera García

Tutor: Santiago Torres Álvarez

Julio2023

Agradecimientos

A todo aquel que en mayor o en menor medida haya contribuido a que este trabajo acabe en buen puerto. En especial a mi familia y a mi tutor del trabajo Santiago Torres.

Resumen

La industria se ha visto revolucionada gracias al desarrollo de las máquinas CNC al mejorar la precisión, repetibilidad y velocidad en la fabricación de productos. El desarrollo de estas ha permitido la producción de piezas idénticas a gran velocidad, sin supervisión y sin tiempos de inactividad.

Este trabajo trata de desarrollar una máquina CNC de bajo costo, abarcando las etapas desde el diseño hasta la puesta en marcha. Para comprender las aplicaciones de las máquinas CNC en la industria, se ha realizado un estudio de mercado. Además, se estudiaron las cinemáticas tanto inversa como directa necesarias para el funcionamiento. Mediante un software de modelado 3D se diseñaron las piezas, lo que permitió visualizar la estructura y mecanismos de la máquina

También se realizó la implementación y puesta en marcha, implementando la electrónica y el software de control que permite el movimiento. Durante esta fase, se realizaron pruebas y ajustes para mejorar el rendimiento.

Este proyecto trata de demostrar que la construcción de una CNC de bajo presupuesto es viable. Y ofrecer una alternativa accesible para sectores que deseen incorporar la tecnología CNC en sus procesos de fabricación

Abstract

The industry has revolutionised thanks to the development of CNC machines to improve precision, repeatability and speed in the manufacture of products. Their development has allowed the production of identical parts at high speed, continuously and without any supervision.

This work tries to develop a low-cost CNC machine, covering the stages from designing to commissioning. To understand the applications of CNC machines in the industry, a market study has been done. Also, the study of both the inverse and forward kinematics was required for operation. The different pieces for the structure were designed using 3D modelling software, which allows the structure and mechanisms of the machine to be visualised.

The start-up and step-up of the machine were also carried out, implementing the electronics and the control software that allows the movement. During this phase, some tests were performed and tuned to improve the general performance of the prototype.

This project tries to demonstrate that the construction of a low-budget CNC is viable. This offers an accessible alternative for sectors to incorporate CNC technology in addition to manufacturing processes.

Índice

Resumen	3
Abstract	4
Índice	5
Índice de figuras	7
1. Introducción	9
1.1. Estado actual de las CNC	9
1.2. Estudio de mercado	10
1.2.1. Opciones comerciales	13
2. Objetivos	20
2.1. Metodología	20
3. Modelo cinemático	21
3.1. Cinemática directa	21
3.2. Cinemática Inversa	22
4. Propuesta de diseño	23
4.1. Software Ultimaker Cura	24
4.2. Diseño de la estructura	26
4.3. Perfiles de la base del robot	27
4.4. Tornillo sin fin y tuerca	28
4.5. Anclaje motor eje Z	29
4.6. Sujeción tornillo sin fin eje Z	30
4.7. Soporte tuerca de tornillo sin fin eje Z	30
4.8. Soporte perfil aluminio eje Z	31
4.9. Soporte tuerca tornillo sin fin ejes X e Y	32
4.10. Soporte motor eje Y	33
4.11. Soporte tornillo sin fin eje Y	34
4.12. Soporte motor eje X	35
4.13. Soporte tornillo sin fin eje X	36
5. Sistema de actuación	37
5.1. Dimensionamiento motor eje Z	37
5.2. Cálculo de velocidad máxima de los motores	40
6. Montaje y electrónica.	41
6.1. Montaje.	41
6.2. Electrónica.	45
6.2.1. Arduino Mega 2560	45
6.2.2. Ramps 1.4	45
6.2.3. Drivers A4988	46
6.2.4. Finales de carrera.	48
6.2.5. Conexión de los componentes.	48
7. Software de arduino mega 2560.	51
8. Generación de GCode con Inkscape.	58
9. Pronterface.	66

10. Planos.	67
11. Presupuesto.	78
12. Conclusiones.	79
13. Conclusions.	80
14. Bibliografia.	81
15. Anexos.	83

Índice de figuras

Figura 0. Robot Cartesiano [4].	11
Figura 1. Robot delta [3].	12
Figura 2. Robot SCARA [8].	12
Figura 3. Robot Antropomórfico [8].	13
Figura 4. Scara ABB “working space” [9].	14
Figura 5. Antropomorfo ABB “working space” [9].	15
Figura 6. Antropomórfico YEC [10].	16
Figura 7. Delta YEC [10].	16
Figura 8. Scara YEC [10].	17
Figura 9. Cartesiano KUKA [11].	17
Figura 10. Delta KUKA [11].	18
Figura 11. Scara KUKA [11].	19
Figura 12. Antropomorfo KUKA [11].	19
Figura 13. Posición efector final.	21
Figura 14. Impresora “LulzBot TAZ 6” [15].	23
Figura 15. Software Ultimake Cura.	24
Figura 16. Previsualización de la impresión con las especificaciones.	25
Figura 17. Imagen de impresión 1.	25
Figura 18. Imagen de impresión 2.	26
Figura 19. Diseño basado en tornillo sin fin.	26
Figura 20. Perfil V-Slot 2040.	27
Figura 21. Perfil V-Slot 2020.	28
Figura 22. Tuerca tornillo sin fin.	28
Figura 23. Tornillo sin fin.	29
Figura 24. Anclaje motor eje Z.	29
Figura 25. Sujeción tornillo sin fin eje Z.	30
Figura 26 y 27. Soporte tuerca de tornillo sin fin eje Z.	30
Figura 28. Soporte perfil aluminio eje Z.	31
Figura 29 y 30. Soporte tuerca tornillo sin fin ejes X e Y.	32
Figura 31. Soporte motor eje Y.	33
Figura 32. Soporte tornillo sin fin eje Y.	34
Figura 33. Soporte motor eje X.	35
Figura 34. Soporte tornillo sin fin eje X.	36
Figura 35. Dimensiones tornillo sin fin (mm).	37
Figura 36. Diagrama sólido rígido del tornillo sin fin del eje Z.	38
Figura 37. Datos del Fabricante Motores Nema17.	39
Figura 38. Ruedas apoyadas sobre los perfiles.	42
Figura 39. Rodamientos encajados en los soportes.	42
Figura 40. Motores encajados en los soportes.	43
Figura 41. Creación de rosca en los perfiles.	43
Figura 42. Tuerca del tornillo sin fin encajada en los soportes.	44

Figura 43. Unión tornillo sin fin con los motores.	44
Figura 44. Montaje completo visto desde el eje X.	45
Figura 45. Montaje completo visto desde arriba.	45
Figura 46. Arduino mega 2560.	46
Figura 47. Ramps 1.4 con jumpers conectados en configuración 1/16 de paso.	47
Figura 48. Driver A4988 con su disipador.	48
Figura 49. Instrucciones del datasheet.	48
Figura 50. Sensor final de carrera.	49
Figura 51. Tabla de micropasos según los jumpers [18].	50
Figura 52. Conexión de la ramps 1.4 [18].	50
Figura 53. Conexión de los finales de carrera [18].	51
Figura 54. Código de la librería Boards.h.	52
Figura 55. Primera modificación.	53
Figura 56. Segunda modificación.	53
Figura 57. Tercera modificación.	54
Figura 58. Cuarta modificación.	54
Figura 59. Quinta modificación.	55
Figura 60. Sexta modificación.	55
Figura 61. Séptima modificación.	55
Figura 62. Octava modificación.	56
Figura 63. Novena modificación.	56
Figura 64. Décima modificación.	56
Figura 65. Undécima modificación.	57
Figura 66. Duodécima modificación.	57
Figura 67. Pins_RAMPS.h.	58
Figura 68. Programa Inkscape.	59
Figura 69. Unidades y dimensiones del espacio de Inkscape.	60
Figura 70. Imagen convertida a mapa de bits.	61
Figura 71. Pestaña de "Trayecto".	62
Figura 72. Puntos de orientación.	63
Figura 73. Biblioteca de herramientas.	64
Figura 74. Parámetros de la CNC.	64
Figura 75. Información completa.	65
Figura 76. Trayecto a GCode.	66
Figura 77. GCode.	66
Figura 78. Interfaz de Pronterface.	67

1. Introducción

Una máquina CNC se define como una herramienta motorizada y maniobrable controlada por un ordenador según las instrucciones de entrada. El CNC o control numérico computerizado, ofrece una productividad mucho mayor que el manual en los procesos repetitivos, esta ventaja tiene un costo significativo en costes [4]. Las instrucciones de forma secuencial como el GCode siguiendo los estándares ISO- 6983 [5].

El desarrollo de las máquinas CNC (Control Numérico Computerizado) provocó una revolución en la industria, en la forma en que se fabrican los productos. Las máquinas CNC han permitido a los fabricantes una mayor precisión, repetibilidad y velocidad en los procesos industriales permitiendo una mayor producción [1].

Las ventajas que presentan las máquinas CNC frente a un trabajador (la precisión, repetibilidad y velocidad), permiten producir piezas idénticas repetidamente a una gran velocidad, lo que las hace ideales para la producción en masa. Otra cualidad añadida es la capacidad de las máquinas CNC de trabajar sin ningún tipo de supervisión, permitiendo producir de forma continua, reduciendo el tiempo de producción al no tener tiempos de inactividad [2].

Actualmente las máquinas CNC son capaces de realizar diferentes procesos en una misma pieza lo que reduce altamente los tiempos y costes. Productos más sofisticados han sido posibles gracias a la implementación de estas máquinas en la industria, ya que son capaces de procesar piezas que de forma manual serían imposibles de realizar. La flexibilidad de las CNC permite una mayor variedad de producción [2].

Actualmente, los robots son muy usados en la industria y de muchos tipos (el robot cartesiano, el robot antropomórfico, el robot delta o el robot SCARA) con el objetivo de automatizar los procesos y aumentar la productividad. Pero para que un robot mejore el rendimiento de producción es vital que estén controlados por un ordenador [3].

1.1. Estado actual de las CNC

A comienzos del siglo XIX, con la revolución industrial, se comienzan a automatizar los procesos industriales sustituyendo la mano de obra por máquinas. Durante sus comienzos las máquinas eran controladas por tarjetas perforadas y no es hasta la década de los 60 del siglo siguiente cuando se comienzan a incorporar los ordenadores a los procesos industriales [6].

Algunas de las ventajas de los procesos automatizados son [2][7]:

1. Mejora de la eficiencia: La automatización permite una mayor eficiencia en la producción y una reducción de los tiempos muertos.
2. Aumento de la calidad: La automatización permite un mayor control y monitoreo de los procesos, lo que a su vez conduce a una producción de mayor calidad.

3. Reducción de costos: La automatización permite reducir los costos asociados con la mano de obra, ya que reduce la necesidad de tener una fuerza laboral para realizar tareas repetitivas.
4. Mejora de la seguridad: La automatización puede mejorar la seguridad en la fábrica, ya que reduce la necesidad de tener trabajadores realizando tareas peligrosas.
5. Mejora de la flexibilidad: La automatización permite una mayor flexibilidad en la producción, ya que los procesos pueden ser modificados y adaptados fácilmente a las necesidades cambiantes del mercado.

En resumen, el estado actual de las máquinas CNC es de constante evolución y mejora, con un enfoque en la precisión, la velocidad, la automatización y la interacción hombre-máquina. La tecnología continúa avanzando y mejorando, lo que llevará a una mayor eficiencia y productividad en la industria.

1.2. Estudio de mercado

A la hora de seleccionar un robot industrial para una aplicación específica, es de vital importancia analizar qué tipo de trabajo va a realizar, y elegir el modelo que más se adecue a su tarea. Entre los parámetros que se deben de tener en cuenta están: la carga que tendrá que mover, la precisión requerida, los grados de libertad necesarios, etc.

En este trabajo se analizan los robots industriales más comunes en la actualidad, con el objetivo de contextualizar el diseño elegido en este proyecto. Los tipos de robots analizados son los antropomórficos, los cartesianos, los robots tipo SCARA y los tipo Delta. Todos ellos presentan una serie de ventajas y desventajas respecto a los demás, como son la velocidad, la precisión, la carga de trabajo, el alcance o el precio [3][8].

La elección del robot dependerá del trabajo que va a desempeñar y de los requisitos que se quieran para el desempeño de dicho trabajo. Se analizarán las opciones de mercado para determinar la más adecuada en cuanto a costos y eficiencia, siempre que cumplan los requisitos para el trabajo.

A. Robot Cartesiano

Los robots cartesianos (figura 0) son robots lineales, es decir, se mueven sobre sus ejes, por lo que son muy fáciles de programar y son escalables sin necesidad de realizar cambios significativos. El mayor problema de este robot es que solamente posee tres grados de libertad. Su característica principal es su alta precisión en toda su zona de trabajo.

Usos comunes del robot cartesiano:

- Packaging,
- Manipulación de materiales,
- Mecanizado,
- Pick & Place,
- Sector agroalimentario,
- Apilado,
- Manipulación, montaje y apilado,

- Corte por láser y por chorro de agua,
- Mesa XY,
- Grabado.

En caso de trabajar con cargas más pesadas que la media, se debería considerar la utilización de robots cartesianos de pórtico. Más imponentes, estos robots tienen una carga útil y un alcance mayores.



Figura 0. Robot Cartesiano [4].

B. Robot Delta

Los robots delta (figura 1) basan su funcionamiento en paralelogramos articulados conectados a una base común. Esta configuración permite una alta precisión en sus movimientos, y son ideales para trabajos que requieran un gran nivel de precisión y velocidad de forma simultánea. Los robots delta pueden llegar a tener hasta cinco grados de libertad. Una de sus principales características es su relación de carga de trabajo con su propio peso, logrando mover un peso considerable teniendo en cuenta su peso reducido. Su mayor desventaja es el alto coste.

Las tareas que desempeñan mejor son tareas de baja carga útil como el ensamblaje y embalaje. Los usos comunes del robot delta son:

- Industria agroalimentaria,
- Industria farmacéutica,
- Industria electrónica.



Figura 1. Robot delta [3].

C. Robot SCARA

Los robots SCARA (figura 2) son una derivación de los antropomórficos con un par de articulaciones que rotan en un mismo plano, mientras que el efector final tiene un desplazamiento en el eje Z. Por este motivo, destacan para tareas de inserción vertical. Son robots que destacan en el mercado por su alta precisión y velocidad, también presentan alguna desventaja como lo es un radio de acción limitado y una baja carga de trabajo útil.

Los robots SCARA son ideales para tareas de ensamblaje y manipulación, y especialmente destacan en la industria electrónica. Usos comunes para el robot SCARA son:

- Tareas de ensamblaje,
- Tareas de manipulación.
- Mesas y bancos ópticos para calibración de instrumentos.



Figura 2. Robot SCARA [8].

D. Robot Antropomórfico

El robot antropomórfico (figura 3) basa su movimiento en los del brazo humano. Compuesto por una serie de articulaciones, cada una le confiere de un grado de libertad, pudiendo tener un total de 4 a 6 grados de libertad. Una de las mejores características de este robot es su flexibilidad, pudiendo alcanzar cualquier punto de su zona de trabajo en cualquier orientación esquivando obstáculos. La principal desventaja que presenta es su alto coste, al igual que su radio de acción limitado. Sin embargo, la ventaja que destaca es por su versatilidad en el desempeño de tareas de distinta índole.

Los usos comunes robot antropomórfico son:

- La soldadura por arco,
- La soldadura por puntos,
- La manipulación de material,
- La alimentación y carga de máquinas,
- La aplicación de pinturas y esmaltes



Figura 3. Robot Antropomórfico [8].

1.2.1. Opciones comerciales

En este apartado se describen las opciones comerciales que, al momento de la redacción de este documento, son las más importantes a nivel mundial.

A nivel mundial, el mercado de la robótica industrial, está dominado en gran medida por empresas japonesas y europeas. ABB, The Yaskawa Electric Corporation, Midea Group (KUKA), The Fanuc Corporation y Kawasaki Heavy Industries son actores líderes en este segmento [9][10][11].

A. ABB

A.1 Scara

El robot de este tipo más barato proporcionado por ABB es el IRB 920 (16.500 € y figura 4) y tiene las siguientes características:

- Alcance: 0,45-0,64 m.
- *Payload*: 6 Kg.
- Grados de libertad: 4.
- Repetibilidad: 0,01 mm.
- Velocidad:
 - Articulación 1: 420°
 - Articulación 2: 701°
 - Articulación Z: 1.1 m/s
 - Articulación 3: 2500°

IRB 920T/IRB 920, working range

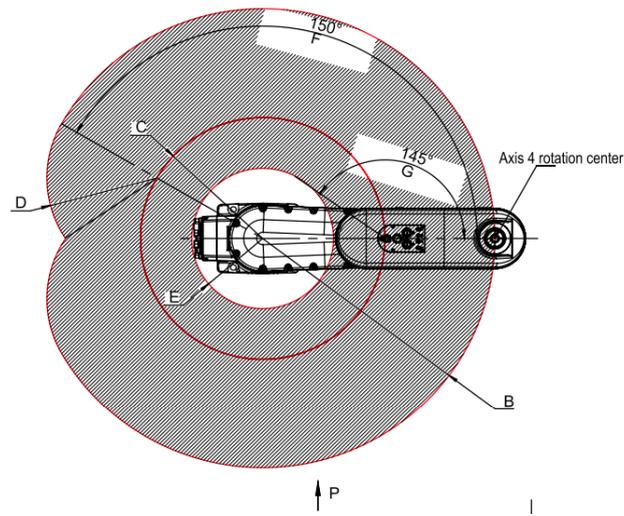


Figura 4. Scara ABB “working space” [9].

A.2. Antropomórfico

El robot de este tipo más barato proporcionado por ABB es el IRB 1100 (21.000€ y figura 5) y tiene las siguientes características:

- Alcance: 0,475-0,58 m.
- *Payload*: 4 Kg.
- Grados de libertad: 6.
- Repetibilidad: 0,01 mm.
- Velocidad:
 - Articulación 1: 460 °/s
 - Articulación 2: 360 °/s
 - Articulación 3: 280 °/s
 - Articulación 4: 560 °/s
 - Articulación 5: 420 °/s
 - Articulación 6: 750 °/s

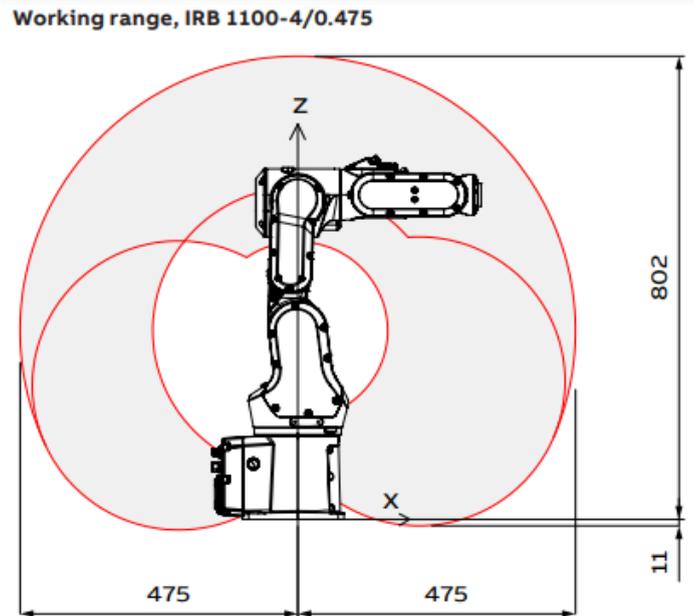


Figura 5. Antropomorfo ABB “working space” [9].

B. The Yaskawa Electric Corporation (YEC)

B.1. Antropomórfico

El robot de este tipo más barato proporcionado por YEC es el MotoMINI (figura 6) y tiene las siguientes características:

- Alcance: 0,350 m.
- *Payload*: 0,5 Kg.
- Grados de libertad: 6.
- Repetibilidad: 0,02 mm.
- Velocidad:
 - Articulación 1: 315 °/s
 - Articulación 2: 315 °/s
 - Articulación 3: 420 °/s
 - Articulación 4: 600 °/s
 - Articulación 5: 600 °/s
 - Articulación 6: 600 °/s



Figura 6. Antropomórfico YEC [10].

B.2. Delta

El robot de este tipo más barato proporcionado por YEC es el MPP3S (figura 7) y tiene las siguientes características:

- Alcance: 0,8 m.
- *Payload*: 3 Kg.
- Grados de libertad: 4.
- Repetibilidad: 0,1 mm.



Figura 7. Delta YEC [10].

B.3. Scara

El robot de este tipo más barato proporcionado por YEC es el SG400 (figura 8) y tiene las siguientes características:

- Alcance: 0,4 m.
- *Payload*: 3 Kg.
- Grados de libertad: 4.
- Repetibilidad: 0,1 mm.
- Velocidad máxima:

- Articulación 1: 740 °/s
- Articulación 2: 800 °/s
- Articulación 3: 1200°/s
- Articulación 4: 3000°/s



Figura 8. Scara YEC [10].

C. KUKA

C.1. Cartesiano

El robot de este tipo más barato proporcionado por KUKA es el KR16L (figura 9) y tiene las siguientes características:

- Alcance: 1500- 500-500 cm.
- *Payload*: 16 Kg.
- Grados de libertad: 3.
- Repetibilidad: 0,1 mm.
- Velocidad máxima:
 - Articulación 1: 3 m/s
 - Articulación 2: 3 m/s
 - Articulación 3: 4 m/s



Figura 9. Cartesiano KUKA [11].

C.2 Delta

El robot de este tipo más barato proporcionado por KUKA es el KR16L (figura 10) y tiene las siguientes características:

- Alcance: 1,200 m.
- *Payload*: 6 Kg.
- Grados de libertad: 4.
- Repetibilidad: 0,05 mm.
- Velocidad máxima:
 - Articulación 1: 600 °/s
 - Articulación 2: 600 °/s
 - Articulación 3: 600 °/s



Figura 10. Delta KUKA [11].

C.3. Scara

El robot de este tipo más barato proporcionado por KUKA es el KR 6 R500 Z200 (figura 11) y tiene las siguientes características:

- Alcance: 0,5 m.
- *Payload*: 6 Kg.
- Grados de libertad: 4.
- Velocidad máxima:
 - Articulación 1: 420°/s
 - Articulación 2: 720 °/s
 - Articulación 3: 2000 °/s
 - Articulación z: 1,1 m/s



Figura 11. Scara KUKA [11].

C.4. Antropomórfico

El robot de este tipo más barato proporcionado por KUKA es el KR 4 AGILUS (figura 12) y tiene las siguientes características:

- Alcance: 0,601 m.
- *Payload*: 4 Kg.
- Grados de libertad: 6.
- Repetibilidad: 0,015 mm.



Figura 12. Antropomorfo KUKA [11].

2. Objetivos

La consigna de este TFG es desarrollar un diseño accesible para el particular de una máquina CNC y dotarla con el software necesario para su puesta en marcha. Para ello se ha desarrollado el trabajo en el siguiente orden:

- Estudio de mercado: se evalúa el estado actual del arte y los robots que predominan hoy en día en la industria.
- Estudio del modelo cinemático: se realiza el estudio y deducen las ecuaciones del robot.
- Realización del diseño: se diseñan las piezas que van a constituir la estructura del robot.
- Elección del sistema de actuación: se analiza y escogen los actuadores según las necesidades del robot.
- Dotación de software: se dota al robot de un software que permita el movimiento de este para trabajos de mecanizado.

2.1. Metodología

Durante el desarrollo de este trabajo, se ha empleado:

- **Fusion 360**: permite el diseño de modelos 3D mediante un software de diseño asistido por ordenador (CAD) fue usado para el diseño de algunas piezas de la CNC.
- **Ultimaker Cura**: programa que permite configurar tu impresora 3D a la hora de imprimir las piezas, fue usado para configurar la impresora antes de imprimir las piezas.
- **Arduino**: plataforma de desarrollo de software y hardware de código abierto.
- **Inkscape**: software de vectores gráficos, en nuestro proyecto, será usado para generar los GCode.
- **Pronterface**: software de código abierto usado para el control de la CNC

3. Modelo cinemático

3.1 Cinemática directa

El modelo matemático conocido como cinemática directa calcula la posición de las diferentes partes de la estructura articulada (en especial, la de su efector final). Esta posición se calcula mediante los componentes fijos (estructura) y los movimientos de los actuadores (articulaciones) [12]. En este modelo se desprecian las fuerzas y momentos generados por el movimiento de la estructura.

Para este trabajo, se deberá calcular la posición del efector final según el giro proporcionado por los motores. Como la estructura del robot es cartesiana (figura 13), esto simplifica mucho el modelo ya que cada eje se mueve de forma independiente.

Para relacionar cada motor con la posición del efector final es necesario conocer el paso de rosca del tornillo sin fin (8mm en nuestro caso).

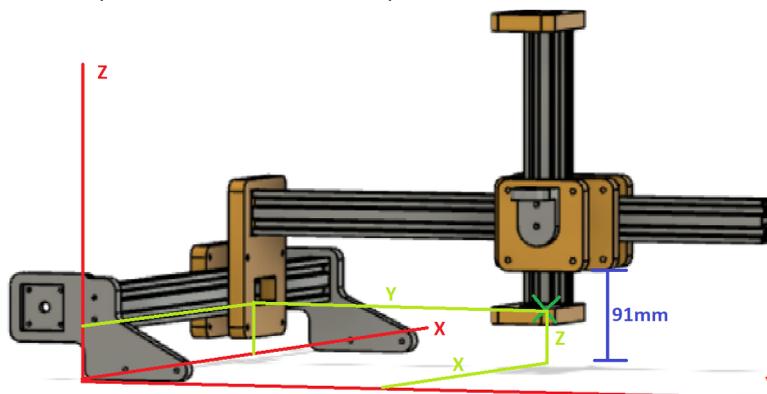


Figura 13. Posición efector final.

Donde h (91 mm en la figura 13) es la altura a la que se encuentra el efector final en la posición de "home". Las ecuaciones de la cinemática directa quedan definidas de la siguiente manera:

- x :
$$x = \frac{8}{360^\circ} * \alpha_x$$
- y :
$$y = \frac{8}{360^\circ} * \alpha_y$$
- z :
$$z = h - \frac{8}{360^\circ} * \alpha_z$$

donde α_x , α_y y α_z indican la posición de las articulaciones, en este caso motores, por lo que estarán en unidades angulares (radianes). El ocho empleado en la ecuación viene determinado por el ángulo de giro de los motores en cada paso.

3.2. Cinemática Inversa

El modelo matemático de la cinemática inversa calcula las posiciones de los ejes de los actuadores (motores) del robot a partir de la posición del efector final y la estructura de la máquina. La utilidad de este modelo es lograr que el efector final se ubique en una posición determinada [13].

Para un robot cartesiano, al depender la posición de cada eje únicamente de un actuador, es posible determinar la cinemática inversa despejando de la cinemática directa.

- $\alpha_x :$ $\alpha_x = \frac{360^\circ * x}{8}$
- $\alpha_y :$ $\alpha_y = \frac{360^\circ * y}{8}$
- $\alpha_z :$ $\alpha_z = \frac{360^\circ * (h-z)}{8}$

4. Propuesta de diseño

Todos los diseños de las piezas han sido diseñados en el software Fusion 360 para su impresión en 3D. La impresora utilizada en la fabricación del robot ha sido la “LulzBot TAZ 6” (figura 14) [15].



Figura 14. Impresora “LulzBot TAZ 6” [15].

Las características de esta impresora son:

- *Tool Head: TAZ Single Extruder Tool Head v2.1 0.50 mm nozzle*
- *Layer Resolution 0.05 mm - 0.4 mm (0.002in - 0.02in)*
- *Max Hot End Temperature 290° C*
- *Level Automatic Z-Axis Compensation*
- *Certifications FCC, CE, WEEE, OSHWA, FSF-RYF*
- 280 mm x 280 mm x 250mm

Las piezas han sido impresas en PLA (ácido poliláctico). Este material de impresión 3D presenta algunas ventajas: es un material biodegradable y no tóxico, el PLA actualmente es material más fácil para imprimir y se adhiere bien a las plataformas de impresión. El PLA tiene una excelente calidad de impresión lo que lo convierte en una opción ideal. Es una opción ecológica y sostenible porque está hecho con recursos renovables como el maíz [16].

4.1. Software Ultimaker Cura

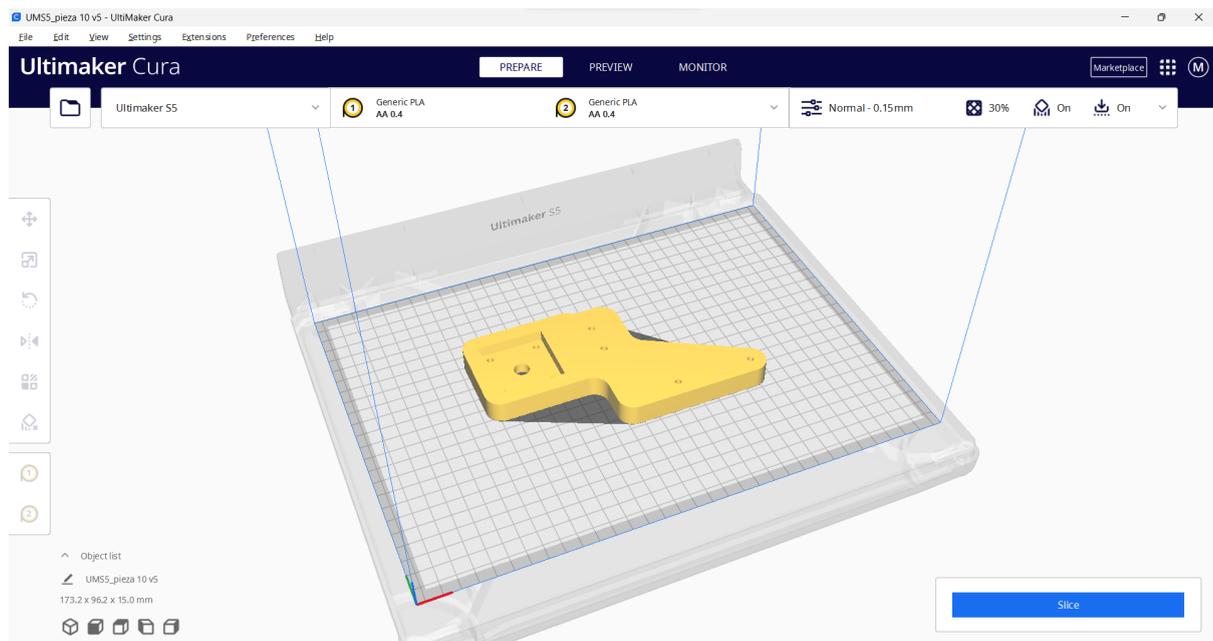


Figura 15. Software Ultimake Cura.

En este proyecto, para la configuración de la impresora 3D se ha decidido emplear el software Ultimaker Cura (figura 15). La principal ventaja que presenta este software frente a otros es que es de código abierto. Otra gran ventaja es que permite una gran flexibilidad en el proceso de imprimir, ya que los usuarios pueden configurar una gran cantidad de parámetros en la impresión para satisfacer sus requisitos. Estos ajustes en los parámetros son:

- Temperatura *hotend*: temperatura del extrusor (200 °C para PLA)
- Temperatura cama impresión: temperatura de la cama de la impresora, el PLA no requiere temperatura en la cama pero es recomendable añadirla para que se adhiera mejor (50°C para PLA)
- Velocidad: velocidad de desplazamiento del extrusor (60 mm/s para PLA).
- Altura de capas: grosor de la capa.
- Retracción: invierte el motor del extrusor para evitar la presión en el cabezal (6mm a 25mm/s para PLA).
- Relleno: cantidad de material depositado en el patrón interior (*lines 25%*) (figura 16 y 17).
- Capa inicial: parámetros para la capa inicial para facilitar que se adhiera a la cama caliente, recomendable reducir la velocidad de impresión en las primeras capas y apagar el ventilador (figura 18).

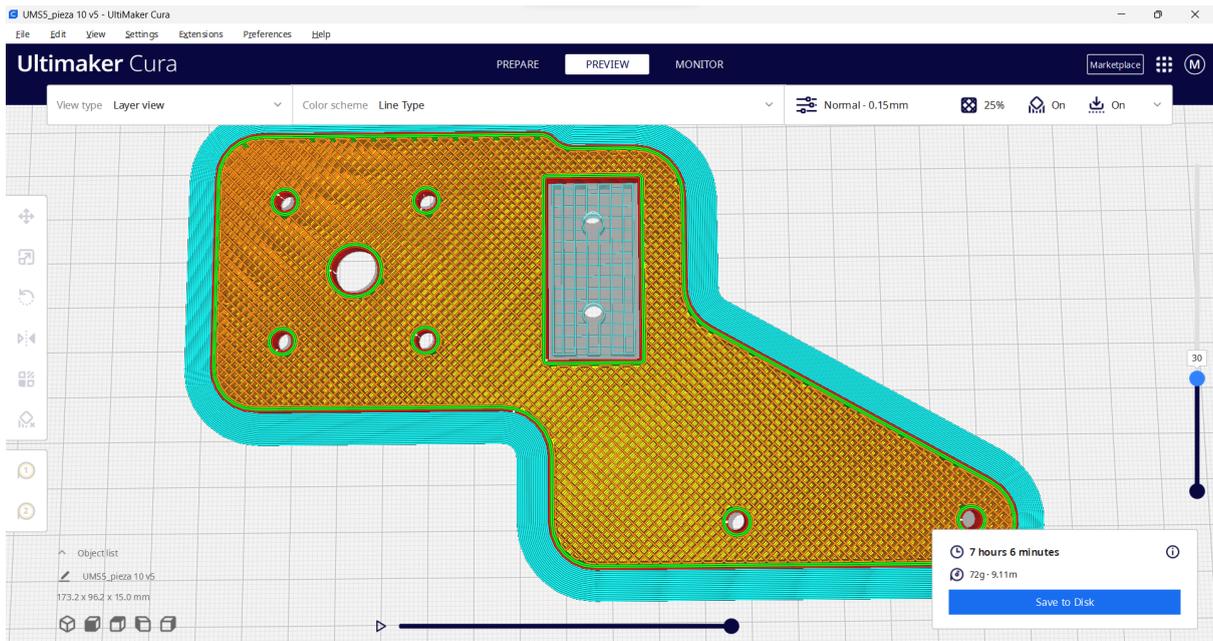


Figura 16. Previsualización de la impresión con las especificaciones.

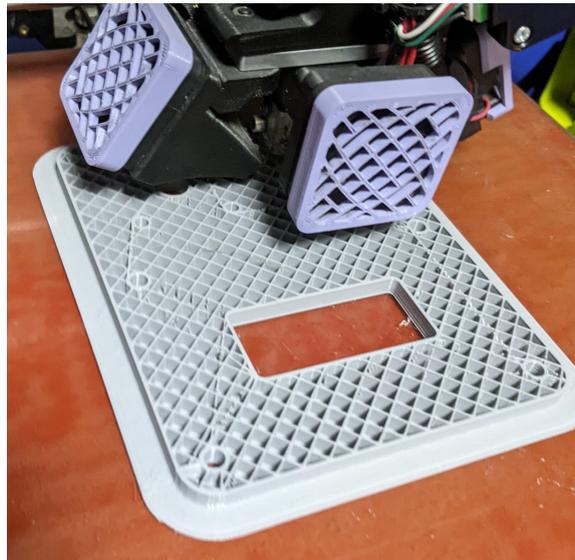


Figura 17. Imagen de impresión 1.

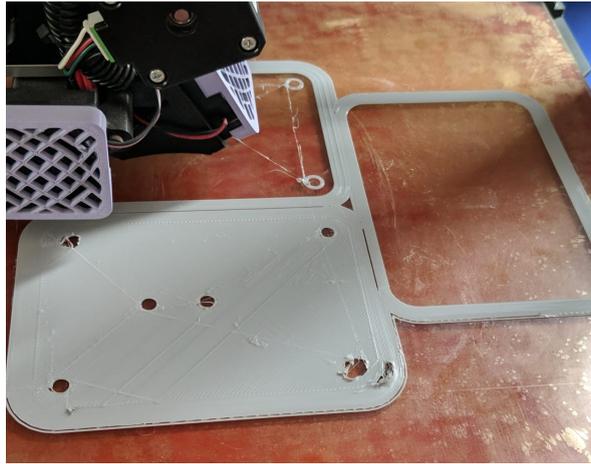


Figura 18. Imagen de impresión 2.

4.2 Diseño de la estructura

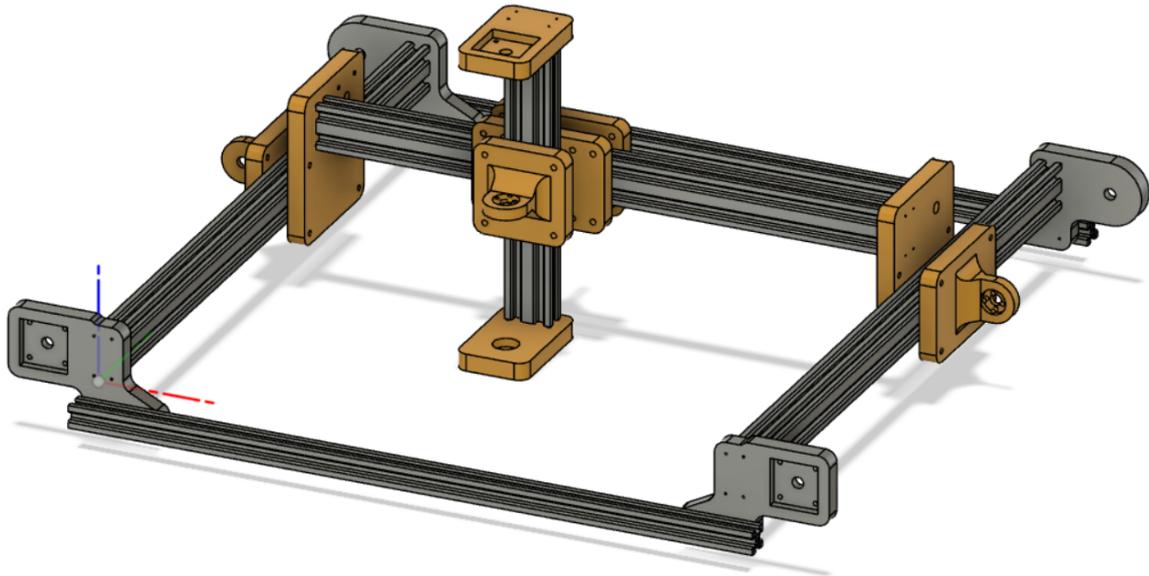


Figura 19. Diseño basado en tornillo sin fin.

Según se puede observar en la figura 19, las piezas utilizadas para el diseño de la estructura de máquina CNC son las siguientes:

Estas piezas son empleadas para el montaje de la estructura en su mayoría siendo solo los tornillos sin fin, los motores nema y los adaptadores montaje-motor piezas pertenecientes a los actuadores que dotan de movimiento a la máquina.

- 1x Anclaje motor eje Z (71 g).
- 1x Sujeción tornillo sin fin eje Z (75 g).
- 1x Soporte la fijación tuerca tornillo sin fin eje Z (68 g).
- 1x Soporte perfil aluminio eje Z (114 g).

- 1x Soporte de la fijación tuerca tornillo sin fin eje Y (100 g).
- 1x Soporte motor eje Y (180 g).
- 1x Sujeción tornillo sin fin eje Y (185 g).
- 2x Soporte de la fijación tuerca tornillo sin fin eje X (100g).
- 2x Soporte motor eje X (80g).
- 2x Sujeción tornillo sin fin eje X (80g).
- 4x Fijación tornillo sin fin. (28g)
- 4x Perfil aluminio V-Slot 2040
- 2x Perfil aluminio V-Slot 2020
- 16x Ruedas para perfiles V-slot
- 4x Tornillos sin fin + tuerca
- 4x Motores Nema17
- 4x Adaptadores tornillo-motor
- Tornillería variada.

4.3. Perfiles de la base del robot

La estructura principal del robot serán 6 perfiles V-Slot que se van a dividir en dos grupos: uno encargado de dar soporte únicamente formado por dos perfiles V-Slot 2020 (figura 21) y un segundo grupo encargado de permitir el desplazamiento de la estructura móvil, este último formado por 4 perfiles V-Slot 2040 (figura 20).

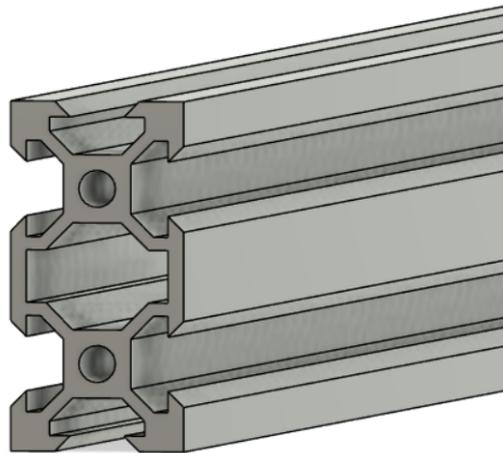


Figura 20. Perfil V-Slot 2040.

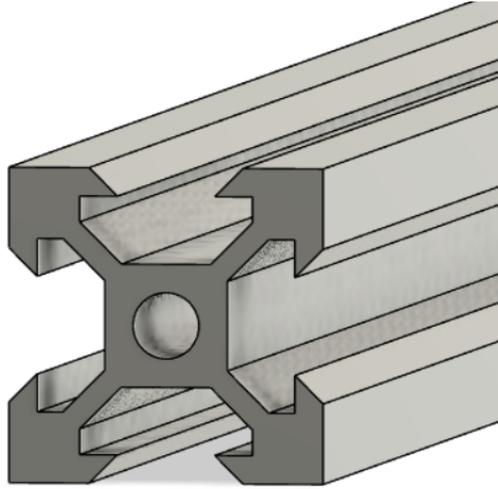


Figura 21. Perfil V-Slot 2020.

Los perfiles del eje X son los encargados de soportar toda la estructura móvil del robot cartesiano y serán dos perfiles 2040 colocados paralelamente entre ellos para permitir el desplazamiento lineal sobre estos.

El eje Y estará constituido por un perfil 2040 y estará situado de forma perpendicular a los anteriores, este permitirá el desplazamiento lineal y soportará el resto de la estructura móvil.

El último perfil 2040 es el del eje Z y estará dispuesto perpendicularmente a todos los anteriores y será el encargado de permitir el desplazamiento lineal en su eje.

Los perfiles 2020 están colocados perpendicularmente al eje X y su función es la de mantener paralelos los perfiles 2040 del eje X así como dotar de solidez a la estructura.

Todos los perfiles estarán unidos entre sí mediante las piezas diseñadas en el software fusion 360 en este proyecto y ruedas con rodamiento específicas de este tipo de perfiles.

4.4. Tornillo sin fin y tuerca

El movimiento de esta estructura se basa en un tornillo de anillo (figura 23), que es girado por un motor, creando una fuerza sobre la tuerca (figura 22); la tuerca se unirá a la estructura móvil.



Figura 22. Tuerca tornillo sin fin.



Figura 23. Tornillo sin fin.

La velocidad y la potencia del robot depende en gran medida del radio y el paso de rosca del tornillo y la tuerca, por lo que hay que tener en cuenta estos valores. El paso también es necesario para calcular la cinemática inversa y directa del robot.

4.5. Anclaje motor eje Z

Este es el soporte de motor más pequeño (figura 24), ya que solo se acopla al perfil de aluminio V-Slot 2040.

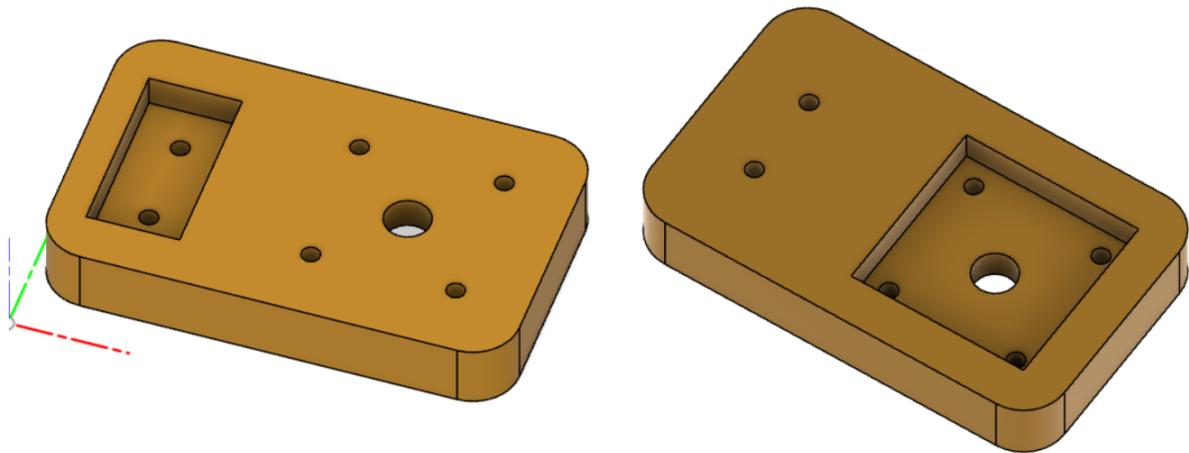


Figura 24. Anclaje motor eje Z.

La pieza fue diseñada para unir el motor al perfil de aluminio V-Slot 2040 usando dos tornillos de 5 mm de diámetro y dos tornillos de 3 mm de diámetro para mantener el motor en su lugar. La hendidura de la pieza se adapta a las dimensiones del perfil y del motor, lo que asegura su conexión. La pieza tiene un grosor de 15 mm y hendiduras de 7,5 mm de profundidad. De izquierda a derecha, ambas hendiduras se muestran en la figura 19.

4.6. Sujeción tornillo sin fin eje Z

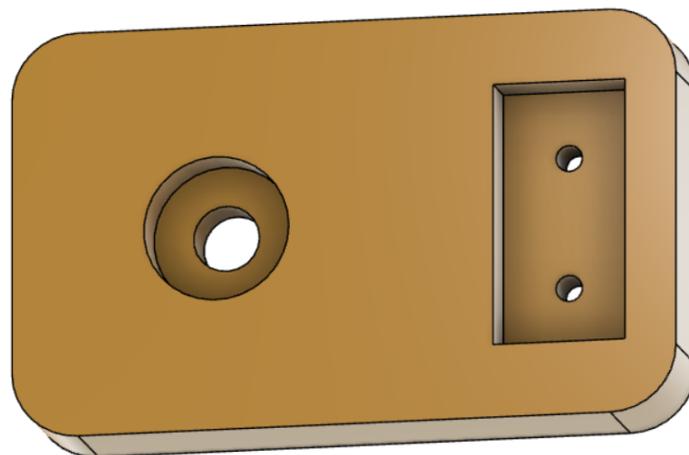


Figura 25. Sujeción tornillo sin fin eje Z.

La pieza (figura 25) está diseñada para sujetar el tornillo en su posición mientras solo permite que gire. Para lograrlo, se perfora un orificio de 22 mm en el que se insertará el rodamiento para sujetar el tornillo sin fin concéntricamente a esta perforación. Además, se realiza un agujero de 10 mm para el paso del tornillo.

Cabe señalar que esta pieza es el actuador final del robot, por lo que su diseño dependerá de su uso. Esto significa agregar los elementos necesarios a la pieza para acomodar las herramientas que necesita.

4.7. Soporte tuerca de tornillo sin fin eje Z

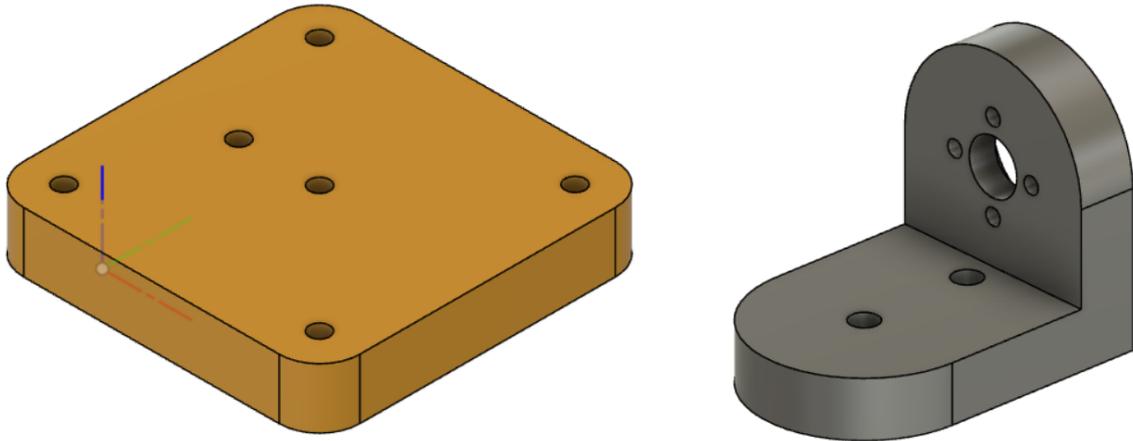


Figura 26 y 27. Soporte tuerca de tornillo sin fin eje Z.

Debido a las limitaciones de la impresora 3D, el trabajo se divide en dos partes (figura 26 y 27). La parte izquierda tiene cuatro agujeros de 5 mm de diámetro para colocar las ruedas para que el perfil pueda moverse, y dos agujeros de 5 mm para unir esta parte a la otra parte de la parte principal mediante dos tornillos de 5 de diámetro. Por otro lado, los orificios restantes en el lado derecho son para sujetar la tuerca del tornillo sin fin mediante tornillos de 3 mm de diámetro.

Vale la pena mencionar que dividir la pieza en dos tenía la intención de adaptarse a las limitaciones de la impresora 3D, lo que permite una impresión más eficiente y precisa.

4.8. Soporte perfil aluminio eje Z

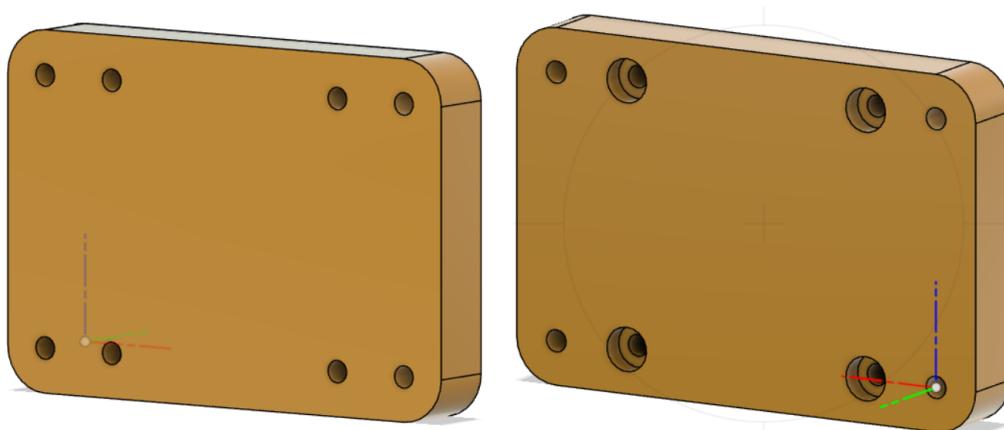


Figura 28. Soporte perfil aluminio eje Z.

La función de esta pieza (figura 28) es unir los perfiles de aluminio de los ejes Y y Z. Cada perfil requiere 4 tornillos, haciendo un total de 8 agujeros con un diámetro de 5 mm.

Es importante tener en cuenta que las 4 hendiduras, que se ven en los 4 agujeros de la parte derecha de la figura 23, se incluyen porque la pieza está intercalada entre dos perfiles y una de sus superficies no tiene suficiente espacio entre la pieza y el perfil para la cabeza del tornillo. La inclusión de estas hendiduras proporciona el espacio necesario para que las cabezas de los tornillos no rocen con el perfil. Vale la pena mencionar que la precisión y calidad de esta parte de la construcción asegura una conexión fuerte y estable entre los perfiles del eje Y y el eje Z, lo que contribuirá a la estabilidad y precisión de los movimientos del robot.

4.9. Soporte tuerca tornillo sin fin ejes X e Y

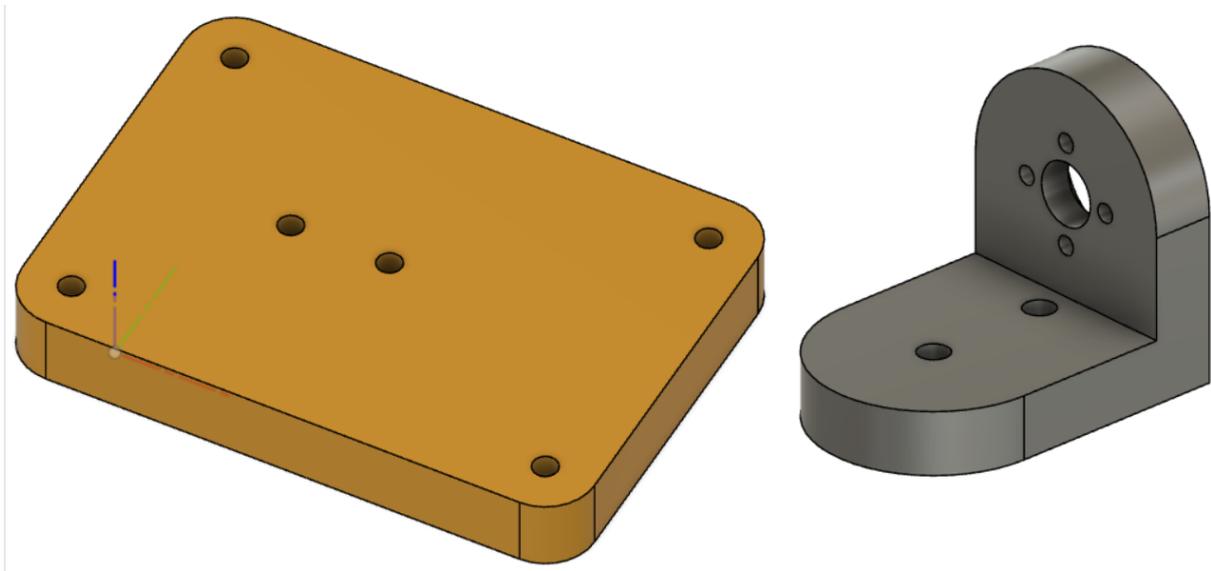


Figura 29 y 30. Soporte tuerca tornillo sin fin ejes X e Y.

Esta es una pieza (figura 29 y 30) casi idéntica al soporte eje Z (figura 26 y 27); la única diferencia es que es 31,2 mm más larga que el soporte del eje Z porque, si fueran del mismo tamaño, los agujeros de los tornillos de la pieza anterior se superpondrían impidiendo así el montaje de la estructura.

Esta pieza tiene la misma función que la pieza de la figura 21, es decir, transferir la fuerza que produce el tornillo sin fin a la estructura, provocando su movimiento. En este caso, esta parte se usa tres veces: una para el movimiento del eje Y y dos veces para el movimiento del eje X.

4.10. Soporte motor eje Y

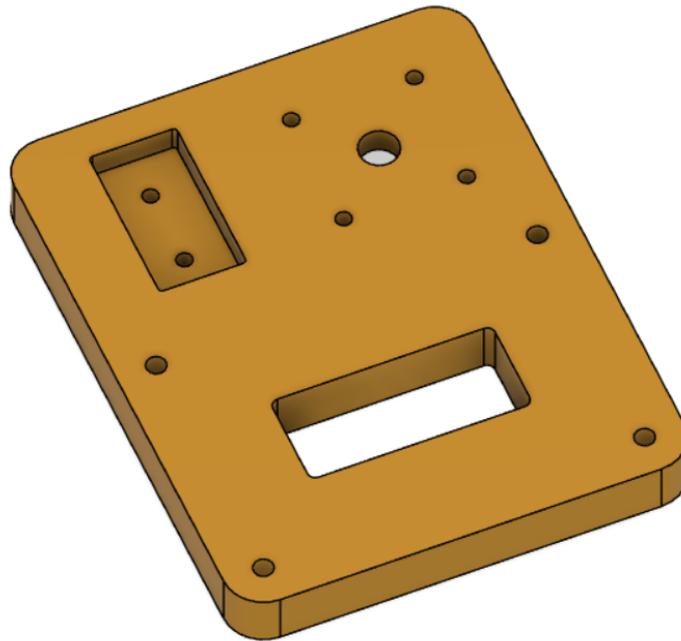


Figura 31. Soporte motor eje Y.

Esta pieza (figura 31) desempeña un papel importante en la estructura del robot cartesiano porque es la encargada de conectar el perfil del eje Y a uno de los ejes X. A diferencia de la conexión entre los perfiles del eje Z y del eje Y, el perfil del eje Y no se mueve en relación con la pieza por lo que es muy importante que la conexión sea fuerte.

Esta parte también funciona como un soporte de motor del eje Y; ha sido diseñada con una hendidura diseñada específicamente para el tamaño del motor y cuatro orificios de 3 mm de diámetro para los tornillos de montaje. También dispone de otros dos agujeros de 5 mm de diámetro para sujetar los perfiles.

Cabe señalar que la presencia de un orificio cuadrado en el centro de la pieza está diseñada para reducir la cantidad de material necesario en el proceso de impresión sin comprometer la funcionalidad de la pieza.

4.11. Soporte tornillo sin fin eje Y

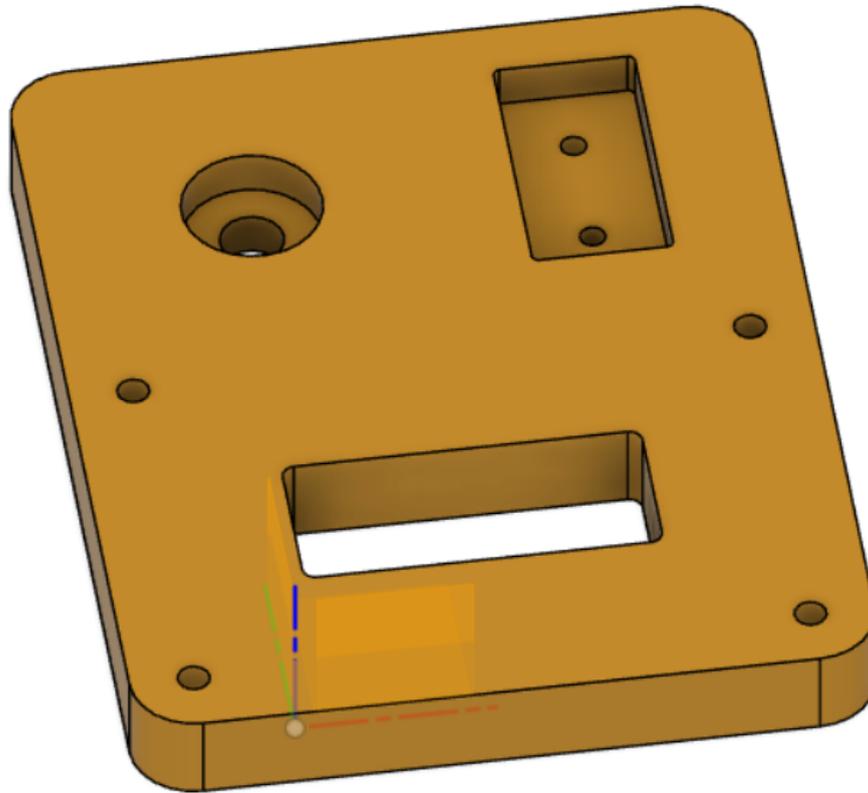


Figura 32. Soporte tornillo sin fin eje Y.

La siguiente pieza (figura 32) que se ha diseñado es de vital importancia para el correcto funcionamiento de la CNC. Aunque su función es similar a la de la pieza anterior, esta pieza tiene la tarea de servir como soporte para el tornillo sin fin en su otro extremo. Al estar diseñada para soportar el movimiento del eje Y, se requiere que la unión entre los perfiles sea altamente resistente y capaz de soportar cargas considerables. Por esta razón, el diseño de esta pieza es crucial para garantizar la estabilidad del eje Y.

Esta pieza se conecta a dos perfiles de aluminio: uno que forma parte del eje Y y otro que forma parte del eje X, uniendo ambos perfiles y permitiendo el movimiento de la plataforma de trabajo en dos direcciones. Para reducir el rozamiento del robot y asegurar un movimiento suave y sin problemas, se ha previsto un hueco en esta pieza para alojar un rodamiento que soportará el giro del tornillo sin fin. De esta manera, se reducirá el esfuerzo necesario para mover la plataforma de trabajo y se mejorará la precisión del movimiento.

El diseño de esta pieza también contempla la instalación de cuatro agujeros de 5 mm de diámetro para fijar los perfiles y otros dos agujeros de 5 mm de diámetro para fijar la pieza al perfil del eje X. Además, se ha añadido un agujero cuadrado en el centro para reducir el material necesario durante la impresión en 3D.

4.12. Soporte motor eje X

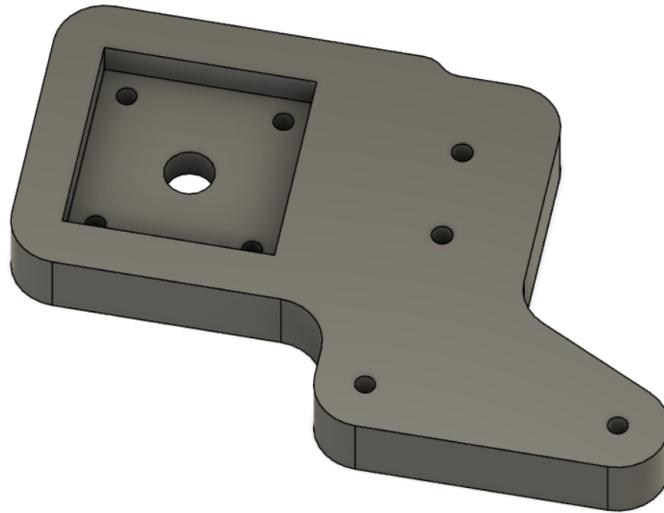


Figura 33. Soporte motor eje X.

Esta pieza (figura 33) es esencial para conectar uno de los perfiles del eje X con uno de los perfiles 2020 en la estructura del robot. Mientras que la unión con el perfil 2040 sigue el mismo proceso que las demás, utilizando dos tornillos de 4 mm de diámetro y una hendidura con las dimensiones del perfil, la unión con el perfil 2020 se realiza por el lateral. Por lo tanto, se necesitarán dos tornillos de menor diámetro (2 mm) que se unirán a una pieza situada en la ranura del perfil.

La fijación del motor sigue el mismo proceso que las otras piezas que sostienen los motores. Se utiliza una hendidura con las dimensiones del motor y se sujetan con cuatro tornillos de 3 mm.

Se requieren dos piezas de este tipo, pero son simétricas entre sí para garantizar una unión estable y equilibrada entre los perfiles del eje X y los perfiles 2020 de la estructura del robot.

4.13. Soporte tornillo sin fin eje X

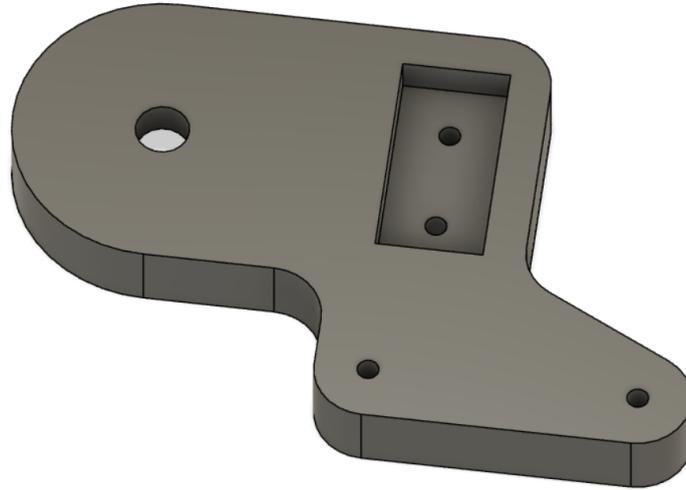


Figura 34. Soporte tornillo sin fin eje X.

Esta pieza (figura 34) realiza la función de unión de perfiles al igual que la pieza anterior entre perfiles 2020 y 2040, a diferencia de la anterior, en vez de sujetar el motor, sujetara el extremo del tornillo sin fin mediante un rodamiento como las piezas con la misma función enseñadas anteriormente.

5. Sistema de actuación

La premisa fundamental a cumplir en esta parte del diseño de la máquina CNC es que la elección de los motores del sistema de actuación asegure que se puedan mover las masas móviles en todos los ejes. En particular, en este caso el eje Z es el más comprometido, ya que debe vencer tanto el peso de la masa móvil como su inercia. Por otro lado, los ejes X e Y, al estar apoyados sobre raíles, solo tienen que vencer la fuerza de inercia.

Es importante tener en cuenta que el motor elegido debe ser lo suficientemente potente para mover la carga máxima a la velocidad requerida, manteniendo al mismo tiempo la precisión necesaria. Además, es recomendable incluir un margen de seguridad en la elección del motor para garantizar su correcto funcionamiento en condiciones extremas. En definitiva, la elección del motor adecuado es un factor crítico para el éxito del diseño, ya que de ello dependerá la eficiencia y fiabilidad del sistema.

5.1. Dimensionamiento motor eje Z

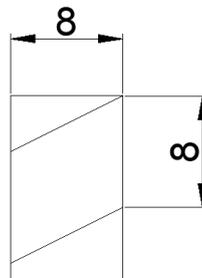


Figura 35. Dimensiones tornillo sin fin (mm).

Primero se calcula el perímetro del tornillo en función de su radio (r). En la figura 35 se ve un esquema de las dimensiones del tornillo, del cual se sacan los datos que se usan a continuación en este caso se trata de un tornillo de dimensiones 8 mm tanto en el diámetro como en el paso de rosca..

$$\begin{aligned} \text{Perímetro} &= 2 * \pi * r \\ \text{Perímetro} &= 2 * \pi * 4 = 25.13 \text{ mm} \end{aligned}$$

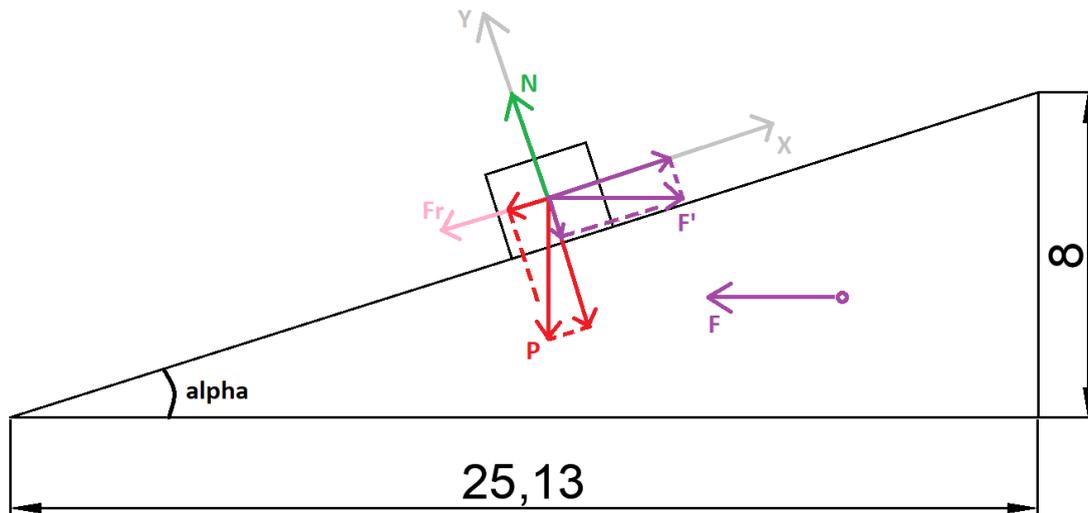


Figura 36. Diagrama sólido rígido del tornillo sin fin del eje Z.

En segundo lugar se obtienen el ángulo de rosca del tornillo (alpha en la figura 36):

$$\alpha = \text{atan}\left(\frac{8}{25.13}\right) = 0.3081 \text{ rad}$$

Se debe calcular la fuerza necesaria (F) que deberá aplicar el motor a la estructura para lograr mover la estructura, para ello se hace un estudio para el caso más crítico (desplazamiento en contra de la gravedad). Las fuerzas que se verán implicadas en el sistema son:

$$P = \text{Peso}$$

$$N = \text{Normal}$$

$$Fr = \text{Fuerza rozamiento}$$

$$F = \text{Fuerza aplicada por el motor}$$

$$F' = \text{Reacción de la estructura a la fuerza aplicada por el motor}$$

En donde la fuerza aplicada por el motor dependerá del momento que genere este y del radio del tornillo sin fin.

$$M = \text{momento aplicado por el motor}$$

$$F = \frac{M}{r}$$

Dado que se quiere dimensionar el motor, se cogen los datos del fabricante y se calcula la masa máxima móvil que puede tener este en el eje Z (figura 37).

Specifications

1.5 Amp motors		Single length	Double length	Triple length
Part number		M-1713-1.5 • (1)	M-1715-1.5 • (1)	M-1719-1.5 • (1)
Holding torque	oz-in	32	60	75
	N-cm	23	42	53
Detent torque	oz-in	1.7	2.1	3.5
	N-cm	1.2	1.5	2.5
Rotor inertia	oz-in-sec ²	0.000538	0.0008037	0.0011562
	kg-cm ²	0.038	0.057	0.082
Weight	oz	7.4	8.1	12.7
	grams	210	230	360
Phase current	amps	1.5	1.5	1.5
Phase resistance	ohms	1.3	2.1	2.0
Phase inductance	mH	2.1	5.0	3.85

(1) Indicate S for single-shaft or D for double-shaft. Example M-1713-1.5S

Figura 37. Datos del Fabricante Motores Nema17.

El primer estudio es para unas condiciones ideales en donde no hay rozamiento:

Mediante la segunda ley de Newton se estipula que la fuerza generada por la tuerca sobre la estructura que mueve sumada a la fuerza del peso de la estructura deberá de ser nula si el robot está estático, mientras que para producir movimiento, el peso deberá de ser menor que la fuerza resultante aplicada por los motores.

$$0 = P_x + F'_x$$

$$0 = F' * \cos(\alpha) - P * \sin(\alpha)$$

$$P < \frac{F'}{\tan(\alpha)}$$

$$P_{\text{máx}} = \frac{M}{r * \tan(\alpha)}$$

Con rozamiento en el sistema, este se vuelve más complicado pues en el eje x aparece la fuerza de rozamiento que dependerá de la fuerza normal:

$$N = P_y + F'_y$$

$$N = P * \cos(\alpha) + F' * \sin(\alpha)$$

$$Fr = \mu * N$$

Una vez obtenida la fuerza de rozamiento la forma de proceder será la misma que en el caso de sin rozamiento:

$$0 = P_x + F'_x + Fr$$

$$0 = F' * \cos(\alpha) - P * \sin(\alpha) - \mu(P * \cos(\alpha) + F' * \sin(\alpha))$$

$$0 = F' \cos(\alpha) - P * \sin(\alpha) - \mu * P * \cos(\alpha) - \mu * F' * \sin(\alpha)$$

$$0 = F'[\cos(\alpha) - \mu * \sin(\alpha)] - P[\sin(\alpha) + \mu * \cos(\alpha)]$$

$$P < \frac{F'[\cos(\alpha) - \mu * \sin(\alpha)]}{\sin(\alpha) + \mu * \cos(\alpha)}$$

$$P_{\text{máx}} = \frac{M * [\cos(\alpha) - \mu * \sin(\alpha)]}{r * [\sin(\alpha) + \mu * \cos(\alpha)]}$$

Datos	Momento motores (Ncm)	Pesos máximos (N)		Masas máximas (Kg)		
		Sin rozamiento	Con rozamiento	Sin rozamiento	Con rozamiento	
Radio (cm)	0,4	23	180,64	59,08	18,41	6,02
Paso de rosca (cm)	0,8	42	329,87	107,89	33,63	11,00
Ángulo (rad)	0,308	53	416,26	136,15	42,43	13,88
Coefficiente de rozamiento	0,5					

Tabla 1. Cálculo de masas máximas.

Este eje (Z) será el que determine el dimensionamiento de los motores nema17, pues es el crítico. Como se puede ver en la tabla 3, con motores Nema capaces de generar 23 Ncm ya estamos obteniendo masas móviles de 6 Kg.

5.2. Cálculo de velocidad máxima de los motores

Por lo general un motor paso a paso posee una velocidad angular máxima de 1000 rpm. Pero esta también se verá determinada por el driver del motor [14].

Si se quiere obtener la velocidad de cada eje de manera lineal, se deberá tener en cuenta el paso de rosca de los tornillos sin fin:

$$V = \text{velocidad lineal}$$

$$w = \text{velocidad angular}$$

$$d = \text{paso de rosca}$$

El paso de rosca es un valor fijo determinado por el tornillo sin fin calculado en el apartado anterior (Figura 36).

$$V = w * d$$

$$V = 1000 \text{ rpm} * 0,8 \text{ cm}$$

$$V = 800 \text{ cm/min} = 13.333 \text{ cm/s}$$

Para obtener la velocidad del efector final hay que sumar las velocidades de cada eje de forma vectorial:

$$V_{\text{max}} = \sqrt{V_x^2 + V_y^2 + V_z^2}$$

En este caso los tres tornillos sin fin tienen el mismo paso de rosca de 8 mm por lo que las velocidades en cada eje son iguales.

$$V_{\text{max}} = \sqrt{V^2 + V^2 + V^2}$$

$$V_{\text{max}} = \sqrt{3V^2}$$

$$V_{\text{max}} = \sqrt{3} * V$$

$$V_{\text{max}} = \sqrt{3} * 13.33 \text{ cm/s} = 23.09 \text{ cm/s}$$

6. Montaje y electrónica.

En este apartado se explicará los puntos claves del montaje y de la dotación de la electrónica, en la electrónica se tratarán los temas como la conexión de los finales de carrera, la conexión de la placa madre arduino con la *ramps* 1.4 y la conexión de los drivers con la *ramps* 1.4.

6.1. Montaje.

Una vez se dispone de todas las piezas, se procederá con el montaje de la CNC siendo los puntos clave de este los siguientes:

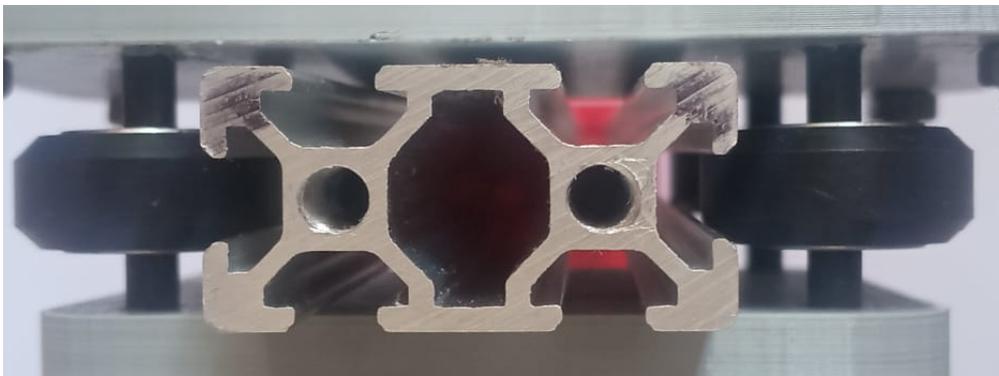


Figura 38. Ruedas apoyadas sobre los perfiles.

Las ruedas deben estar situadas en el punto medio de los tornillos (figura 38) y acomodarse sobre las hendiduras de los perfiles, de esta forma el perfil quedará perfectamente centrado entre las dos piezas permitiendo un desplazamiento sin rozamiento.



Figura 39. Rodamientos encajados en los soportes.

Las hendiduras de las piezas (figura 39) que albergan los rodamientos han sido diseñadas con un diámetro de iguales dimensiones al del rodamiento. De esta forma queda sujeto sin necesidad de otra pieza o algún tipo de fijador.



Figura 40. Motores encajados en los soportes.

Tal y como se indicó en el capítulo 7, las piezas que albergan a los motores poseen una hendidura de dimensiones semejantes a los motores (figura 40). Una vez el motor esté situado en la hendidura, se fija la sujeción con los tornillos correspondientes (M3).



Figura 41. Creación de rosca en los perfiles.

El roscado de los perfiles (figura 41) es un paso obligatorio, pues la rosca que se realice en este proceso será la que sujete los perfiles con las piezas encargadas de soportar los perfiles. Esta unión pieza-perfil debe ser sólida por lo que la rosca debe ser de las dimensiones correctas (m5).

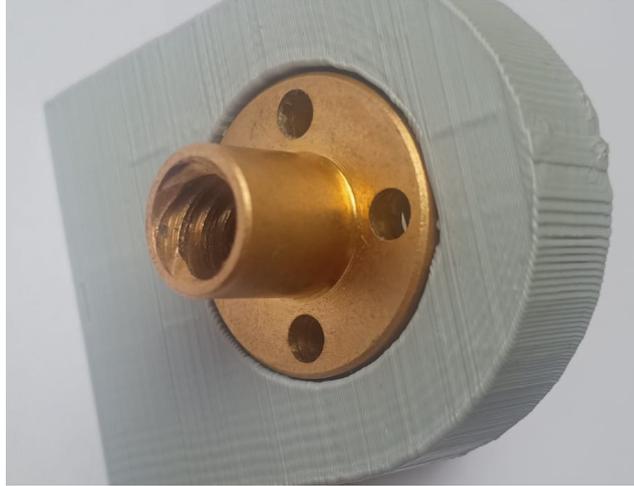


Figura 42. Tuerca del tornillo sin fin encajada en los soportes.

Al igual que los rodamientos (figura 39), las dimensiones de la hendidura que albergará la tuerca tiene la mismas dimensiones que ésta, por lo que quedará sujeta sin necesidad de fijadores o piezas añadidas (figura 42). Si la máquina CNC va a trabajar con cargas elevadas, la pieza posee 4 agujeros para agregar tornillos m4 y dotar de más solidez a la unión.



Figura 43. Unión tornillo sin fin con los motores.

Esta unión se realiza mediante una pieza comercial (figura 43) dicha pieza deberá de ser capaz de soportar y transmitir las fuerzas generadas por los motores, diseñar una posible pieza que reemplace la comercial no es viable porque la pieza comercial consta de una

espiral (véase en la figura 43) que permite transmitir dicha fuerza incluso cuando el eje del motor y el tornillo sin fin no están perfectamente alineados.

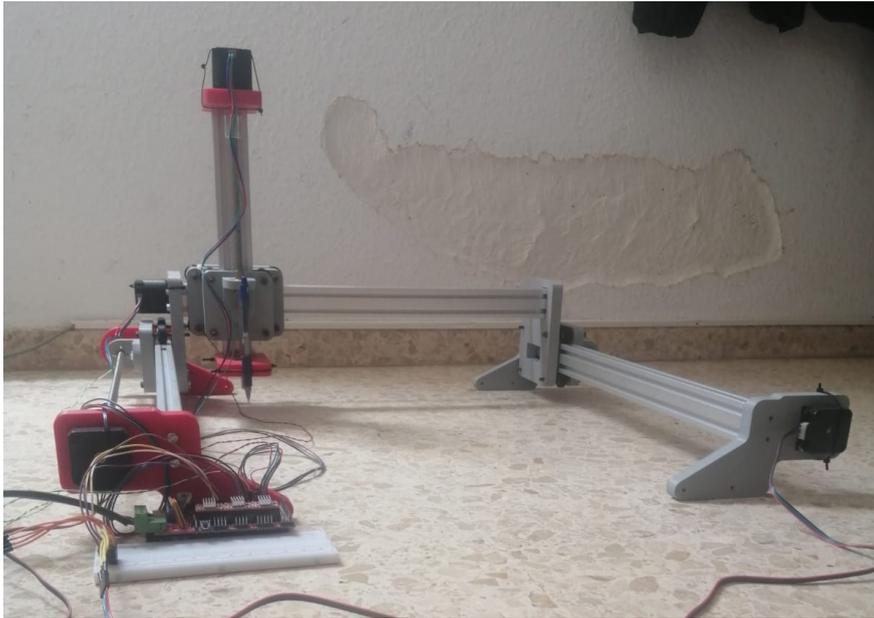


Figura 44. Montaje completo visto desde el eje X.

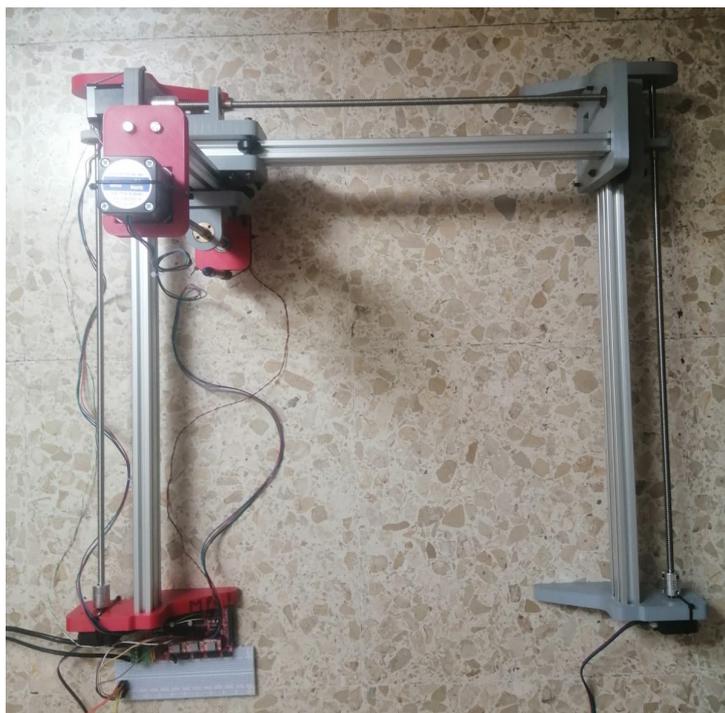


Figura 45. Montaje completo visto desde arriba.

Con el montaje terminado se tendrá una CNC como se muestra en las figuras 44 y 45.

6.2. Electrónica.

El control de la máquina CNC diseñada en este trabajo se realiza mediante un Arduino Mega 2560, una *ramps* 1.4, drivers A4988 y finales de carrera. La tarea de control es una parte fundamental para el correcto funcionamiento de la CNC, para lograr que ejecute trabajos de precisión de forma eficiente.

La elección de estos componentes fue en gran medida por su disponibilidad en el mercado así como su precio asequible. Como añadido se buscaron componentes que tuvieran una alta compatibilidad y que fueran modulares, permitiendo una fácil instalación.

6.2.1 Arduino Mega 2560

El Arduino Mega 2560 (figura 46) es un dispositivo electrónico de código abierto. Está basado en el microcontrolador ATmega2560. Es una placa compatible con una gran variedad de módulos [17]. En este trabajo se emplea el módulo Ramps 1.4, que es un módulo específico para el control de motores paso a paso en una máquina CNC.

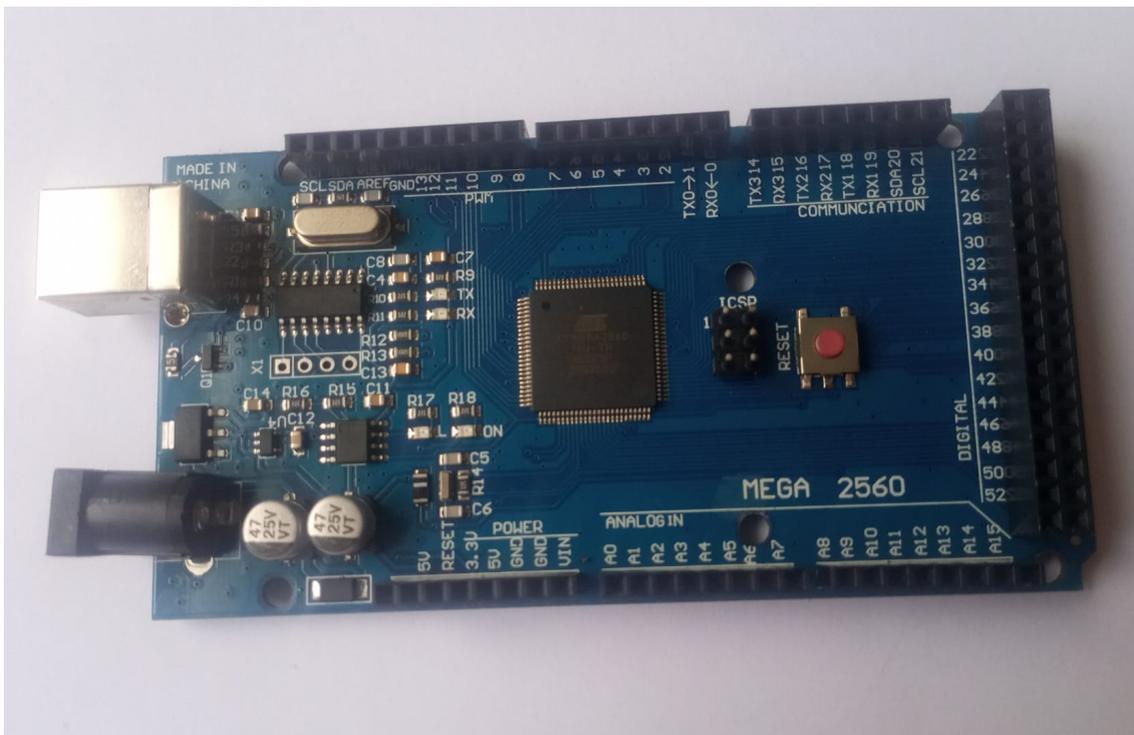


Figura 46. Arduino mega 2560.

6.2.2 Ramps 1.4

RAMPS (figura 47) es la abreviatura de “RepRap Arduino Mega Pololu Shield”, y tiene la capacidad de controlar hasta 5 motores paso a paso: 3 para el movimiento de los ejes X, Y y Z, además 2 motores adicionales para el control de la/s extrusora/s en caso de que se use para una impresora 3D con una extrusora o varias. La ventaja que presenta este módulo

frente a los demás, es la capacidad de implementar el control de la herramienta en este mismo módulo sin necesidad de añadir más hardware [18].

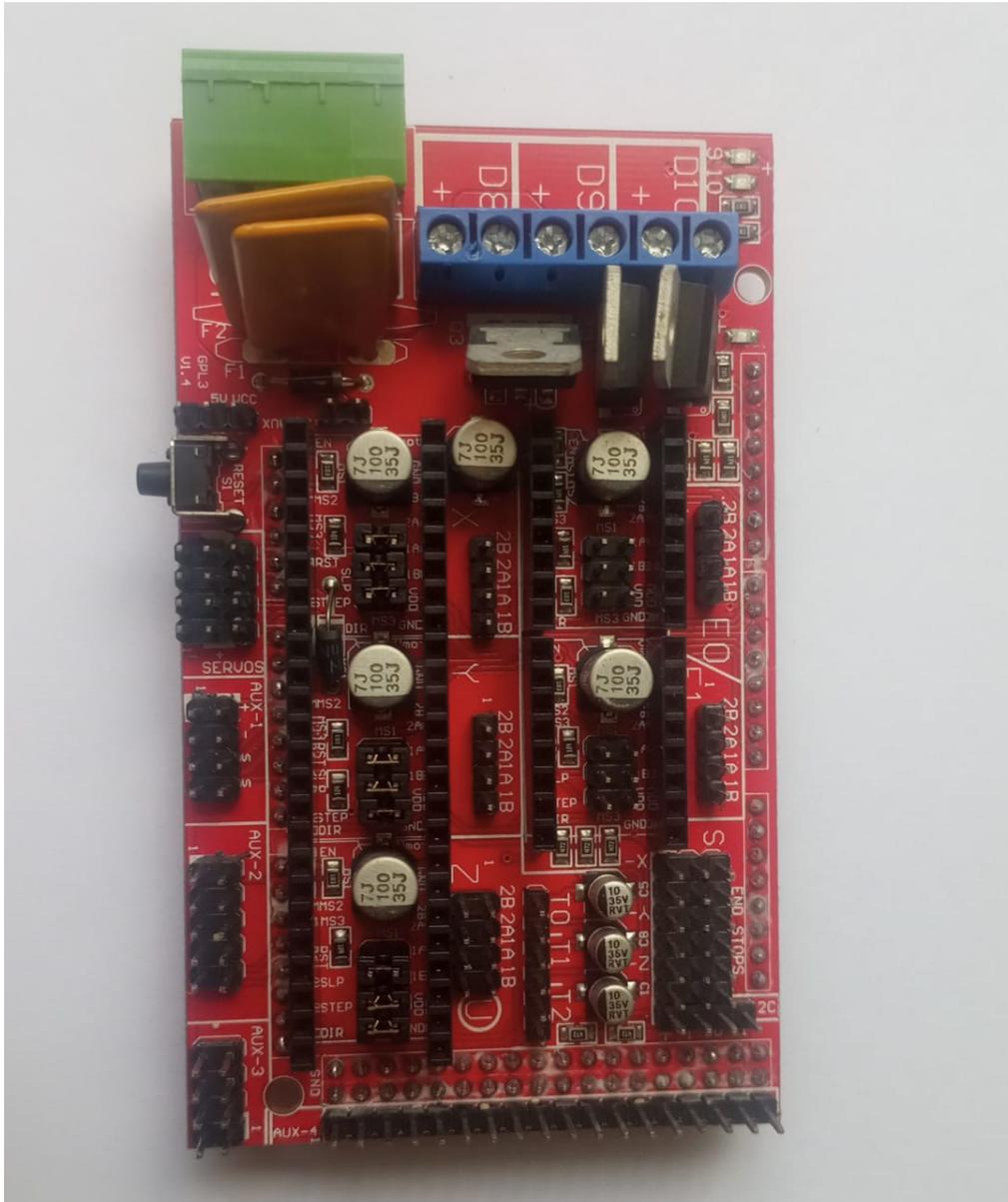


Figura 47. Ramps 1.4 con jumpers conectados en configuración 1/16 de paso.

6.2.3. Drivers A4988

El driver A4988 (figura 48) es una placa de soporte o una placa de arranque para el controlador del A4988 de Allegro con protección contra sobrecorriente [19].

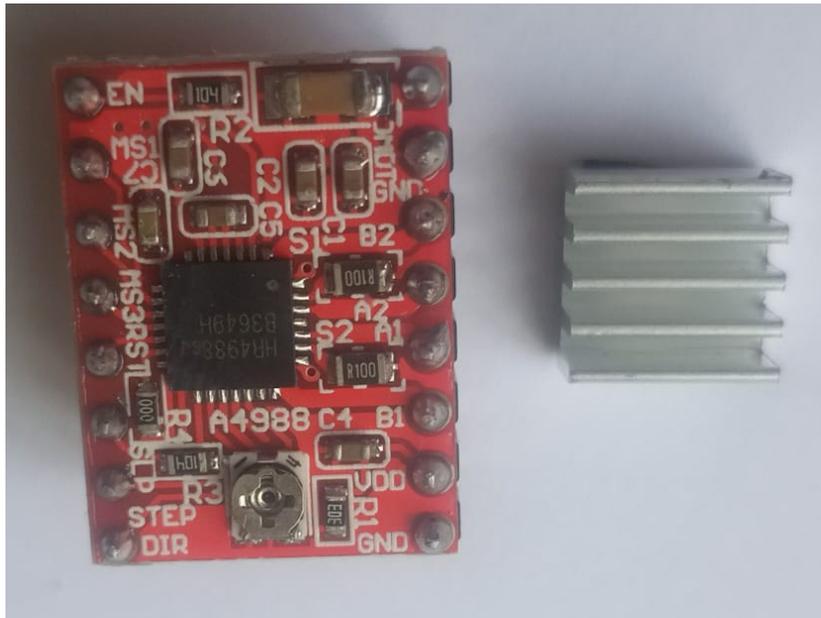


Figura 48. Driver A4988 con su disipador.

Para hacer un correcto ajuste de los drivers es importante ajustar el potenciómetro (R3 en la figura 48) según dice el fabricante en el *datasheet* (figura 49):

$$I_{tripMAX} = V_{ref} / (8 * R_s)$$

donde la intensidad máxima será la máxima intensidad tolerada por el motor y la resistencia “S” es la “*sense resistance*”.

The maximum value of current limiting is set by the selection of R_{Sx} and the voltage at the VREF pin. The transconductance function is approximated by the maximum value of current limiting, $I_{TripMAX}$ (A), which is set by

$$I_{TripMAX} = V_{REF} / (8 \times R_S)$$

where R_S is the resistance of the sense resistor (Ω) and V_{REF} is the input voltage on the REF pin (V).

Figura 49. Instrucciones del *datasheet*.

En el caso de nuestros drivers, la R_s tiene un valor de R100 (0,1 ohmios), por lo que se puede particularizar la ecuación para este caso concreto a la siguiente expresión:

$$V_{ref} = I_{tripMAX} * 0,8$$

6.2.4 Finales de carrera.

El último componente electrónico perteneciente a la parte de control son los finales de carrera (figura 50), que son los encargados de determinar cuál es la posición *home* indispensable para el correcto funcionamiento de la CNC. Es necesario uno por cada eje y, adicionalmente, se puede añadir uno más por cada eje para así determinar el inicio y el final de recorrido. Son interruptores que se activan cuando se alcanza el límite del recorrido.



Figura 50. Sensor final de carrera.

6.2.5. Conexión de los componentes.

La conexión de los componentes modulares se realiza mediante las indicaciones del fabricante siguiendo el siguiente orden:

1. Unir el módulo ramps 1.4 a la placa de arduino mega 2560.
2. Seleccionar la configuración de Jumpers para la ramps 1.4 que van a definir la cantidad de micropasos para los motores (figura 51).

A4988			
	MS1 Jumper	MS2 Jumper	MS3 Jumper
Single Step	0	0	0
1/2 Step	1	0	0
1/4 Step	0	1	0
1/8 Step	1	1	0
1/16 Step	1	1	1

Figura 51. Tabla de micropasos según los jumpers [18].

A la hora de seleccionar la configuración de micropasos se deberá tener en cuenta los siguientes aspectos: aumentar el número de micropasos produce un aumento de la resolución pero por otro lado produce una disminución del par incremental por micropaso. Pese a que el diseño propuesto no requeriría de más resolución, hacer uso de los micropasos también presenta algunas ventajas frente a no hacerlo; disminución del ruido mecánico, accionamiento más suave mecánicamente y reduce los problemas de resonancia [20][21].

3. Colocar los drivers A4988 en el módulo ramps 1.4 (figura 52).

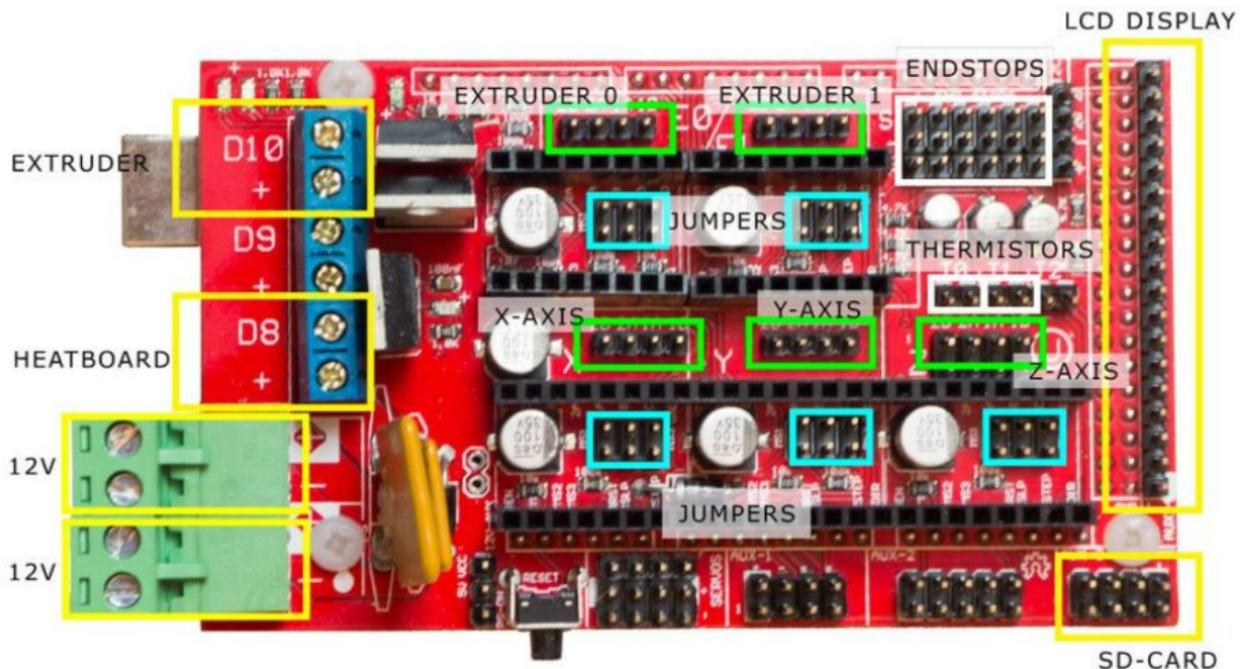


Figura 52. Conexión de la ramps 1.4 [18].

Cada driver controla un eje por lo que se colocará un driver por cada eje, en el eje X la máquina CNC es movida por dos motores, pero aún así es controlada por solo un driver. Es importante conocer la intensidad máxima que tienen los motores a la hora de escoger los drivers, en el caso de los A4988 la intensidad máxima que toleran son 2A, en el caso de necesitar motores superiores, se deberá de escoger unos drivers que toleren una intensidad superior.

En nuestro caso, la salida D8 mostrada en la figura 52 como *heatboard* será empleada para el ventilador de los drivers.

4. Y por último conectar los finales de carrera conectados en los pines mostrados en la figura 53.

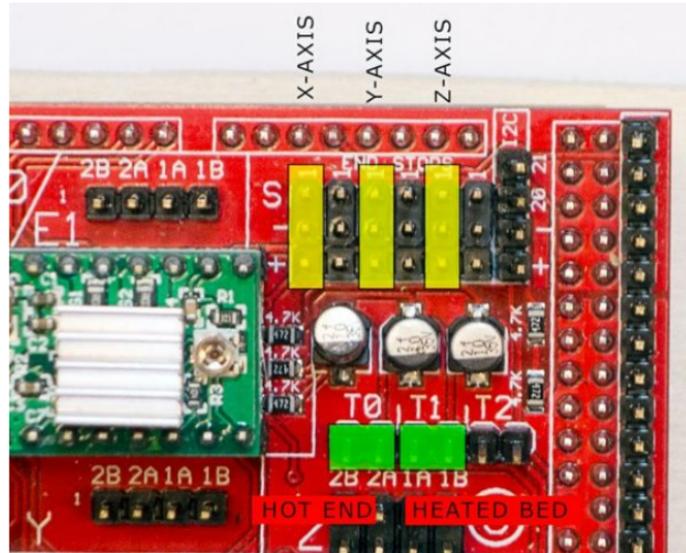


Figura 53. Conexión de los finales de carrera [18].

Como se explicará en el capítulo 7, es importante conectar los interruptores finales de carrera en el modo NC o *normally closed*.

La última parte de la electrónica es la dotación de energía al sistema mediante una fuente de alimentación de corriente continua. Las características necesarias de esta son: dotar un voltaje de salida de 12V y una corriente mínima de 16A (los 16A solo serán demandados si la CNC es usada como una impresora 3D para controlar la temperatura del extrusor y la cama caliente) como se indica el *datasheet* de la ramps 1.4.

7. Software de arduino mega 2560.

Para el control de la CNC se ha optado por el software Marlin (versión bugfix 2.1 obtenido en la plataforma *GitHub*), una solución *open source* ampliamente utilizada. Marlin permite el control de múltiples ejes de manera simultánea, lo que lo convierte en una herramienta ideal para el control de una CNC. A continuación se muestran los cambios a realizar en el código obtenido en para su correcto funcionamiento. Los primero ajustes que se debe realizar son en el archivo “configuration.h” que está en la ruta .../Marlin-bugfix-2.1.x/Marlin/Configuration.h.

El primer cambio en el código que se debe realizar es indicar que el Arduino Mega 2560 es la placa madre, a través de la ruta /Marlin-bugfix-2.1.x/Marlin/src/core/boards.h se encuentran las placas que sirven para este software.

```
// RAMPS 1.3 / 1.4 - ATmega1280, ATmega2560
//
#define BOARD_RAMPS_OLD          1000 // MEGA/RAMPS up to 1.2

#define BOARD_RAMPS_13_EFB       1010 // RAMPS 1.3 (Power outputs: Hotend, Fan, Bed)
#define BOARD_RAMPS_13_EEB       1011 // RAMPS 1.3 (Power outputs: Hotend0, Hotend1, Bed)
#define BOARD_RAMPS_13_EFF       1012 // RAMPS 1.3 (Power outputs: Hotend, Fan0, Fan1)
#define BOARD_RAMPS_13_EEF       1013 // RAMPS 1.3 (Power outputs: Hotend0, Hotend1, Fan)
#define BOARD_RAMPS_13_SF        1014 // RAMPS 1.3 (Power outputs: Spindle, Controller Fan)

#define BOARD_RAMPS_14_EFB       1020 // RAMPS 1.4 (Power outputs: Hotend, Fan, Bed)
#define BOARD_RAMPS_14_EEB       1021 // RAMPS 1.4 (Power outputs: Hotend0, Hotend1, Bed)
#define BOARD_RAMPS_14_EFF       1022 // RAMPS 1.4 (Power outputs: Hotend, Fan0, Fan1)
#define BOARD_RAMPS_14_EEF       1023 // RAMPS 1.4 (Power outputs: Hotend0, Hotend1, Fan)
#define BOARD_RAMPS_14_SF        1024 // RAMPS 1.4 (Power outputs: Spindle, Controller Fan)
```

Figura 54. Código de la librería Boards.h.

Como se observa en la Figura 54, aún sabiendo que la placa madre es la Arduino Mega 2560, se tienen diferentes configuraciones según el uso que vayamos a darle a nuestra máquina CNC. La configuración elegida en este proyecto es la BOARD_RAMPS_14_SF, en donde se controla la velocidad de rotación de la herramienta y la velocidad del ventilador que se encarga de enfriar los drivers.

```

69 #ifndef MOTHERBOARD
70 #define MOTHERBOARD BOARD_RAMPS_14_SF
71 #endif
72
73 /**
74  * Select the serial port on the board to
75  * This allows the connection of wireless
76  * Serial port -1 is the USB emulated seri
77  * Note: The first serial port (-1 or 0) w
78  *
79  * :[-1, 0, 1, 2, 3, 4, 5, 6, 7]
80  */
81 #define SERIAL_PORT 0
82
83 /**
84  * Serial Port Baud Rate
85  * This is the default communication speed
86  * Set the baud rate defaults for addition
87  *
88  * 250000 works in most cases, but you mig
89  * you commonly experience drop-outs durin
90  * You may try up to 1000000 to speed up S
91  *
92  * :[2400, 9600, 19200, 38400, 57600, 1152
93  */
94 #define BAUDRATE 250000
95

```

Figura 55. Primera modificación.

El *serial port*, al estar trabajando con Arduino, se debe poner en -1 o 0 (figura 55), mientras que el *baudrate*, que es la velocidad de comunicación del ordenador con la máquina CNC se deberá colocar en 250000.

```

209 // :[0, 1, 2, 3, 4, 5, 6, 7, 8]
210 #define EXTRUDERS 0

```

Figura 56. Segunda modificación.

Al estar configurando el software para una máquina CNC general y no una impresora, se tendrá que indicar que no se tienen extrusores (figura 56). Más adelante se ven las modificaciones de código que se deben realizar para que este cambio no presente ningún problema a la hora de compilar.

Un parámetro importante a configurar son los *steps/mm* pues un valor incorrecto en este parámetro provocará que el movimiento de la máquina CNC no sea el deseado. Para calcular este parámetro se debe tener en cuenta el paso de rosca del tornillo sin fin ($d=8$ mm), la configuración de los drivers ($a=1/16$ steps) y los grados por paso de los motores paso a paso ($b=1.8$ °/step). Debido a todos estos valores el parámetro *steps/mm* se calcula de la siguiente forma:

$$\text{steps/mm} = \frac{1}{a} * \frac{360^\circ}{b} * \frac{1}{d}$$

Como se ve en la fórmula los tres parámetros que influyen en el movimiento de la CNC son la configuración de los drivers, la resolución de los motores y el paso de rosca de los tornillos sin fin.

$$\begin{aligned} \text{steps/mm} &= 16 * \frac{360^\circ}{1.8} * \frac{1}{8} \\ \text{steps/mm} &= 400 \end{aligned}$$

```
/**
 * Default Axis Steps Per Unit (linear=steps/mm, rotational=steps/°)
 * Override with M92
 *
 *                               X, Y, Z [, I [, J [, K...]]], E0 [, E1[, E2...]]
 */
#define DEFAULT_AXIS_STEPS_PER_UNIT { 400, 400, 400 }
```

Figura 57. Tercera modificación.

Es muy importante que se elimine el cuarto valor del vector “DEFAULT_AXIS_STEPS_PER_UNIT” que trae de forma predeterminada (figura 57), ya que este es el valor del motor del extrusor y como se indica anteriormente, no se está incluyendo el extrusor en el código.

```
/**
 * Default Max Feed Rate (linear=mm/s, rotational=°/s)
 * Override with M203
 *
 *                               X, Y, Z [, I [, J [, K...]]], E0 [, E1[, E2...]]
 */
#define DEFAULT_MAX_FEEDRATE { 100, 100, 100 }
```

Figura 58. Cuarta modificación.

El siguiente parámetro a modificar es la velocidad máxima por eje de la máquina CNC. Tal y como se calculó en el apartado 8, la velocidad máxima que permiten los motores en este diseño era de aproximadamente 13 cm/s (figura 58), pues en el código se debe establecer un valor igual o inferior a este para que la máquina CNC opere de forma correcta. Al igual que en el caso anterior se debe de eliminar el cuarto elemento del vector “DEFAULT_MAX_FEEDRATE”.

```
/**
 * Default Max Acceleration (speed change with time) (linear=mm/(s^2), rotational=°/(s^2))
 * (Maximum start speed for accelerated moves)
 * Override with M201
 *
 *                               X, Y, Z [, I [, J [, K...]]], E0 [, E1[, E2...]]
 */
#define DEFAULT_MAX_ACCELERATION { 3000, 3000, 500 }
```

Figura 59. Quinta modificación.

El siguiente parámetro es la aceleración. El valor de este parámetro dependerá de las cualidades que se le quiera dar a la CNC, pero habría que tener en cuenta que mientras más grande sea el diseño de la máquina, mayores serán las fuerzas ejercidas sobre la estructura, si no se reduce este parámetro puede dañar las piezas impresas en 3D. Al igual que con los dos cambios anteriores, se debe de eliminar el valor de la aceleración del extrusor en el vector "DEFAULT_MAX_ACCELERATION" (figura 59).

```
// The size of the printable area
#define X_MAX_SIZE 400
#define Y_MAX_SIZE 350
#define Z_MAX_SIZE 100

// Travel limits (linear=mm, rotational=°) after homing, corresponding to endstop positions.
#define X_MIN_POS 0
#define Y_MIN_POS 0
#define Z_MIN_POS 0
#define X_MAX_POS X_MAX_SIZE
#define Y_MAX_POS Y_MAX_SIZE
#define Z_MAX_POS Z_MAX_SIZE
//#define T_MIN_POS 0
```

Figura 60. Sexta modificación.

Un parámetro muy importante a modificar son las dimensiones de la zona de trabajo de nuestra máquina. Esta dependerá de la longitud de los perfiles, a la que se tiene que restar únicamente la distancia desaprovechada por las piezas diseñadas (figura 60).

```
141 #define X_DRIVER_TYPE A4988
142 #define Y_DRIVER_TYPE A4988
143 #define Z_DRIVER_TYPE A4988
```

Figura 61. Séptima modificación.

En la parte del código mostrada en la figura 61. Se indican los drivers que se están usando para controlar los motores.

```
#define USE_XMIN_PLUG
#define USE_YMIN_PLUG
#define USE_ZMIN_PLUG
//#define USE_IMIN_PLUG
//#define USE_JMIN_PLUG
//#define USE_KMIN_PLUG
//#define USE_UMIN_PLUG
```

Figura 62. Octava modificación.

Para activar los finales de carrera que se estén utilizando, se debe descomentar la línea que los define (figura 62). Para un funcionamiento más seguro de la máquina CNC se usarán los finales de carrera en el estado normalmente cerrado, ya que este estado permite detectar la presencia de errores. Por ejemplo, si el cable se desconecta y los finales de carrera están colocados en configuración normalmente abiertos, nunca detectará el final de carrera haciendo que la máquina se desplace hasta el límite sin detenerse. El indicador del modo que se está usando con los finales de carrera está en la siguiente parte del código.

```
#define X_MIN_ENDSTOP_HIT_STATE HIGH
#define X_MAX_ENDSTOP_HIT_STATE HIGH
#define Y_MIN_ENDSTOP_HIT_STATE HIGH
#define Y_MAX_ENDSTOP_HIT_STATE HIGH
#define Z_MIN_ENDSTOP_HIT_STATE HIGH
#define Z_MAX_ENDSTOP_HIT_STATE HIGH
```

Figura 63. Novena modificación.

El último parámetro a la hora de configurar los finales de carrera, es indicar la dirección en la que se encuentran (figura 63).

```
// Direction of endstops when homing; 1=MAX, -1=MIN
// :[-1,1]
#define X_HOME_DIR -1
#define Y_HOME_DIR -1
#define Z_HOME_DIR -1
```

Figura 64. Décima modificación.

Indicando un -1 si los finales de carrera se encontrarán desplazando los motores en sentido negativo (en el límite inferior del recorrido) o desplazando estos en sentido positivo (en el límite superior del recorrido)(figura 64).

Los siguientes parámetros a configurar, deberán modificarse en el archivo "Configuration.h", al que se accede mediante la ruta ../Marlin-bugfix-2.1.x/Marlin/Configuration.h. En este archivo se modifican los parámetros del ventilador.

Con la intención de coordinar el movimiento del eje Z con el GCode generado en el capítulo siguiente, se debe hacer que el motor del eje Z se mueva en sentido contrario. Para ello se debe modificar la línea de código mostrada en la figura 65. El sentido de giro de los motores se puede modificar tanto en el código como físicamente, para realizar el cambio de forma física se deberá invertir el conector del motor al driver A4988.

```

1698 // Invert the stepper direction. Change (or reverse the motor connector) if an axis goes the wrong way.
1699 #define INVERT_X_DIR false
1700 #define INVERT_Y_DIR true
1701 #define INVERT_Z_DIR true

```

Figura 65. Undécima modificación.

```

#define USE_CONTROLLER_FAN
#if ENABLED(USE_CONTROLLER_FAN)
  #define CONTROLLER_FAN_PIN MOSFET_C_PIN
  // #define CONTROLLER_FAN2_PIN -1
  // #define CONTROLLER_FAN_USE_Z_ONLY
  // #define CONTROLLER_FAN_IGNORE_Z
  #define CONTROLLERFAN_SPEED_MIN 40
  #define CONTROLLERFAN_SPEED_ACTIVE 70
  #define CONTROLLERFAN_SPEED_IDLE 50
  #define CONTROLLERFAN_IDLE_TIME 60

```

Figura 66. Duodécima modificación.

Esta parte del código controla el ventilador encargado de enfriar los drivers en un rango que va de 0 a 255, siendo 0 estado apagado y 255 a máxima potencia (figura 66). Hay una serie de parámetros que se pueden controlar, entre los que están, estos parámetros controlar la el ventilador encargado de enfriar los drivers según el estado de operación de la máquina CNC, como se comentó anteriormente, dependiendo de los drivers escogidos se deberá de configurar el ventilador de una forma y otra para tener un rendimiento óptimo:

- **CONTROLLER_FAN_PIN:** parámetro que indica qué pin va a controlar el ventilador. En la ramps 1.4 existen tres pines que se pueden encargar de esta tarea y son los pines conectados a los mosfets. Para saber que pines son estos, se puede acceder al archivo de pines con la ruta `.../Marlin-bugfix-2.1.x/Marlin/src/pins/ramps/pins_RAMPS.h`.

```
//
// Heaters / Fans
//
#ifndef MOSFET_A_PIN
#define MOSFET_A_PIN ..... 10
#endif
#ifndef MOSFET_B_PIN
#define MOSFET_B_PIN ..... 9
#endif
#ifndef MOSFET_C_PIN
#define MOSFET_C_PIN ..... 8
#endif
#ifndef MOSFET_D_PIN
#define MOSFET_D_PIN ..... 1
#endif
```

Figura 67. Pins_RAMPS.h.

Se debe definir con el nombre del mosfet o con el número correspondiente a dicho mosfet (Figura 67).

- **CONTROLLERFAN_SPEED_MIN:** indica la velocidad mínima a la que se encenderá el ventilador.
- **CONTROLLERFAN_SPEED_ACTIVE:** indica la velocidad del ventilador cuando alguno de los motores esté en funcionamiento, es decir los drivers estén trabajando.
- **CONTROLLERFAN_SPEED_IDLE:** indica la velocidad del ventilador después de que los motores se hayan detenido, es decir cuando los drivers hayan dejado de funcionar.
- **CONTROLLERFAN_IDLE_TIME:** indica el tiempo que tardará en apagarse el ventilador después de que los motores se hayan detenido. Este parámetro trabajará conjuntamente con el **CONTROLLERFAN_SPEED_IDLE**.

8. Generación de GCode con Inkscape.

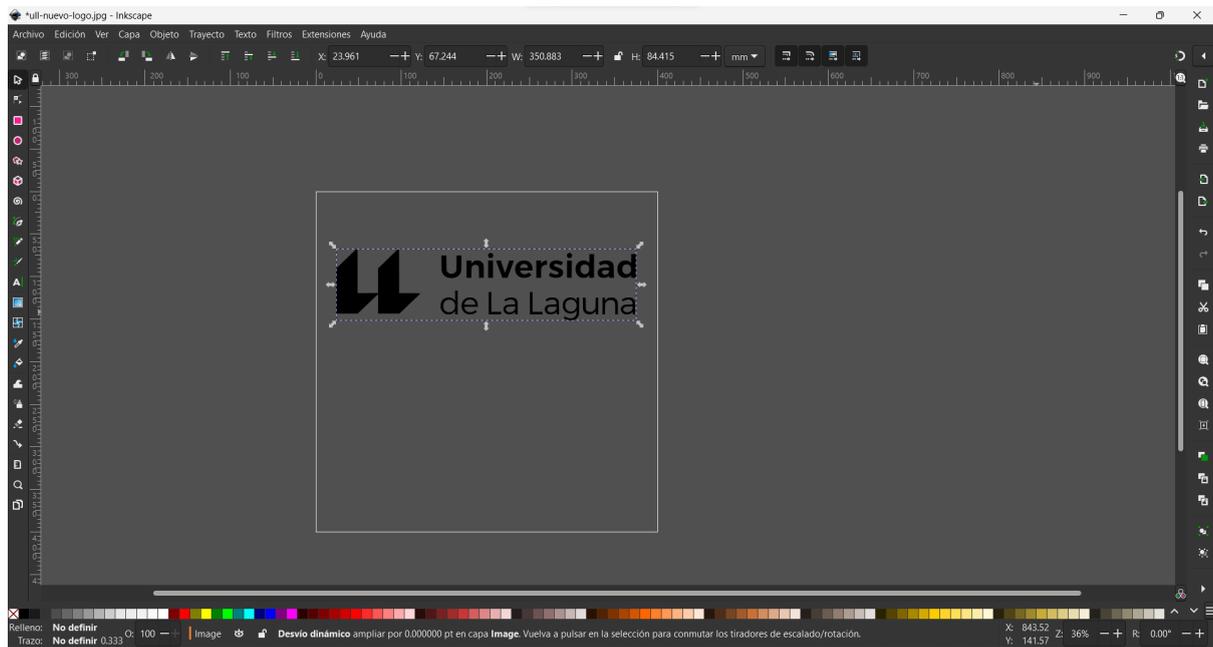


Figura 68. Programa Inkscape.

GCode es un lenguaje de programación que se usa en la industria para controlar las máquinas CNC siguiendo los estándares de la ISO-6983. Se compone de una serie de instrucciones para que la máquina CNC se mueva de forma deseada. Este se compone de letras y números, donde cada letra corresponde a una acción específica y los números corresponden a parámetros asociados a esta acción [22].

El GCode puede generarse manualmente, pero normalmente se realiza mediante un software. En este caso, el software empleado para la generación de GCodes es el Inkscape (<https://inkscape.org/es/> vease figura 68) pues es una opción de código abierto.

A la hora de generar el GCode, en primer lugar se debe tener la imagen de lo que se quiera convertir a GCode. Con la imagen abierta en el programa, es importante comprobar que las unidades del programa están en milímetros y no en pixeles (formato predeterminado figura 69). De no ser así deberá cambiarse en la ruta Archivo/Propiedades del documento. Posteriormente se debe ajustar el ancho y el alto de la imagen al espacio de trabajo que tiene nuestra máquina CNC. Si la imagen excede los límites la máquina no será capaz de llegar a cabo el trabajo.

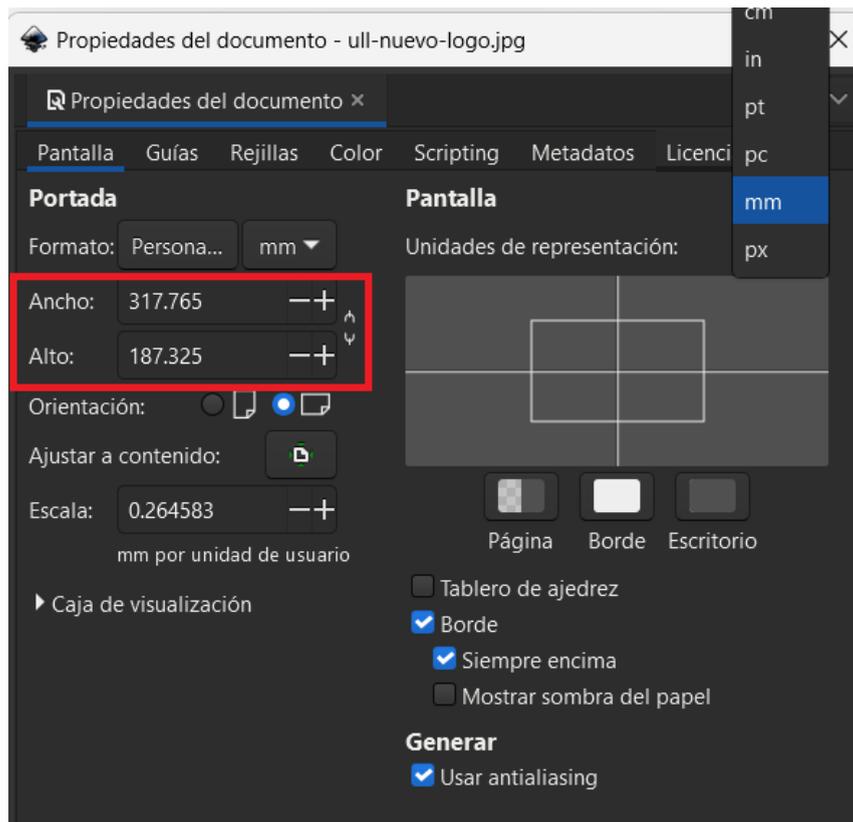


Figura 69. Unidades y dimensiones del espacio de Inkscape.

El siguiente paso será vectorizar la imagen para convertirla en una imagen de vectores que la máquina CNC pueda interpretar. Esto se realiza a través de la ruta Trayecto/Vectorizar mapa de bits. Con esta opción, se abrirá una ventana emergente donde se ajustará el parámetro “umbral” para vectorizar de forma correcta la imagen. Este valor servirá para separar la imagen de los detalles que se desean implementar en el GCode.

Una vez el “umbral” esté en el valor deseado (figura 70), al aceptar el programa generará sobre la imagen original una imagen vectorizada. La imagen vectorizada es la que importa a la hora de generar el gcode, por ello se debe borrar la imagen original una vez realizado este paso.

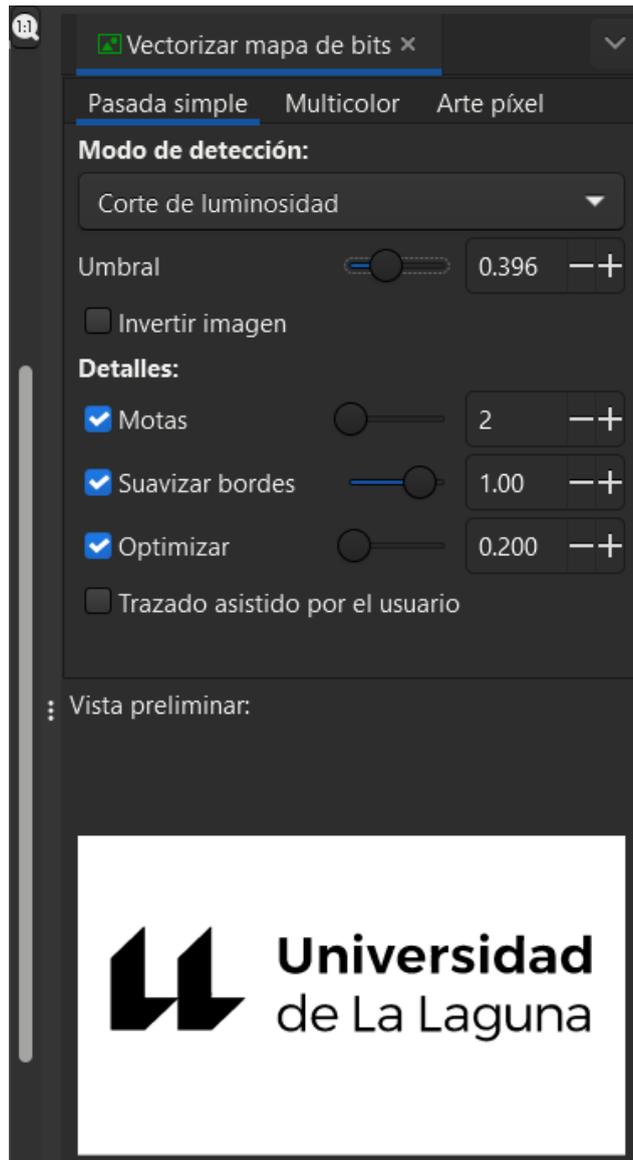


Figura 70. Imagen convertida a mapa de bits.

Para poder generar un trayecto dinámico de una imagen vectorizada se debe realizar dos operaciones con la imagen vectorizada. La primera operación a través de la ruta Herramientas/Objeto a trayecto (figura 71), y la segunda a través de la ruta Herramientas/Desvío dinámico (figura 71). Al seleccionar la primera opción se convierte la imagen vectorizada en un objeto trazable, lo que permite tratar las líneas y curvas de manera independiente. Mientras que al seleccionar la última opción, se crea una copia de la imagen vectorizada pero distorsionada [23].

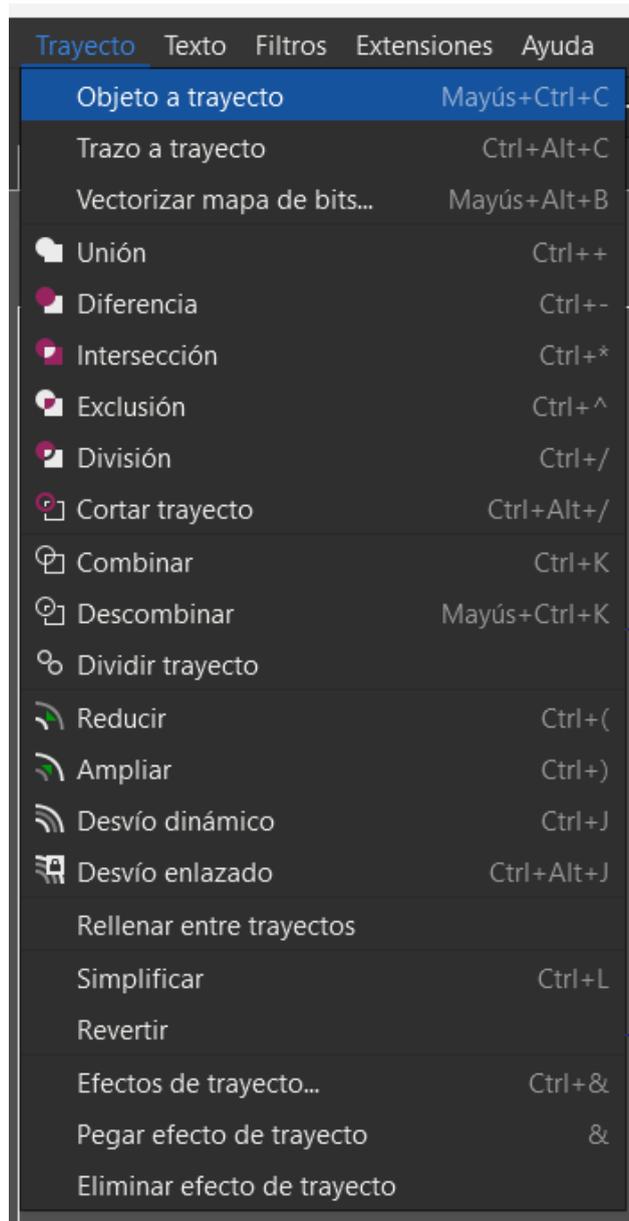


Figura 71. Pestaña de "Trayecto".

En adelante, se usa la ventana de extensiones. El primer paso en esta ventana será usar los puntos de orientación. Son pequeñas marcas situadas de forma estratégica para que la máquina pueda detectarlos y ajuste bien la posición y orientación del cabezal de la herramienta.

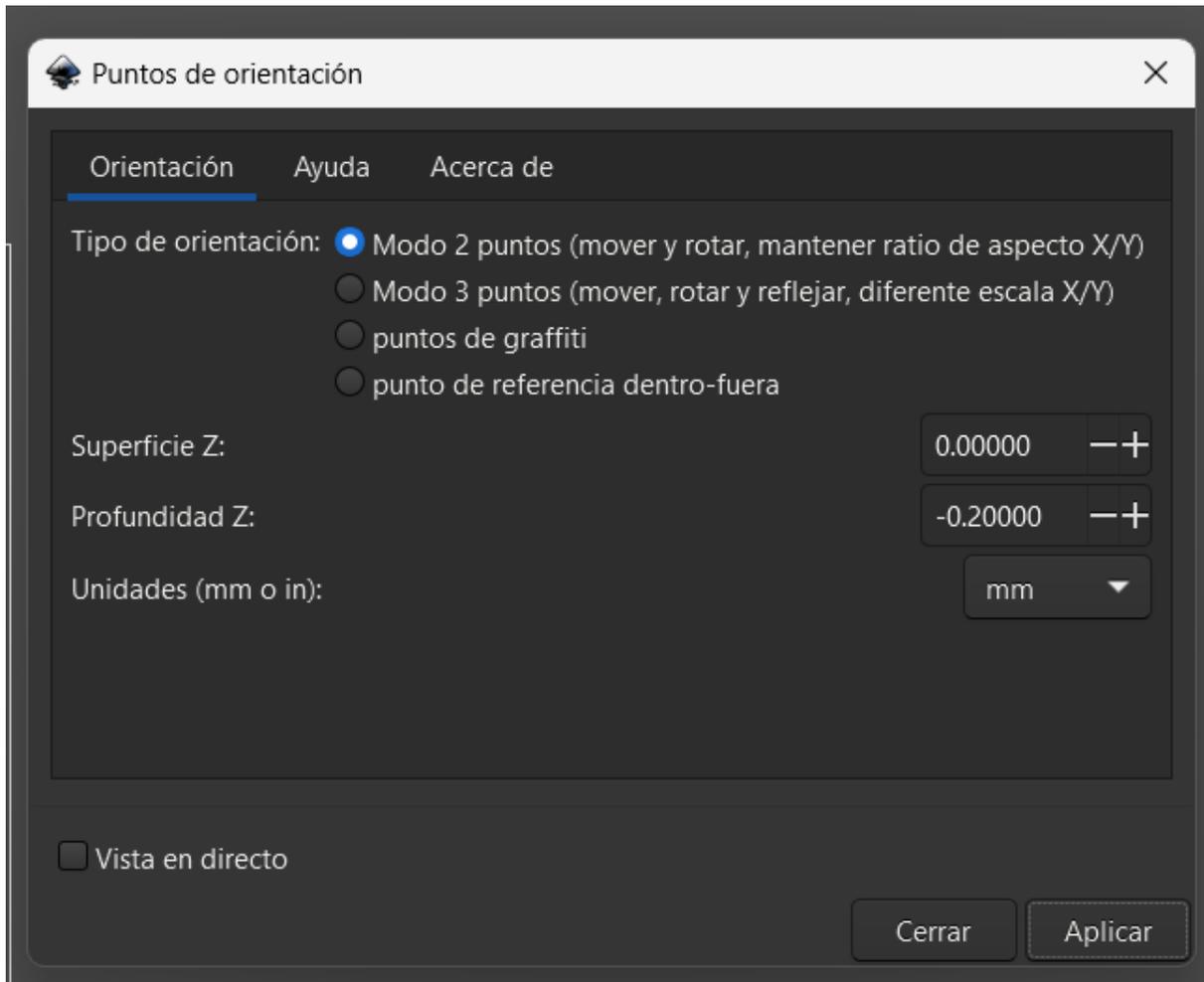


Figura 72. Puntos de orientación.

La profundidad Z es el parámetro importante en esta pestaña (figura 72), pues aquí es donde se determina el alcance que va tener la máquina CNC en el eje Z. Por ejemplo, si se quiere realizar un dibujo, habrá que indicarle la distancia que hay desde la posición “home” hasta el papel.

El siguiente paso en la pestaña de extensiones, será indicar que herramienta se va a utilizar (figura 73), para que ajuste el recorrido de la máquina, para ello, a través de la ruta Extensiones/GCodeTools/Biblioteca de herramientas, se indica la herramienta que vamos a usar.

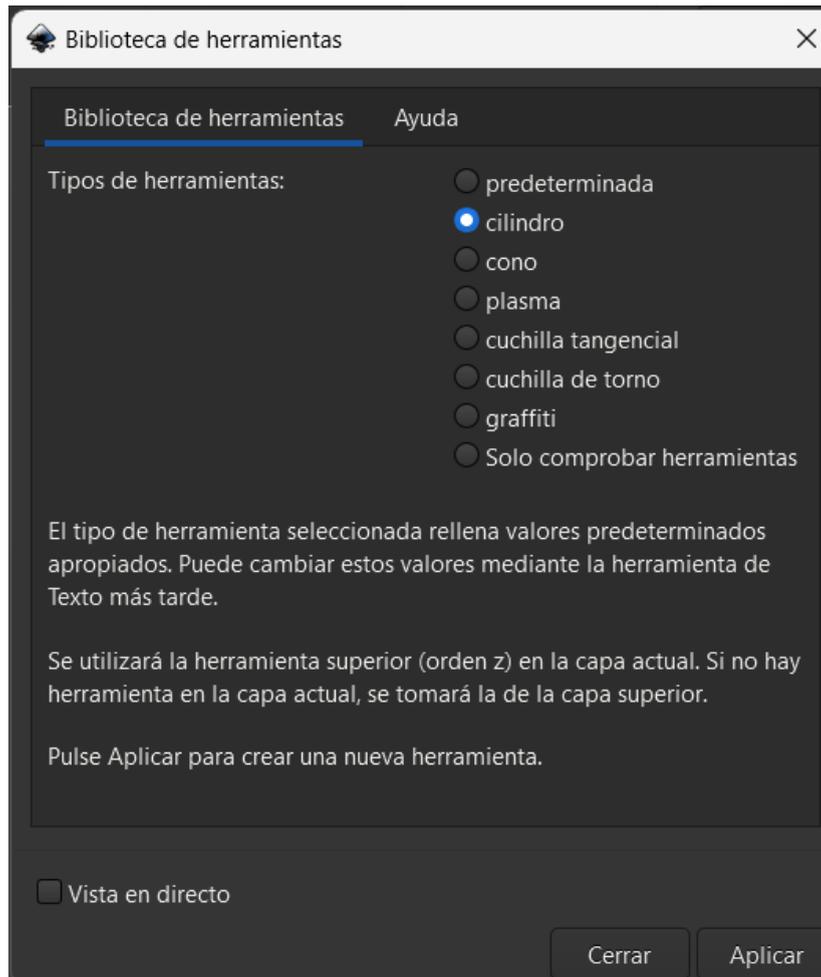


Figura 73. Biblioteca de herramientas.

name	Cylindrical cutter
id	Cylindrical cutter 0001
diameter	0.10
feed	2000
penetration angle	90
penetration feed	100
depth step	0.1
tool change gcode	(None)

Figura 74. Parámetros de la CNC.

Al realizar la última operación se abrirá una pestaña verde como el de la figura 74. En esta pestaña verde se indican los parámetros que tendrá nuestra herramienta a la hora de

generar el GCode, en el caso de la figura 74 se están teniendo en cuenta, por ejemplo, el diámetro de la fresadora, la velocidad de desplazamiento a la hora de desbastar el material, el ángulo de ataque sobre la pieza, etc:

- **Diameter:** diámetro de la herramienta, que el software tendrá en cuenta para que la máquina ajuste su recorrido y no se coma material al desplazarse.
- **Feed:** velocidad de la máquina CNC mientras esté trabajando, valor que se debe ajustar dependiendo del trabajo que esté llevando a cabo la misma.
- **Penetration angle:** ángulo que tiene la herramienta con respecto al plano XY.
- **Penetration feed:** velocidad de movimiento en el eje Z.
- **Depth step:** avance que tendrá la máquina en el eje Z por cada vuelta, importante para el caso de fresadoras.

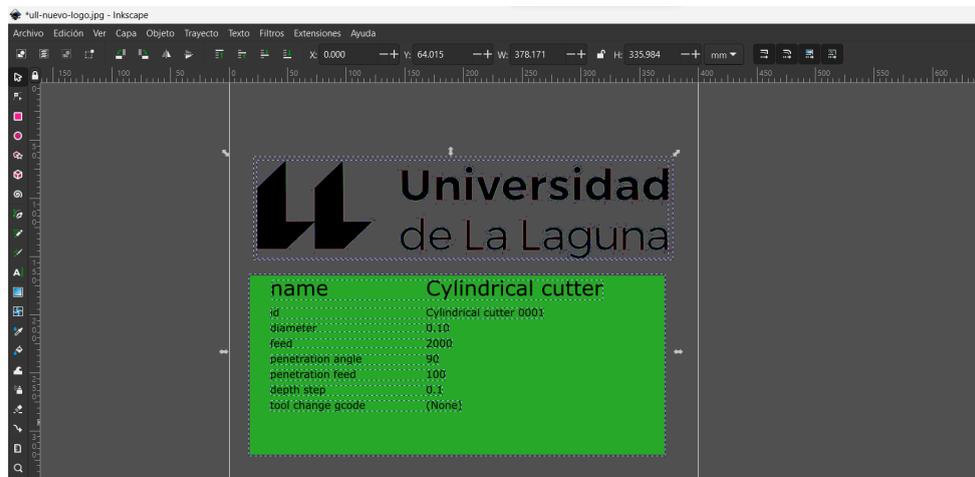


Figura 75. Información completa.

Una vez todo esté preparado, para generar el GCode, se selecciona todo y a través de la ruta Extensiones/GCodetools/Trayecto a GCode (figura 75 y 76). De esta manera se generará el archivo con el GCode en un formato .ngc, este paso generará el GCode que la máquina CNC usará para el mecanizado de las piezas.

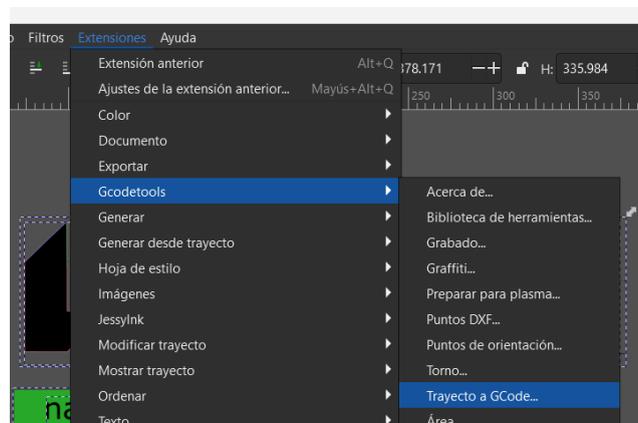


Figura 76. Trayecto a GCode.

```

(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: path177)
(Change tool to Cylindrical cutter)

G00 Z3.000000
G00 X182.919920 Y1257.657660

G01 Z-0.100000 F100.0(Penetrates)
G03 X180.386811 Y1256.649089 Z-0.100000 I-0.000000 J-3.685341 F2000.000000
G03 X99.066406 Y1178.302190 Z-0.100000 I4337.750719 J-4583.758559
G01 X90.560547 Y1169.956480 Z-0.100000
G01 X90.603517 Y1069.747500 Z-0.100000

```

Figura 77. GCode.

El significado de los comandos generados [22] están determinados por unos estándares que determinan el movimiento de las máquinas CNC para que todas funcionen de la misma forma. En la figura 77 se pueden observar varios de estos comandos. Dichos comandos se determinan por la primera letra de la instrucción, la cual puede tomar los siguientes valores:

El software encargado de generar el GCode, generará una sucesión de líneas de código que permitirá a la máquina desplazar el efector final por la zona deseada a la velocidad deseada.

- G: comando que indica desplazamiento.
- M: comando que actúa sobre la herramienta.
- X, Y, Z: comandos usados para indicar posición.
- F: comando que indica la velocidad de avance de la herramienta.
- S: comando empleado para controlar la velocidad de rotación del cabezal de la herramienta de corte.
- T: comando para indicar un cambio de herramienta.
- I, J, K: comandos empleados para el control de un desplazamiento circular.

9. Pronterface.

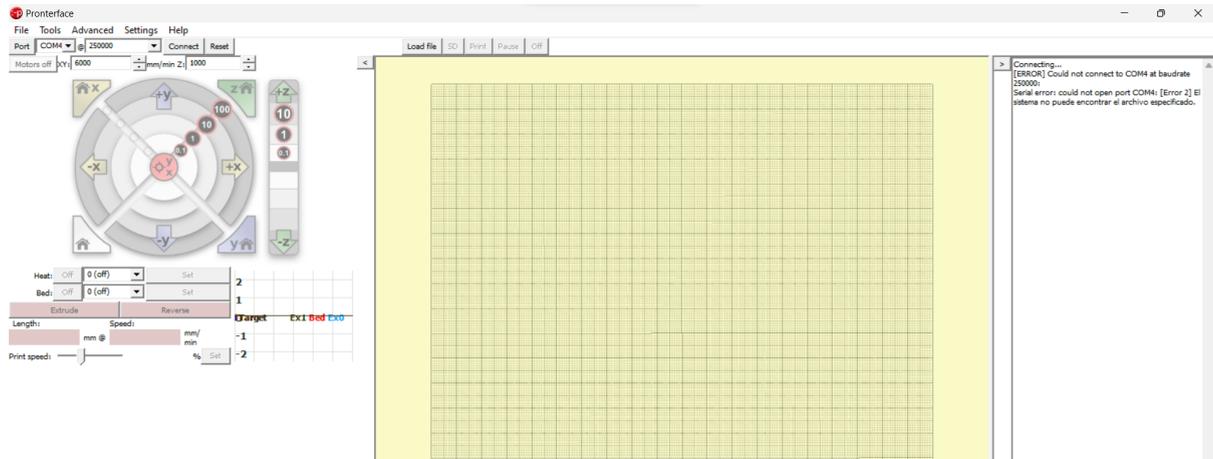
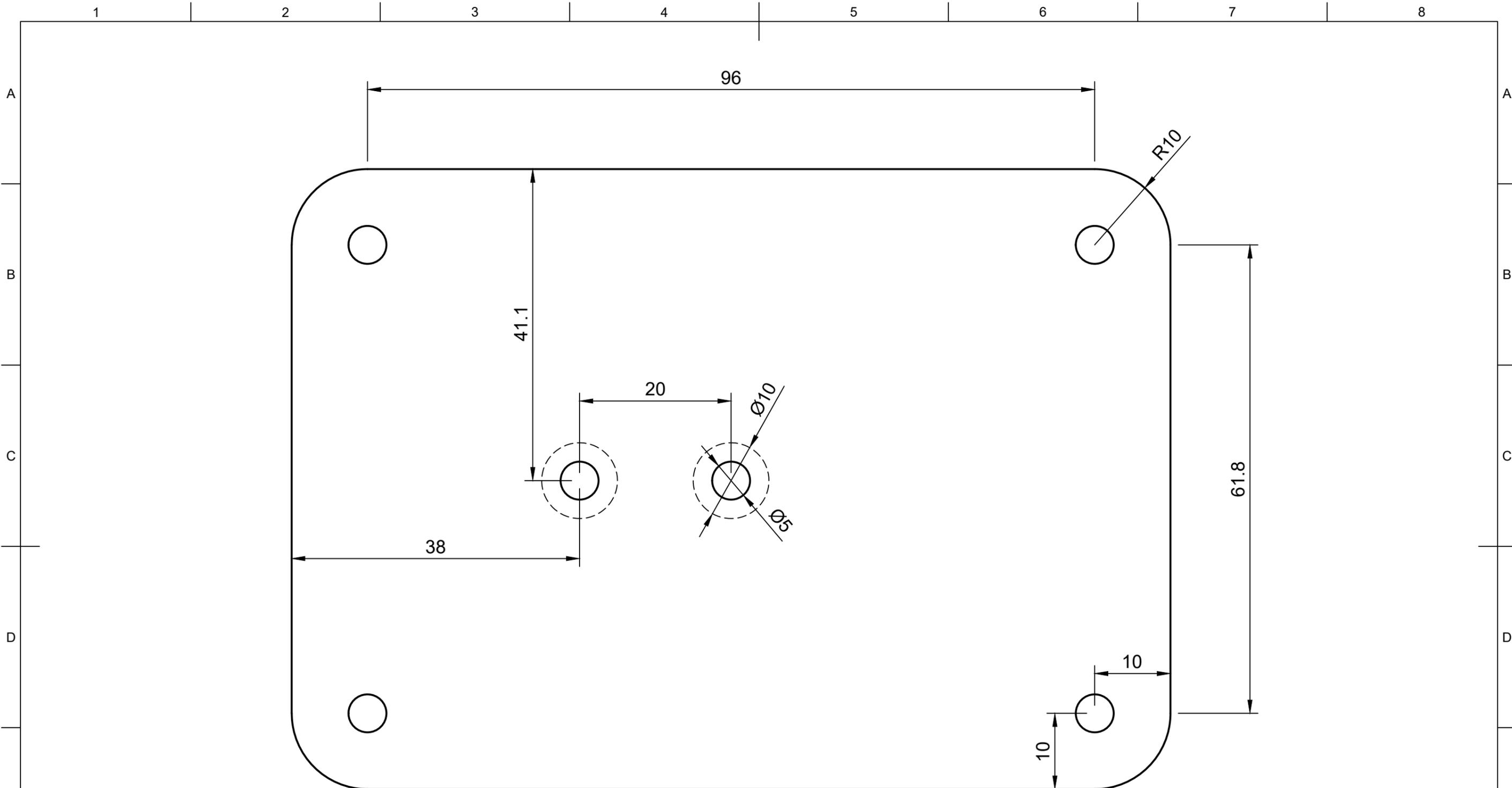


Figura 78. Interfaz de Pronterface.

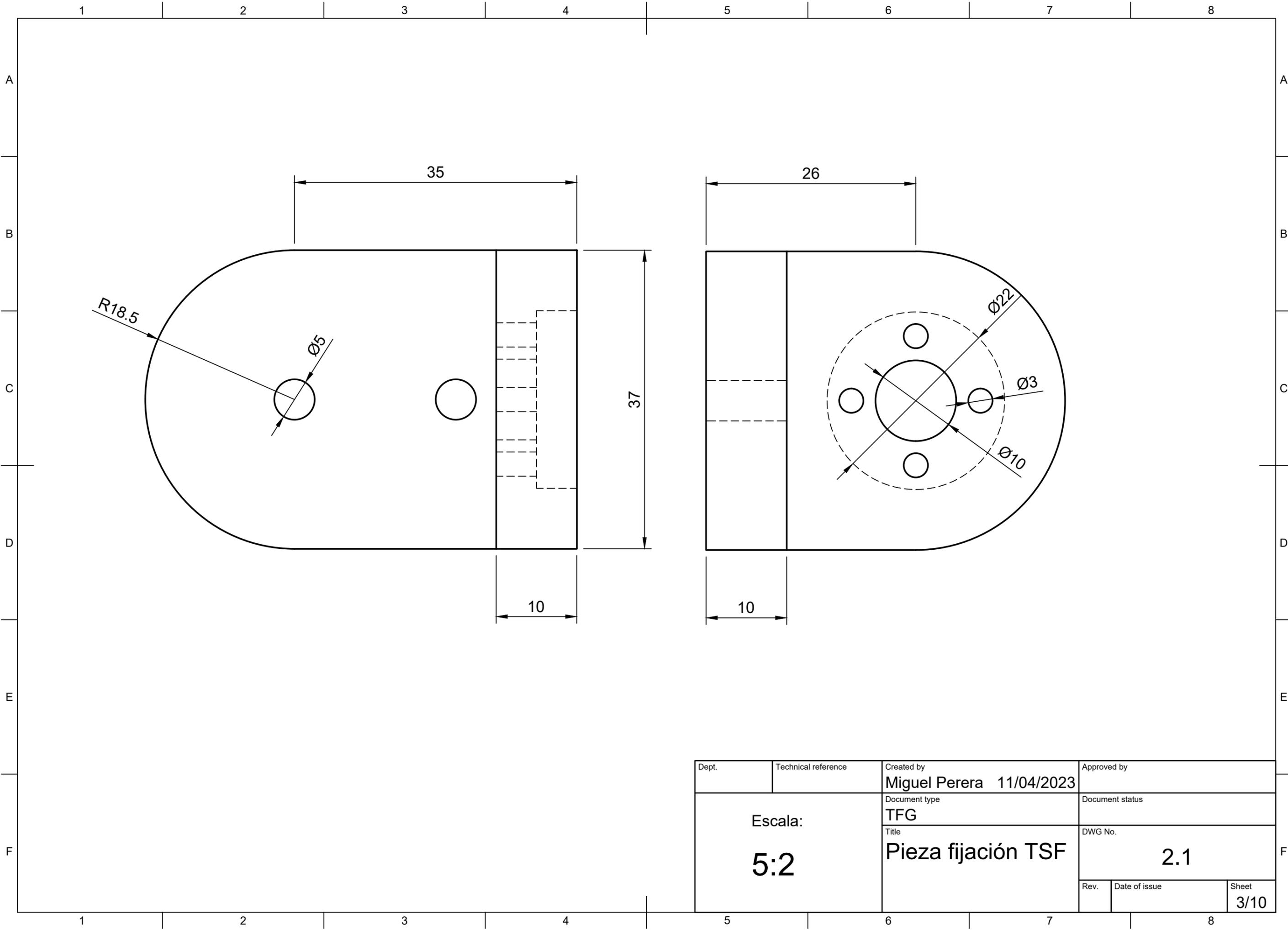
Pronterface es un programa de código abierto (figura 78), diseñado para controlar máquinas CNC. Permite a los usuarios controlar las máquinas CNC a través de su ordenador mediante una interfaz fácil de usar. (<https://www.pronterface.com/>).

El uso que se le va a dar en este proyecto es el de ejercer de puente entre el GCode y el software del arduino, es decir, este programa se encargará de ir transmitiendo línea a línea el GCode al software Marlin

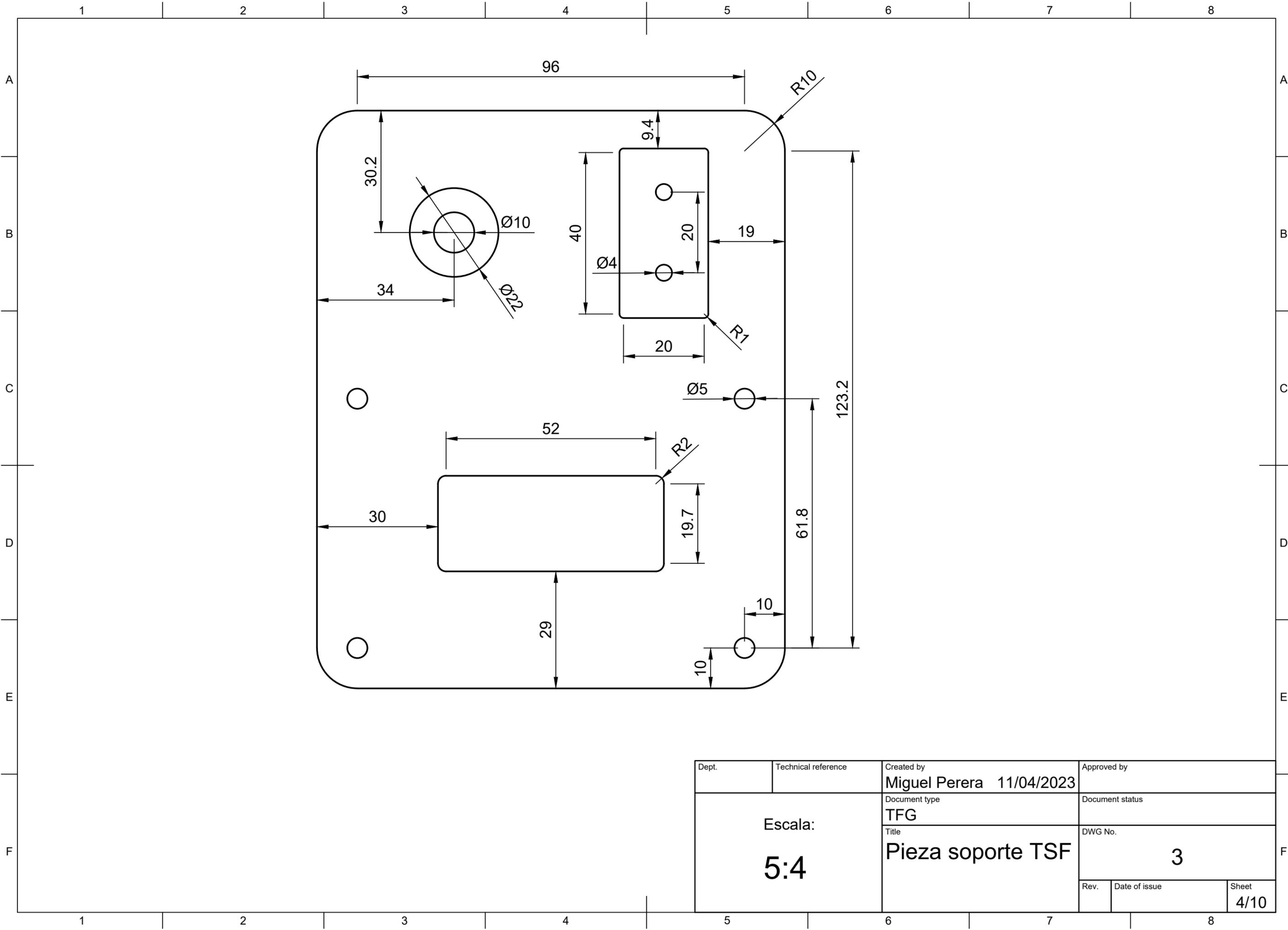
10. Planos.



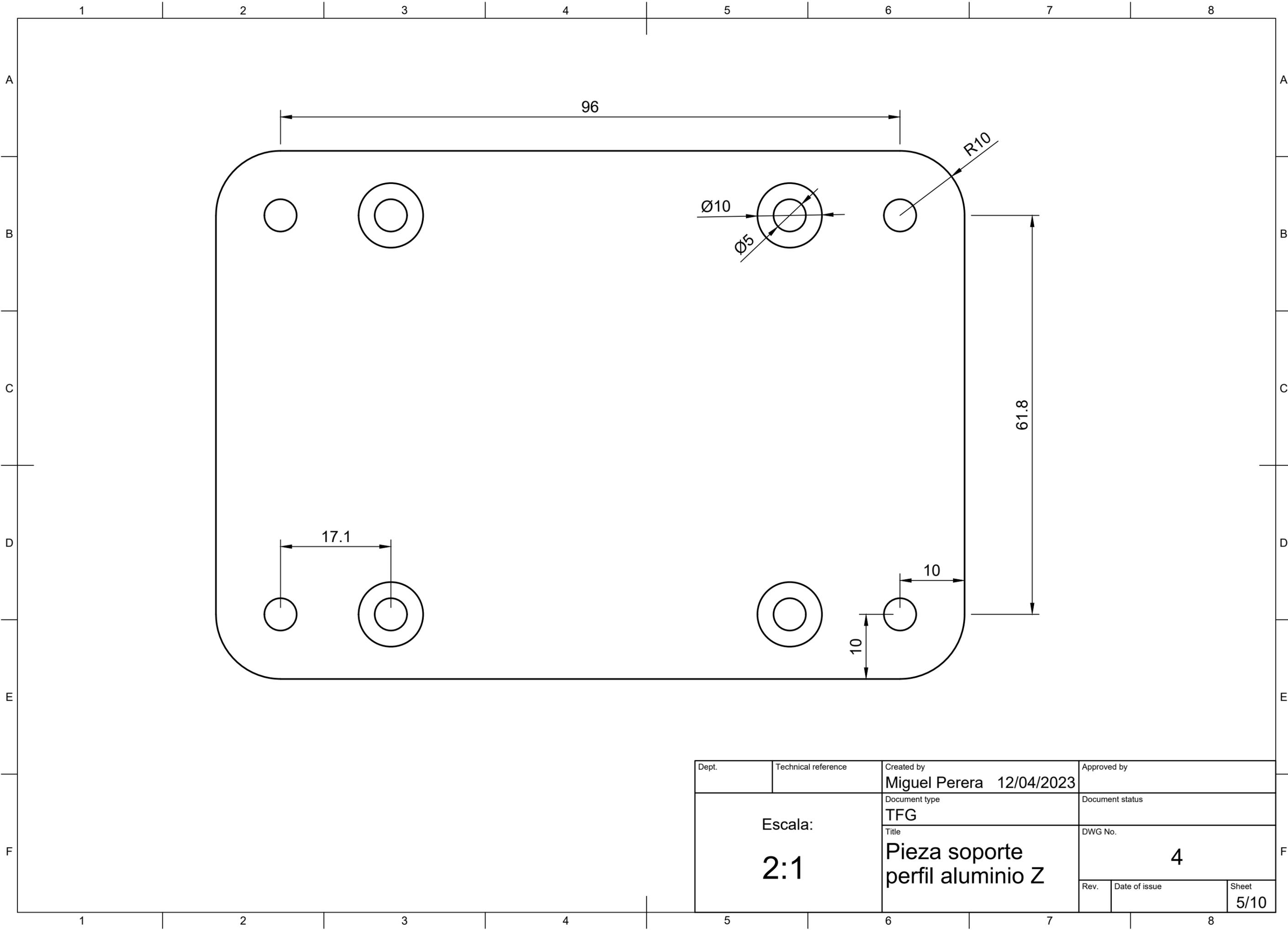
Dept.	Technical reference	Created by Miguel Perera 11/04/2023	Approved by
Escala: 2:1		Document type TFG	Document status
		Title Pieza soporte de la fijación del TSF	DWG No. 2
	Rev.	Date of issue	Sheet 2/10



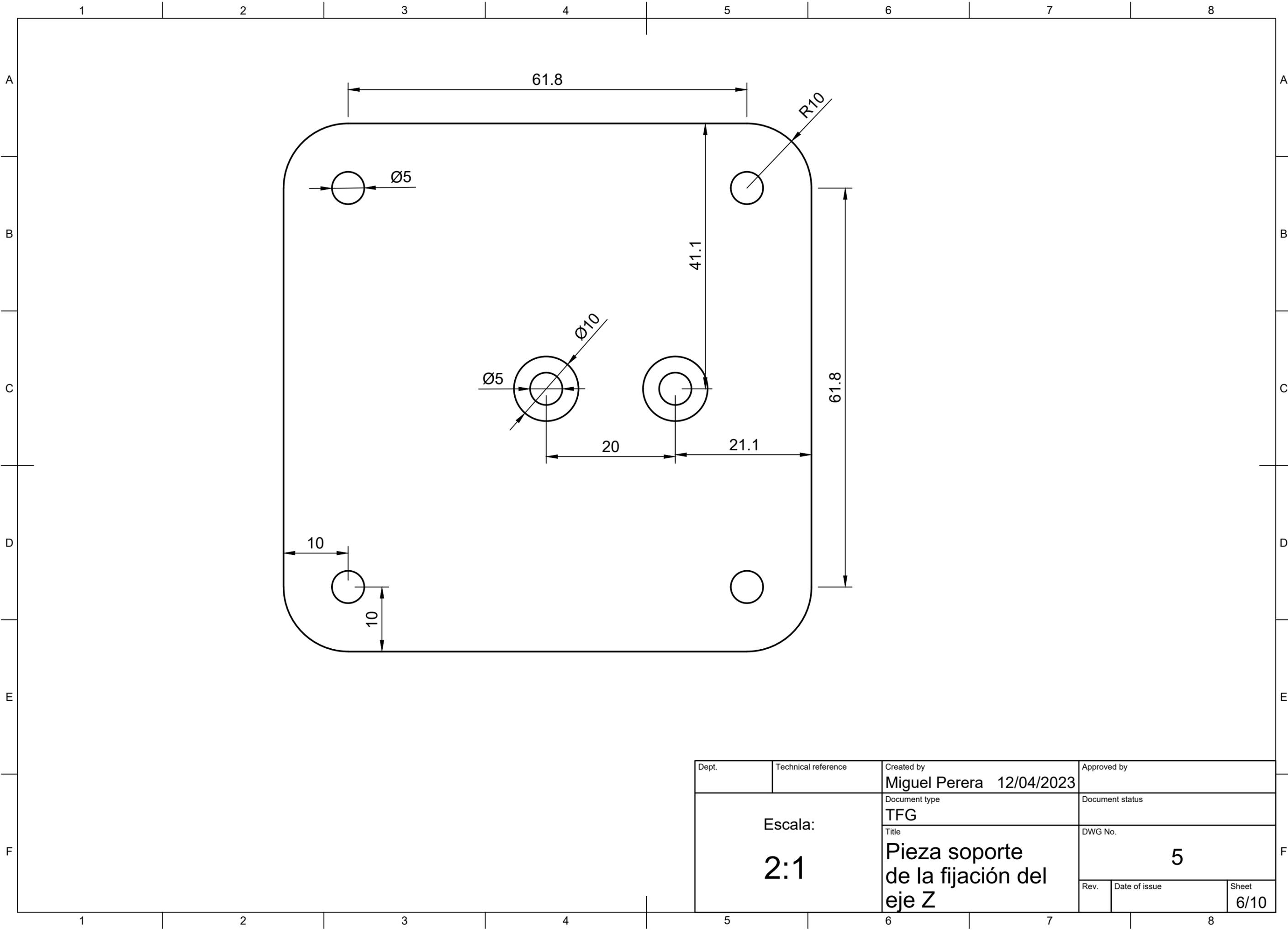
Dept.	Technical reference	Created by Miguel Perera 11/04/2023	Approved by
Escala: 5:2		Document type TFG	Document status
		Title Pieza fijación TSF	DWG No. 2.1
	Rev.	Date of issue	Sheet 3/10



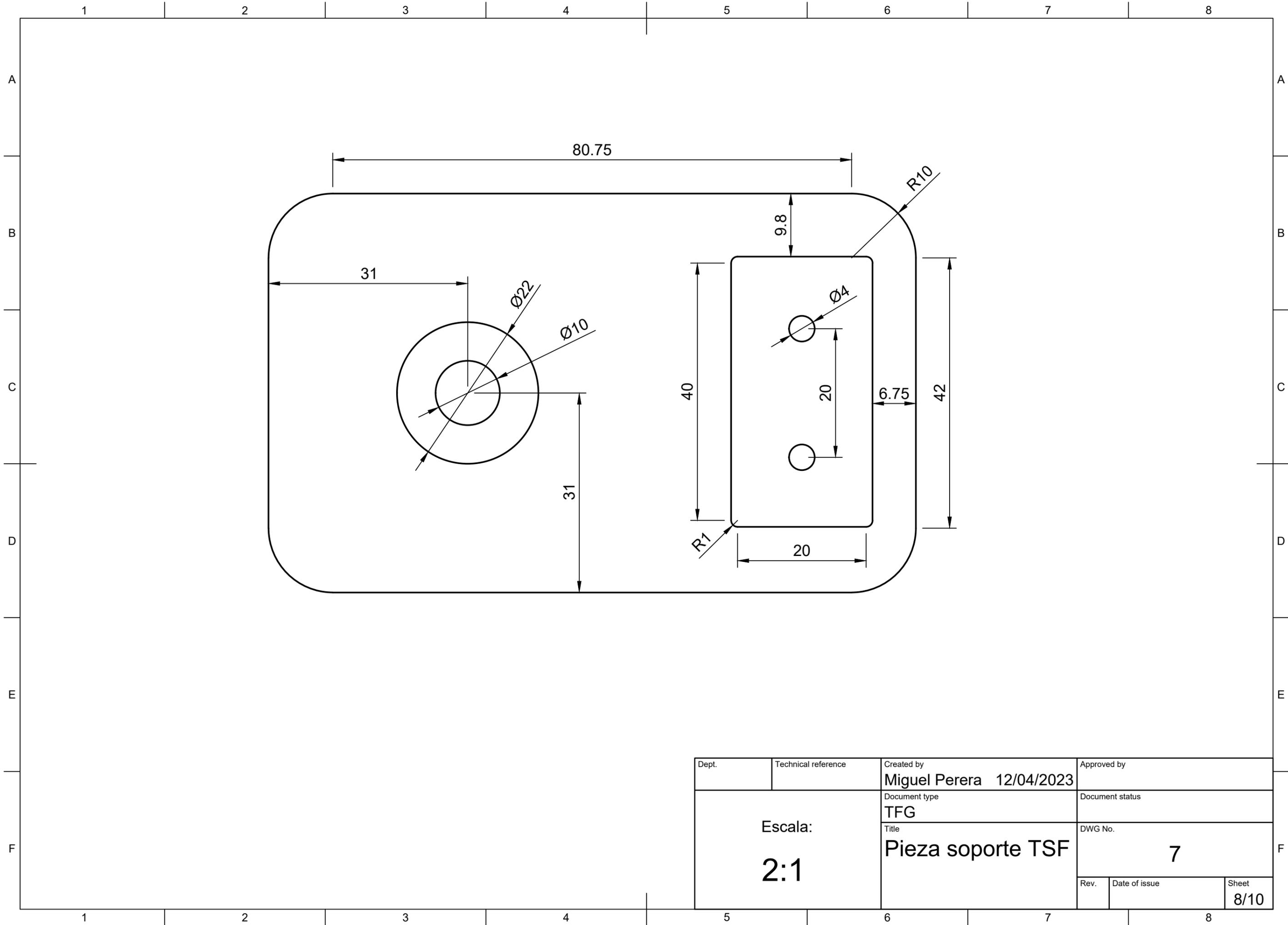
Dept.	Technical reference	Created by Miguel Perera 11/04/2023	Approved by
Escala: 5:4		Document type TFG	Document status
		Title Pieza soporte TSF	DWG No. 3
		Rev.	Date of issue
		Sheet 4/10	



Dept.	Technical reference	Created by Miguel Perera 12/04/2023	Approved by
Escala: 2:1		Document type TFG	Document status
		Title Pieza soporte perfil aluminio Z	DWG No. 4
	Rev.	Date of issue	Sheet 5/10



Dept.	Technical reference	Created by Miguel Perera 12/04/2023	Approved by
Escala: 2:1		Document type TFG	Document status
		Title Pieza soporte de la fijación del eje Z	DWG No. 5
		Rev.	Date of issue
			Sheet 6/10



Dept.	Technical reference	Created by Miguel Perera 12/04/2023	Approved by
Escala: 2:1		Document type TFG	Document status
		Title Pieza soporte TSF	DWG No. 7
	Rev.	Date of issue	Sheet 8/10

11. Presupuesto

Concepto	Unidades	Precio	Subtotal	Total
Costes Materiales				275,73 €
Componentes Electrónicos				135,31 €
Placa Arduino Mega 2560	1	49,95 €	49,95 €	
Módulo Ramps 1.4	1	14,95 €	14,95 €	
Interruptores tipo endstops	3	1,95 €	5,85 €	
Drivers modelo A4988	3	4,50 €	13,50 €	
Motores paso paso	2	10,13 €	20,26 €	
Motores paso paso	2	15,40 €	30,80 €	
Impresión 3D				20,00 €
Filamento	1	20,00 €	20,00 €	
Otros Elementos				120,42 €
V-Slot 2040	2	20,33 €	40,67 €	
Tornillo sin fin	2	11,55 €	23,10 €	
Tornillo sin fin	1	10,50 €	10,50 €	
Tornillo sin fin	1	7,27 €	7,27 €	
Tornillería 1	16	0,09 €	1,41 €	
Tornillería 2	16	0,09 €	1,41 €	
Tornillería 3	8	0,09 €	0,70 €	
Rodamientos	4	0,88 €	3,52 €	
Ruedas	16	1,99 €	31,84 €	
Costes de Ejecución				6.152,00 €
Impresión 3D				152,00 €
Horas de Impresión 3D	40	3,80 €	152,00 €	
Trabajo de ingeniería				6.000,00 €
Horas de trabajo ingeniero técnico industrial	300	20,00 €	6.000,00 €	
Concepto				
Costes Materiales				275,73 €
Costes de Ejecución				6.152,00 €
Total de Costes				6.427,73 €
Gastos Generales 13%				835,60 €
Beneficio Industrial 6%				385,66 €
Suma GC y BI				1.221,27 €
Coste Estimado				7.648,99 €
Igic 7%				535,43 €
Coste Total del Proyecto				8.184,42 €

EL COSTE TOTAL DEL PROYECTO ES DE OCHO MIL CIENTO OCHENTA Y CUATRO EUROS CON CUARENTA Y DOS CENTIMOS.

12. Conclusiones.

Durante la realización de este trabajo, se han alcanzado los objetivos inicialmente propuestos, diseñar una CNC y dotarla de la electrónica y software necesarios para su puesta en marcha.

Para lograr los objetivos, se ha realizado un estudio de mercado, el diseño de la CNC en un software de modelado 3D, la búsqueda de los materiales empleados, la electrónica y el software que se emplearía en la máquina posteriormente.

Una vez se tenía todo lo necesario, se ha realizado el montaje de la máquina cartesiana y dotado de la electrónica y software, dando como resultado una máquina CNC completamente operativa.

Por último indicar que el diseño de esta máquina CNC ha sido diseñada de forma que el proceso de escalarla sea lo más sencillo posible abriendo un abanico de posibilidades a las tareas que se le pueden asignar.

13. Conclusions.

While carrying out this work, the initially proposed objectives have been achieved, designing a CNC and providing it with the electronics and software necessary for its start-up.

To achieve the objectives, a market study has been carried out, as well as the design of the CNC in a 3D modelling software, the search for the adequate materials, the electronics and the software that would be used later in the machine.

Once all the needed components were available, the cartesian machine was assembled and equipped with electronics and software, resulting in a fully operational CNC machine.

Finally, it should be mentioned that the design of this CNC machine has been done in such a way that the process of scaling results as simple as possible, opening up a range of possibilities for the tasks that can be assigned to the machine.

14. Bibliografía.

- [1]<https://www.inter2000mecanizados.com/post/el-mecanizado-cnc-qu%C3%A9-es-y-c%C3%B3mo-ha-influido-en-la-industria-del-autom%C3%B3vil#:~:text=El%20mecanizado%20CN C%20puede%20entenderse,a%20esta%20tecnolog%C3%ADa%20de%20mecanizado.>
- [2]<https://www.inter2000mecanizados.com/post/ventajas-y-desventajas-de-una-maquina-cnc>
- [3]<https://guide.directindustry.com/es/que-robot-industrial-elegir/#:~:text=Los%20robots%20c artesianos%20solo%20se,toda%20la%20zona%20de%20trabajo.>
- [4]https://en.wikipedia.org/wiki/Numerical_control#Positioning_control_system
- [5]<https://www.iso.org/standard/34608.html>
- [6]<https://automatizacionindustrial360.com/historia/#:~:text=La%20historia%20de%20la%20 automatizaci%C3%B3n%20industrial%20comienza%20con%20las%20empresas,autom%C 3%A1tico%20controlado%20por%20tarjetas%20perforadas>
- [7]<https://www.stanser.com/conoce-ultimos-avances-maquinas-cnc/>
- [8]<https://www.sicma21.com/robots-industriales-tecnologia-y-aplicaciones/>
- [9]https://webshop.robotics.abb.com/es/robots.html?rs_collaboration_n=Industrial&gad=1&gclid=CjwKCAjw9pGjBhB-EiwAa5jl3BlauKAoPLInD_hK7brTYmKeAzsMz3YA_OyAtdRIhlgJNkqzmvaOVRoCp8sQAvD_BwE
- [10]<https://www.yaskawa.es/productos/robots>
- [11]<https://www.kuka.com/es-es/productos-servicios/sistemas-de-robot/robot-industrial>
- [12]https://es.wikipedia.org/wiki/Cinem%C3%A1tica_directa#:~:text=Se%20denomina%20cn em%C3%A1tica%20directa%20a,las%20articulaciones%20de%20la%20estructura.
- [13]https://es.wikipedia.org/wiki/Cinem%C3%A1tica_inversa#:~:text=En%20Rob%C3%B3tic a%20C%20la%20Cinem%C3%A1tica%20inversa,ubique%20en%20una%20posici%C3%B3n %20concreta.
- [14]<https://www.zikodrive.com/es/ufags/whats-maxima-velocidad-i-can-get-stepper-motor/>
- [15]<https://lulzbot.com/store/taz-6>
- [16]<https://www.3dnatives.com/es/guia-filamento-pla-en-la-impresion-3d-190820192/#!>
- [17]<https://store.arduino.cc/products/arduino-mega-2560-rev3>
- [18]<http://domoticx.com/arduino-mega-shield-ramps/>

[19]<https://www.diarioelectronicohoy.com/blog/descripcion-del-driver-a4988>

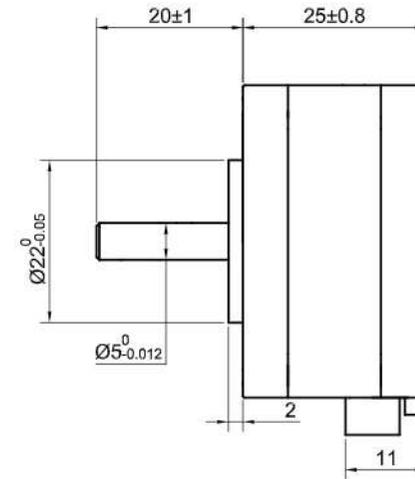
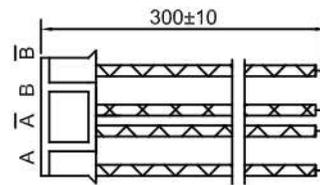
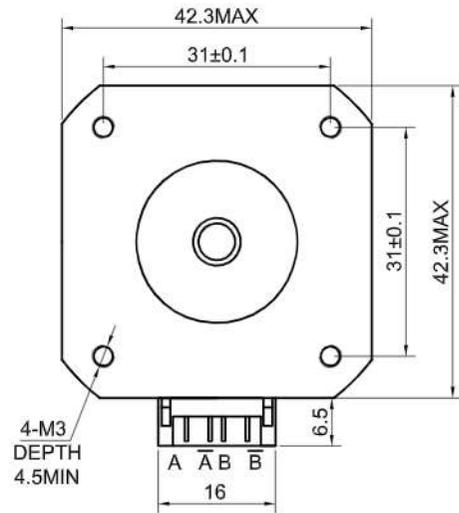
[20]<https://www.faulhaber.com/it/know-how/tutorials/stepper-motor-tutorial-microstepping-my-hs-and-realities/>

[21]<https://zikodrive.com/es/uFAQS/microstepping-puede-utilizado-mejora-stepper-motor-funcionamiento/#:~:text=Microstepping%20es%20donde%20se%20conduce,del%20motor%20de%20pasos.>

[22]https://linuxcnc.org/docs/html/gcode/overview_es.html

[23]http://tavmjong.free.fr/INKSCAPE/MANUAL_v15/html_es/index.html

15. Anexos.

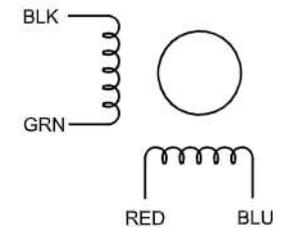


SPECIFICATION	CONNECTION	BIPOLAR
AMPS/PHASE		0.70
RESISTANCE/PHASE(Ohms)@25°C		4.20±10%
INDUCTANCE/PHASE(mH)@1KHz		5.50±20%
HOLDING TORQUE(Nm)[lb-in]		0.13[1.15]
STEP ANGLE(°)		1.80
STEP ACCURACY(NON-ACCUM)		±5.00%
ROTOR INERTIA(g-cm ²)		20.00
WEIGHT(Kg)[lb]		0.18[0.40]
TEMPERATURE RISE:MAX.80°C (MOTOR 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	
2	A\ —	GRN	
3	B —	RED	
4	B\ —	BLU	

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

STEP	A	B	A\	B\		CCW
1	+	+	-	-		
2	-	+	+	-		
3	-	-	+	+		
4	+	-	-	+		

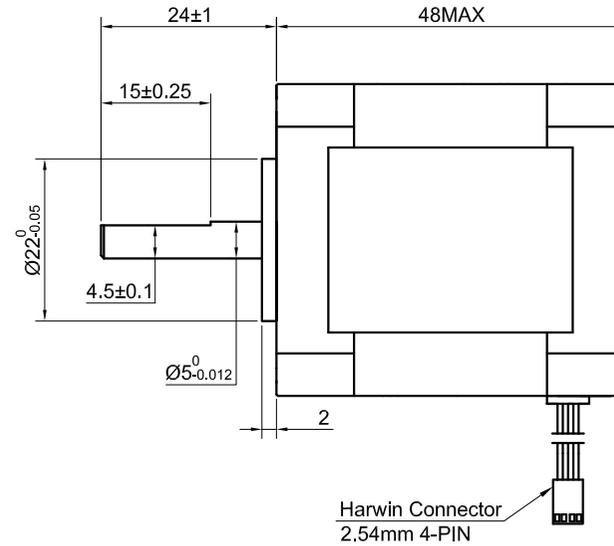
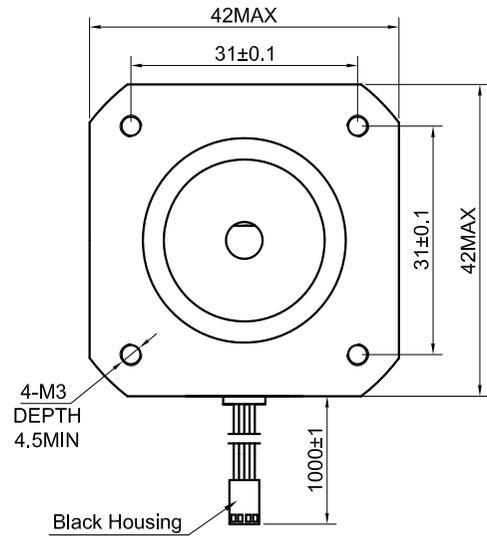


STEPPERONLINE®

APVD		8.18.2018
CHKD		
1:1	DRN	
SCALE	SIGNATURE	DATE

STEPPER MOTOR

17HS10-0704S

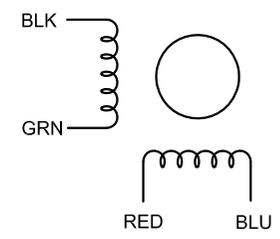


SPECIFICATION	CONNECTION	BIPOLAR
AMPS/PHASE		2.00
RESISTANCE/PHASE(Ohms)@25°C		1.40±10%
INDUCTANCE/PHASE(mH)@1KHz		3.00±20%
HOLDING TORQUE(Nm)[lb-in]		0.59[5.22]
STEP ANGLE(°)		1.80
STEP ACCURACY(NON-ACCUM)		±5.00%
ROTOR INERTIA(g-cm ²)		82.00
WEIGHT(Kg)[lb]		0.40[0.88]
TEMPERATURE RISE:MAX.80°C (MOTOR 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\		CCW
1	+	+	-	-	↓	↑
2	-	+	+	-		
3	-	-	+	+	↑	↓
4	+	-	-	+		



APVD		8.18.2018
CHKD		
1:1	DRN	
SCALE	SIGNATURE	DATE

STEPPER MOTOR

17HS19-2004S1



A4988

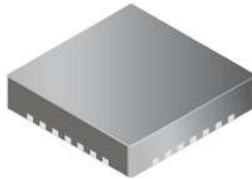
DMOS Microstepping Driver with Translator and Overcurrent Protection

Features and Benefits

- Low $R_{DS(ON)}$ outputs
- Automatic current decay mode detection/selection
- Mixed and Slow current decay modes
- Synchronous rectification for low power dissipation
- Internal UVLO
- Crossover-current protection
- 3.3 and 5 V compatible logic supply
- Thermal shutdown circuitry
- Short-to-ground protection
- Shorted load protection
- Five selectable step modes: full, $1/2$, $1/4$, $1/8$, and $1/16$

Package:

28-contact QFN
with exposed thermal pad
5 mm × 5 mm × 0.90 mm
(ET package)



Approximate size

Description

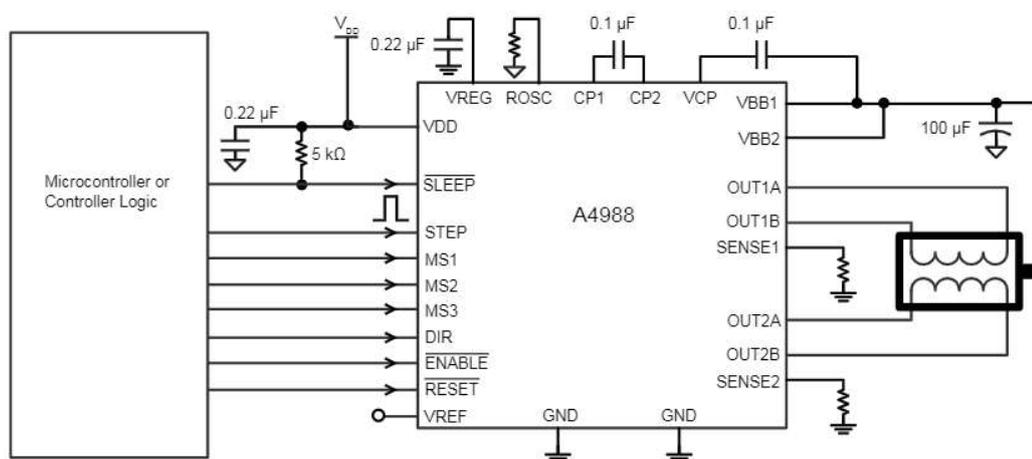
The A4988 is a complete microstepping motor driver with built-in translator for easy operation. It is designed to operate bipolar stepper motors in full-, half-, quarter-, eighth-, and sixteenth-step modes, with an output drive capacity of up to 35 V and ± 2 A. The A4988 includes a fixed off-time current regulator which has the ability to operate in Slow or Mixed decay modes.

The translator is the key to the easy implementation of the A4988. Simply inputting one pulse on the STEP input drives the motor one microstep. There are no phase sequence tables, high frequency control lines, or complex interfaces to program. The A4988 interface is an ideal fit for applications where a complex microprocessor is unavailable or is overburdened.

During stepping operation, the chopping control in the A4988 automatically selects the current decay mode, Slow or Mixed. In Mixed decay mode, the device is set initially to a fast decay for a proportion of the fixed off-time, then to a slow decay for the remainder of the off-time. Mixed decay current control results in reduced audible motor noise, increased step accuracy, and reduced power dissipation.

Continued on the next page...

Typical Application Diagram



A4988

DMOS Microstepping Driver with Translator and Overcurrent Protection

Description (continued)

Internal synchronous rectification control circuitry is provided to improve power dissipation during PWM operation. Internal circuit protection includes: thermal shutdown with hysteresis, undervoltage lockout (UVLO), and crossover-current protection. Special power-on sequencing is not required.

The A4988 is supplied in a surface mount QFN package (ES), 5 mm × 5 mm, with a nominal overall package height of 0.90 mm and an exposed pad for enhanced thermal dissipation. It is lead (Pb) free (suffix -T), with 100% matte tin plated leadframes.

Selection Guide

Part Number	Package	Packing
A4988SETTR-T	28-contact QFN with exposed thermal pad	1500 pieces per 7-in. reel

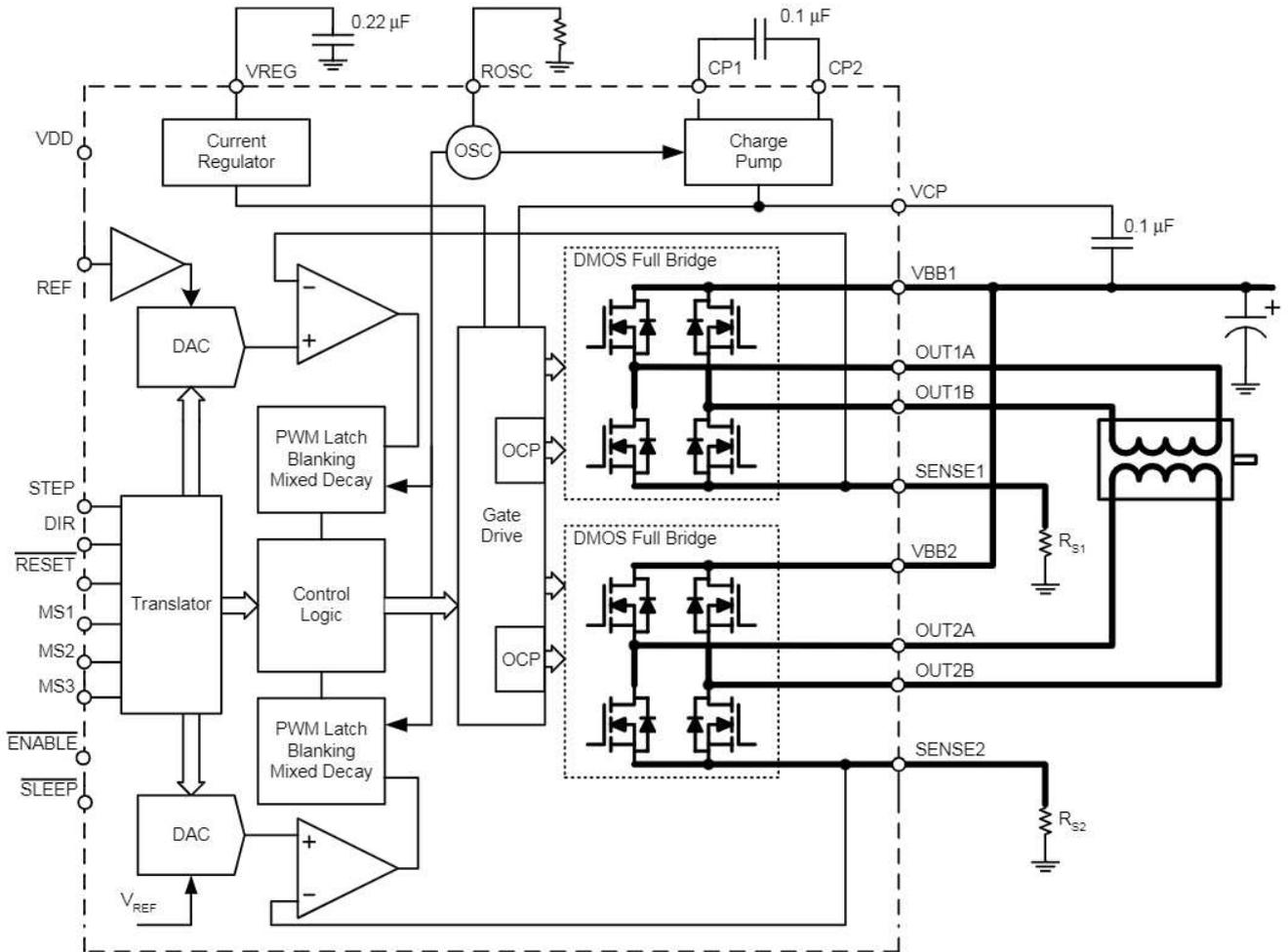
Absolute Maximum Ratings

Characteristic	Symbol	Notes	Rating	Units
Load Supply Voltage	V_{BB}		35	V
Output Current	I_{OUT}		±2	A
Logic Input Voltage	V_{IN}		-0.3 to 5.5	V
Logic Supply Voltage	V_{DD}		-0.3 to 5.5	V
VBBx to OUTx			35	V
Sense Voltage	V_{SENSE}		0.5	V
Reference Voltage	V_{REF}		5.5	V
Operating Ambient Temperature	T_A	Range S	-20 to 85	°C
Maximum Junction	$T_J(max)$		150	°C
Storage Temperature	T_{stg}		-55 to 150	°C

A4988

DMOS Microstepping Driver with Translator and Overcurrent Protection

Functional Block Diagram



A4988***DMOS Microstepping Driver with Translator
and Overcurrent Protection*****ELECTRICAL CHARACTERISTICS¹** at $T_A = 25^\circ\text{C}$, $V_{BB} = 35\text{ V}$ (unless otherwise noted)

Characteristics	Symbol	Test Conditions	Min.	Typ. ²	Max.	Units
Output Drivers						
Load Supply Voltage Range	V_{BB}	Operating	8	–	35	V
Logic Supply Voltage Range	V_{DD}	Operating	3.0	–	5.5	V
Output On Resistance	R_{DSON}	Source Driver, $I_{OUT} = -1.5\text{ A}$	–	320	430	m Ω
		Sink Driver, $I_{OUT} = 1.5\text{ A}$	–	320	430	m Ω
Body Diode Forward Voltage	V_F	Source Diode, $I_F = -1.5\text{ A}$	–	–	1.2	V
		Sink Diode, $I_F = 1.5\text{ A}$	–	–	1.2	V
Motor Supply Current	I_{BB}	$f_{PWM} < 50\text{ kHz}$	–	–	4	mA
		Operating, outputs disabled	–	–	2	mA
Logic Supply Current	I_{DD}	$f_{PWM} < 50\text{ kHz}$	–	–	8	mA
		Outputs off	–	–	5	mA
Control Logic						
Logic Input Voltage	$V_{IN(1)}$		$V_{DD} \times 0.7$	–	–	V
	$V_{IN(0)}$		–	–	$V_{DD} \times 0.3$	V
Logic Input Current	$I_{IN(1)}$	$V_{IN} = V_{DD} \times 0.7$	–20	<1.0	20	μA
	$I_{IN(0)}$	$V_{IN} = V_{DD} \times 0.3$	–20	<1.0	20	μA
Microstep Select	R_{MS1}	MS1 pin	–	100	–	k Ω
	R_{MS2}	MS2 pin	–	50	–	k Ω
	R_{MS3}	MS3 pin	–	100	–	k Ω
Logic Input Hysteresis	$V_{HYS(IN)}$	As a % of V_{DD}	5	11	19	%
Blank Time	t_{BLANK}		0.7	1	1.3	μs
Fixed Off-Time	t_{OFF}	OSC = V_{DD} or GND	20	30	40	μs
		$R_{OSC} = 25\text{ k}\Omega$	23	30	37	μs
Reference Input Voltage Range	V_{REF}		0	–	4	V
Reference Input Current	I_{REF}		–3	0	3	μA
Current Trip-Level Error ³	err_I	$V_{REF} = 2\text{ V}$, % $I_{TripMAX} = 38.27\%$	–	–	± 15	%
		$V_{REF} = 2\text{ V}$, % $I_{TripMAX} = 70.71\%$	–	–	± 5	%
		$V_{REF} = 2\text{ V}$, % $I_{TripMAX} = 100.00\%$	–	–	± 5	%
Crossover Dead Time	t_{DT}		100	475	800	ns
Protection						
Overcurrent Protection Threshold	I_{OCPST}		2.1	–	–	A
Thermal Shutdown Temperature	T_{TSD}		–	165	–	$^\circ\text{C}$
Thermal Shutdown Hysteresis	T_{TSDHYS}		–	15	–	$^\circ\text{C}$
VDD Undervoltage Lockout	V_{DDUVLO}	V_{DD} rising	2.7	2.8	2.9	V
VDD Undervoltage Hysteresis	$V_{DDUVLOHYS}$		–	90	–	mV

¹For input and output current specifications, negative current is defined as coming out of (sourcing) the specified device pin.

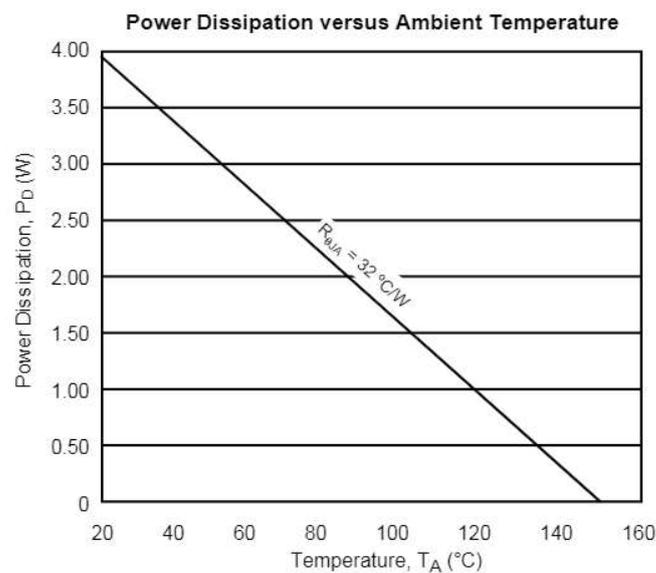
²Typical data are for initial design estimations only, and assume optimum manufacturing and application conditions. Performance may vary for individual units, within the specified maximum and minimum limits.

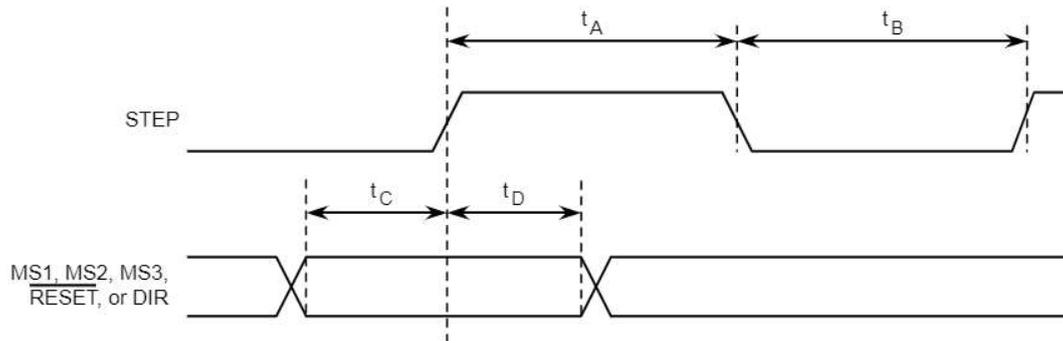
³ $V_{ERR} = [(V_{REF}/8) - V_{SENSE}] / (V_{REF}/8)$.

A4988***DMOS Microstepping Driver with Translator
and Overcurrent Protection*****THERMAL CHARACTERISTICS**

Characteristic	Symbol	Test Conditions*	Value	Units
Package Thermal Resistance	$R_{\theta JA}$	Four-layer PCB, based on JEDEC standard	32	$^{\circ}\text{C}/\text{W}$

*Additional thermal information available on Allegro Web site.



A4988***DMOS Microstepping Driver with Translator
and Overcurrent Protection***

Time Duration	Symbol	Typ.	Unit
STEP minimum, HIGH pulse width	t_A	1	μs
STEP minimum, LOW pulse width	t_B	1	μs
Setup time, input change to STEP	t_C	200	ns
Hold time, input change to STEP	t_D	200	ns

Figure 1. Logic Interface Timing Diagram

Table 1. Microstepping Resolution Truth Table

MS1	MS2	MS3	Microstep Resolution	Excitation Mode
L	L	L	Full Step	2 Phase
H	L	L	Half Step	1-2 Phase
L	H	L	Quarter Step	W1-2 Phase
H	H	L	Eighth Step	2W1-2 Phase
H	H	H	Sixteenth Step	4W1-2 Phase

A4988

DMOS Microstepping Driver with Translator and Overcurrent Protection

Functional Description

Device Operation. The A4988 is a complete microstepping motor driver with a built-in translator for easy operation with minimal control lines. It is designed to operate bipolar stepper motors in full-, half-, quarter-, eighth, and sixteenth-step modes. The currents in each of the two output full-bridges and all of the N-channel DMOS FETs are regulated with fixed off-time PWM (pulse width modulated) control circuitry. At each step, the current for each full-bridge is set by the value of its external current-sense resistor (R_{S1} and R_{S2}), a reference voltage (V_{REF}), and the output voltage of its DAC (which in turn is controlled by the output of the translator).

At power-on or reset, the translator sets the DACs and the phase current polarity to the initial Home state (shown in figures 8 through 12), and the current regulator to Mixed Decay Mode for both phases. When a step command signal occurs on the STEP input, the translator automatically sequences the DACs to the next level and current polarity. (See table 2 for the current-level sequence.) The microstep resolution is set by the combined effect of the MSx inputs, as shown in table 1.

When stepping, if the new output levels of the DACs are lower than their previous output levels, then the decay mode for the active full-bridge is set to Mixed. If the new output levels of the DACs are higher than or equal to their previous levels, then the decay mode for the active full-bridge is set to Slow. This automatic current decay selection improves microstepping performance by reducing the distortion of the current waveform that results from the back EMF of the motor.

Microstep Select (MSx). The microstep resolution is set by the voltage on logic inputs MSx, as shown in table 1. The MS1 and MS3 pins have a 100 k Ω pull-down resistance, and the MS2 pin has a 50 k Ω pull-down resistance. When changing the step mode the change does not take effect until the next STEP rising edge.

If the step mode is changed without a translator reset, and absolute position must be maintained, it is important to change the step mode at a step position that is common to both step modes in order to avoid missing steps. When the device is powered down, or reset due to TSD or an over current event the translator is set to the home position which is by default common to all step modes.

Mixed Decay Operation. The bridge operates in Mixed decay mode, at power-on and reset, and during normal running according to the ROSC configuration and the step sequence, as shown in figures 8 through 12. During Mixed decay, when the trip point is reached, the A4988 initially goes into a fast decay mode for 31.25% of the off-time, t_{OFF} . After that, it switches to Slow decay mode for the remainder of t_{OFF} . A timing diagram for this feature appears on the next page.

Typically, mixed decay is only necessary when the current in the winding is going from a higher value to a lower value as determined by the state of the translator. For most loads automatically-selected mixed decay is convenient because it minimizes ripple when the current is rising and prevents missed steps when the current is falling. For some applications where microstepping at very low speeds is necessary, the lack of back EMF in the winding causes the current to increase in the load quickly, resulting in missed steps. This is shown in figure 2. By pulling the ROSC pin to ground, mixed decay is set to be active 100% of the time, for both rising and falling currents, and prevents missed steps as shown in figure 3. If this is not an issue, it is recommended that automatically-selected mixed decay be used, because it will produce reduced ripple currents. Refer to the Fixed Off-Time section for details.

Low Current Microstepping. Intended for applications where the minimum on-time prevents the output current from regulating to the programmed current level at low current steps. To prevent this, the device can be set to operate in Mixed decay mode on both rising and falling portions of the current waveform. This feature is implemented by shorting the ROSC pin to ground. In this state, the off-time is internally set to 30 μ s.

Reset Input (\overline{RESET}). The \overline{RESET} input sets the translator to a predefined Home state (shown in figures 8 through 12), and turns off all of the FET outputs. All STEP inputs are ignored until the \overline{RESET} input is set to high.

Step Input (STEP). A low-to-high transition on the STEP input sequences the translator and advances the motor one increment. The translator controls the input to the DACs and the direc-

A4988

DMOS Microstepping Driver with Translator and Overcurrent Protection

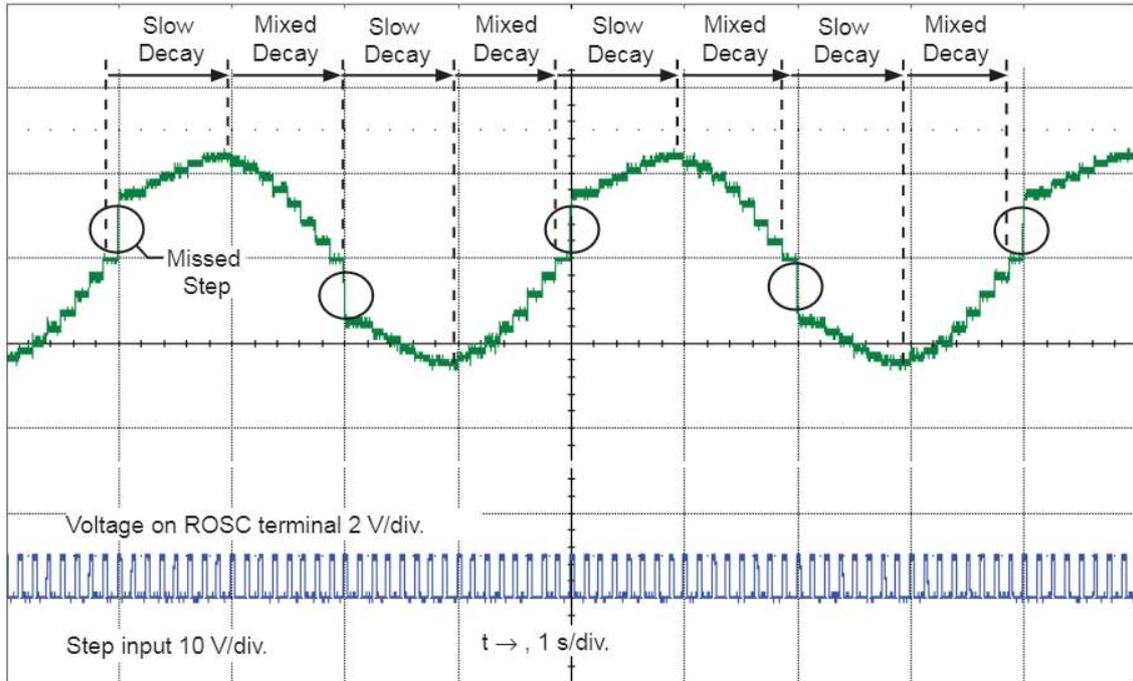


Figure 2. Missed steps in low-speed microstepping

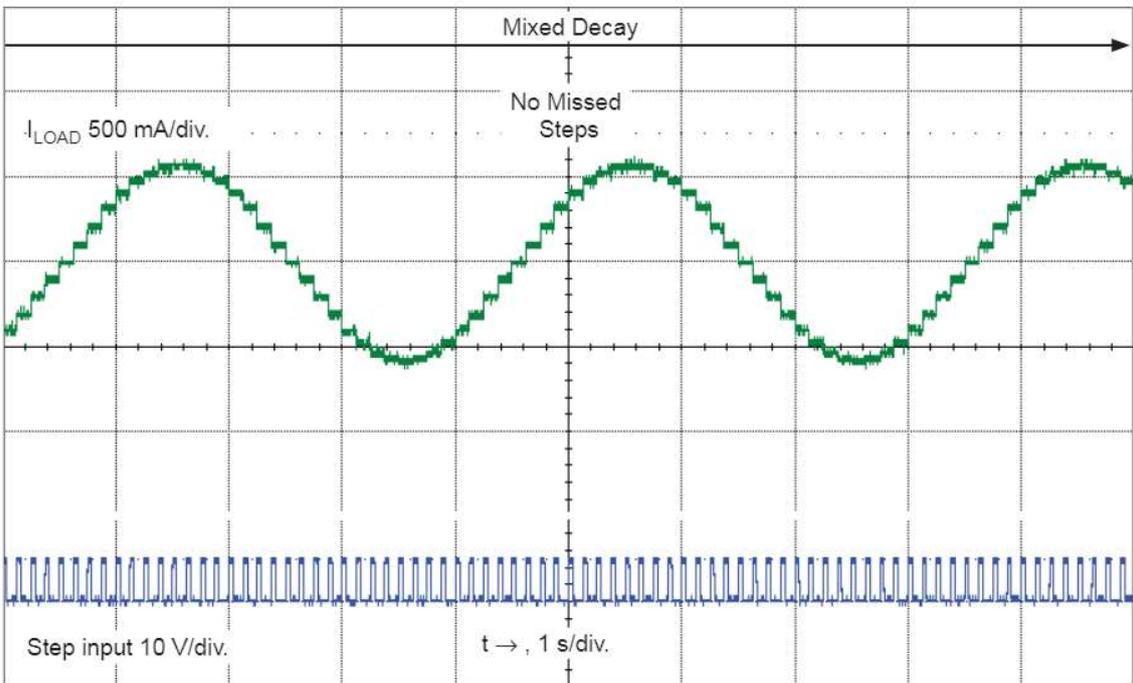


Figure 3. Continuous stepping using automatically-selected mixed stepping (ROSC pin grounded)

A4988

DMOS Microstepping Driver with Translator and Overcurrent Protection

tion of current flow in each winding. The size of the increment is determined by the combined state of the MSx inputs.

Direction Input (DIR). This determines the direction of rotation of the motor. Changes to this input do not take effect until the next STEP rising edge.

Internal PWM Current Control. Each full-bridge is controlled by a fixed off-time PWM current control circuit that limits the load current to a desired value, I_{TRIP} . Initially, a diagonal pair of source and sink FET outputs are enabled and current flows through the motor winding and the current sense resistor, R_{Sx} . When the voltage across R_{Sx} equals the DAC output voltage, the current sense comparator resets the PWM latch. The latch then turns off the appropriate source driver and initiates a fixed off time decay mode

The maximum value of current limiting is set by the selection of R_{Sx} and the voltage at the VREF pin. The transconductance function is approximated by the maximum value of current limiting, $I_{TRIPMAX}$ (A), which is set by

$$I_{TRIPMAX} = V_{REF} / (8 \times R_S)$$

where R_S is the resistance of the sense resistor (Ω) and V_{REF} is the input voltage on the REF pin (V).

The DAC output reduces the V_{REF} output to the current sense comparator in precise steps, such that

$$I_{trip} = (\%I_{TRIPMAX} / 100) \times I_{TRIPMAX}$$

(See table 2 for $\%I_{TRIPMAX}$ at each step.)

It is critical that the maximum rating (0.5 V) on the SENSE1 and SENSE2 pins is not exceeded.

Fixed Off-Time. The internal PWM current control circuitry uses a one-shot circuit to control the duration of time that the DMOS FETs remain off. The off-time, t_{OFF} , is determined by the ROSC terminal. The ROSC terminal has three settings:

- ROSC tied to VDD — off-time internally set to 30 μ s, decay mode is automatic Mixed decay except when in full step where decay mode is set to Slow decay
- ROSC tied directly to ground — off-time internally set to 30 μ s, current decay is set to Mixed decay for both increasing and decreasing currents, except in full step where decay mode is set to Slow decay. (See Low Current Microstepping section.)

- ROSC through a resistor to ground — off-time is determined by the following formula, the decay mode is automatic Mixed decay for all step modes.

$$t_{OFF} \approx R_{OSC} / 825$$

Where t_{OFF} is in μ s.

Blanking. This function blanks the output of the current sense comparators when the outputs are switched by the internal current control circuitry. The comparator outputs are blanked to prevent false overcurrent detection due to reverse recovery currents of the clamp diodes, and switching transients related to the capacitance of the load. The blank time, t_{BLANK} (μ s), is approximately

$$t_{BLANK} \approx 1 \mu s$$

Shorted-Load and Short-to-Ground Protection.

If the motor leads are shorted together, or if one of the leads is shorted to ground, the driver will protect itself by sensing the overcurrent event and disabling the driver that is shorted, protecting the device from damage. In the case of a short-to-ground, the device will remain disabled (latched) until the SLEEP input goes high or VDD power is removed. A short-to-ground overcurrent event is shown in figure 4.

When the two outputs are shorted together, the current path is through the sense resistor. After the blanking time ($\approx 1 \mu$ s) expires, the sense resistor voltage is exceeding its trip value, due to the overcurrent condition that exists. This causes the driver to go into a fixed off-time cycle. After the fixed off-time expires the driver turns on again and the process repeats. In this condition the driver is completely protected against overcurrent events, but the short is repetitive with a period equal to the fixed off-time of the driver. This condition is shown in figure 5.

During a shorted load event it is normal to observe both a positive and negative current spike as shown in figure 3, due to the direction change implemented by the Mixed decay feature. This is shown in figure 6. In both instances the overcurrent circuitry is protecting the driver and prevents damage to the device.

Charge Pump (CP1 and CP2). The charge pump is used to generate a gate supply greater than that of VBB for driving the source-side FET gates. A 0.1 μ F ceramic capacitor, should be connected between CP1 and CP2. In addition, a 0.1 μ F ceramic capacitor is required between VCP and VBB, to act as a reservoir for operating the high-side FET gates.

Capacitor values should be Class 2 dielectric $\pm 15\%$ maximum, or tolerance R, according to EIA (Electronic Industries Alliance) specifications.

A4988

DMOS Microstepping Driver with Translator and Overcurrent Protection

V_{REG} (VREG). This internally-generated voltage is used to operate the sink-side FET outputs. The VREG pin must be decoupled with a 0.22 μ F ceramic capacitor to ground. V_{REG} is internally monitored. In the case of a fault condition, the FET outputs of the A4988 are disabled.

Capacitor values should be Class 2 dielectric \pm 15% maximum, or tolerance R, according to EIA (Electronic Industries Alliance) specifications.

Enable Input ($\overline{\text{ENABLE}}$). This input turns on or off all of the FET outputs. When set to a logic high, the outputs are disabled. When set to a logic low, the internal control enables the outputs as required. The translator inputs STEP, DIR, and MSx, as well as the internal sequencing logic, all remain active, independent of the $\overline{\text{ENABLE}}$ input state.

Shutdown. In the event of a fault, overtemperature (excess T_J) or an undervoltage (on VCP), the FET outputs of the A4988 are disabled until the fault condition is removed. At power-on, the UVLO (undervoltage lockout) circuit disables the FET outputs and resets the translator to the Home state.

Sleep Mode ($\overline{\text{SLEEP}}$). To minimize power consumption when the motor is not in use, this input disables much of the internal circuitry including the output FETs, current regulator, and charge pump. A logic low on the $\overline{\text{SLEEP}}$ pin puts the A4988 into Sleep mode. A logic high allows normal operation, as well as start-up (at which time the A4988 drives the motor to the Home microstep position). When emerging from Sleep mode, in order to allow the charge pump to stabilize, provide a delay of 1 ms before issuing a Step command.

Mixed Decay Operation. The bridge operates in Mixed Decay mode, depending on the step sequence, as shown in figures 8 through 12. As the trip point is reached, the A4988 initially goes into a fast decay mode for 31.25% of the off-time, t_{OFF} . After that, it switches to Slow Decay mode for the remainder of t_{OFF} . A timing diagram for this feature appears in figure 7.

Synchronous Rectification. When a PWM-off cycle is triggered by an internal fixed-off time cycle, load current recirculates according to the decay mode selected by the control logic. This synchronous rectification feature turns on the appropriate FETs during current decay, and effectively shorts out the body diodes with the low FET $R_{\text{DS(ON)}}$. This reduces power dissipation significantly, and can eliminate the need for external Schottky diodes in many applications. Synchronous rectification turns off when the load current approaches zero (0 A), preventing reversal of the load current.

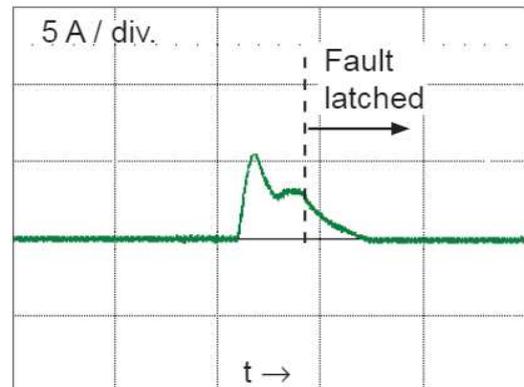


Figure 4. Short-to-ground event

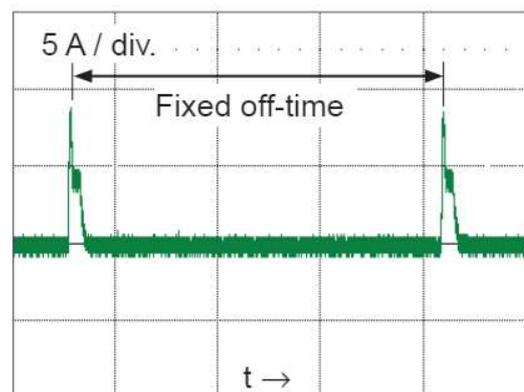


Figure 5. Shorted load (OUTxA → OUTxB) in Slow decay mode

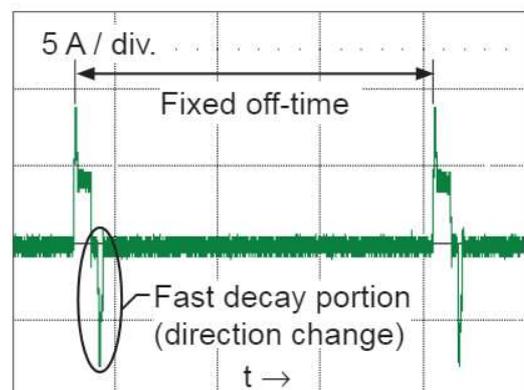
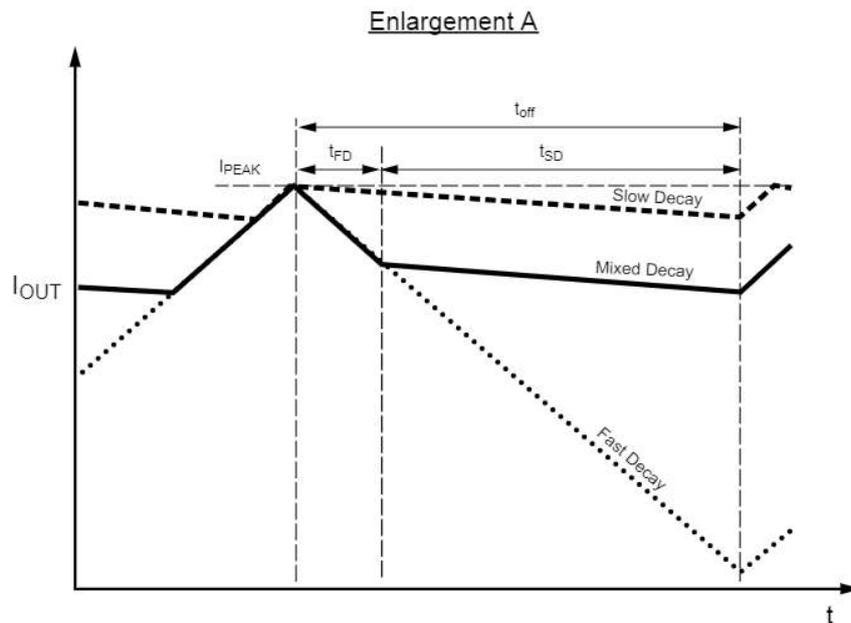
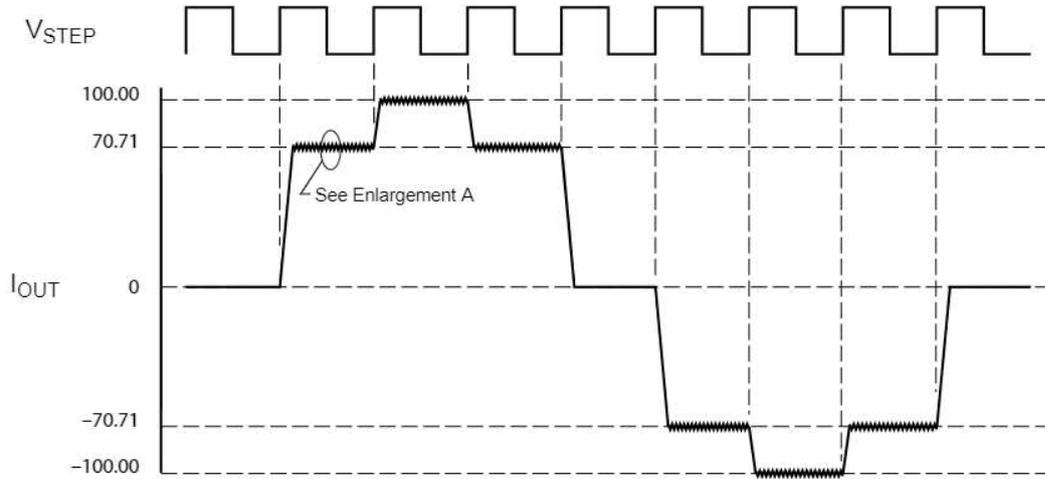


Figure 6. Shorted load (OUTxA → OUTxB) in Mixed decay mode

A4988

DMOS Microstepping Driver with Translator and Overcurrent Protection



Symbol	Characteristic
t_{off}	Device fixed off-time
I_{PEAK}	Maximum output current
t_{SD}	Slow decay interval
t_{FD}	Fast decay interval
I_{OUT}	Device output current

Figure 7. Current Decay Modes Timing Chart

A4988

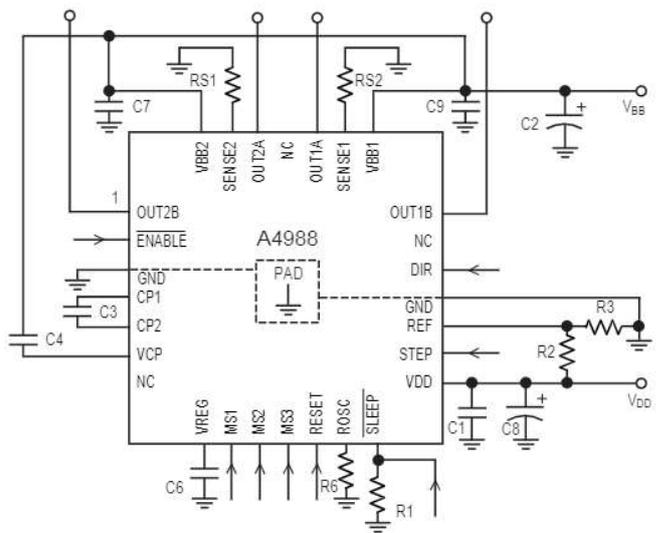
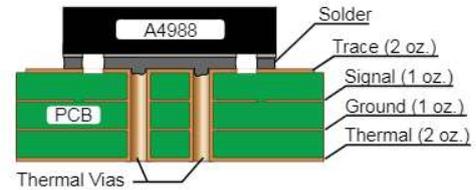
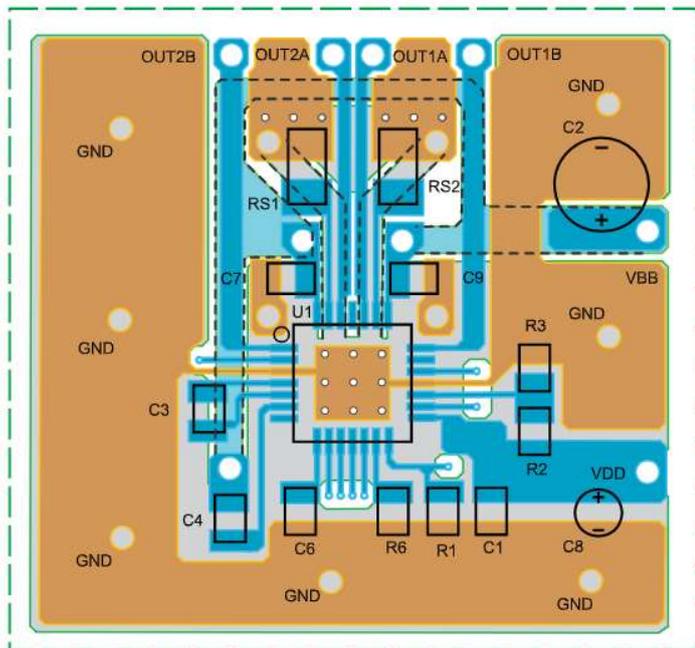
DMOS Microstepping Driver with Translator and Overcurrent Protection

Application Layout

Layout. The printed circuit board should use a heavy ground-plane. For optimum electrical and thermal performance, the A4988 must be soldered directly onto the board. Pins 6, 7, 18, and 19 are internally fused, which provides a path for enhanced thermal dissipation. These pins should be soldered directly to an exposed surface on the PCB that connects to thermal vias are used to transfer heat to other layers of the PCB.

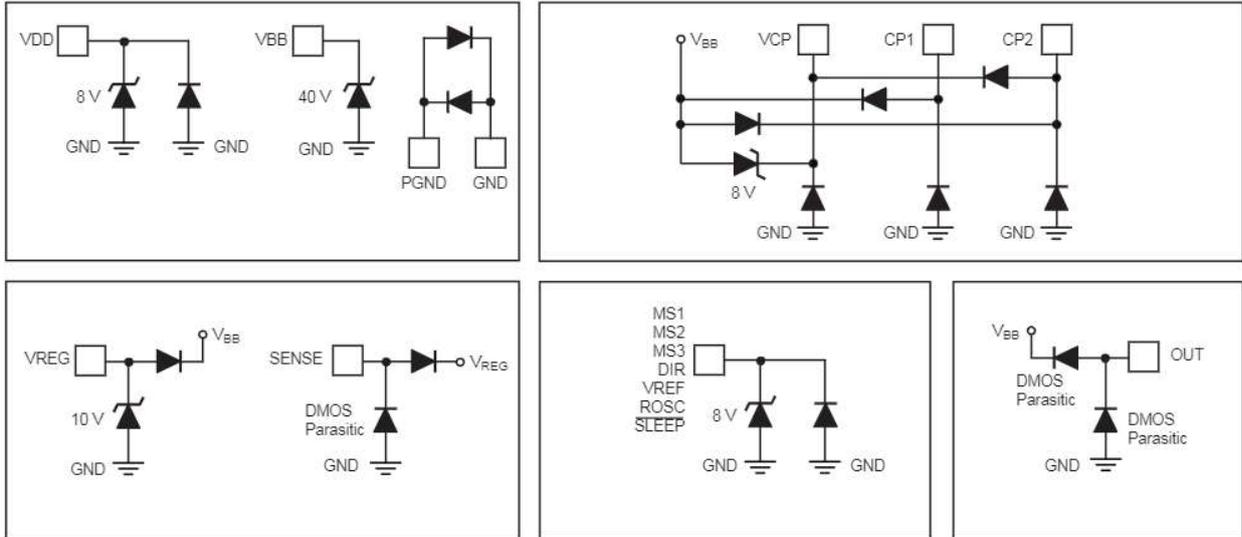
In order to minimize the effects of ground bounce and offset issues, it is important to have a low impedance single-point ground, known as a *star ground*, located very close to the device. By making the connection between the pad and the ground plane directly under the A4988, that area becomes an ideal location for a star ground point. A low impedance ground will prevent ground bounce during high current operation and ensure that the supply voltage remains stable at the input terminal.

The two input capacitors should be placed in parallel, and as close to the device supply pins as possible. The ceramic capacitor (CIN1) should be closer to the pins than the bulk capacitor (CIN2). This is necessary because the ceramic capacitor will be responsible for delivering the high frequency current components. The sense resistors, RSx, should have a very low impedance path to ground, because they must carry a large current while supporting very accurate voltage measurements by the current sense comparators. Long ground traces will cause additional voltage drops, adversely affecting the ability of the comparators to accurately measure the current in the windings. The SENSEx pins have very short traces to the RSx resistors and very thick, low impedance traces directly to the star ground underneath the device. If possible, there should be no other components on the sense circuits.



A4988***DMOS Microstepping Driver with Translator
and Overcurrent Protection***

Pin Circuit Diagrams



A4988

DMOS Microstepping Driver with Translator and Overcurrent Protection

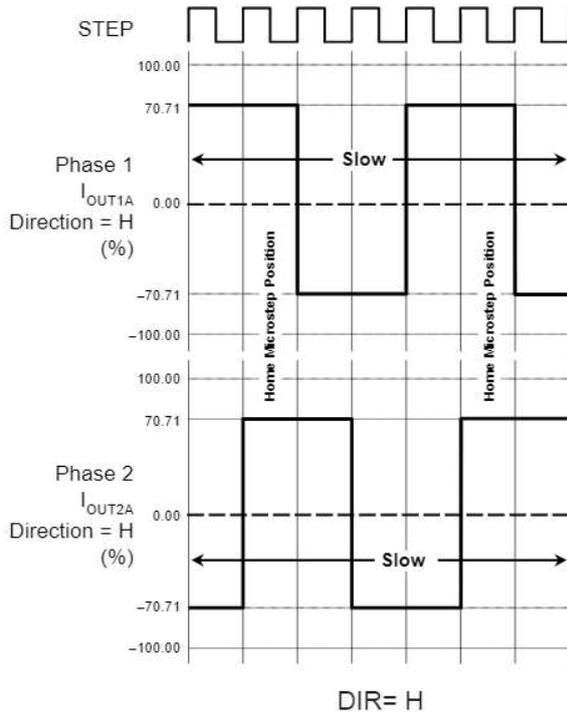


Figure 8. Decay Mode for Full-Step Increments

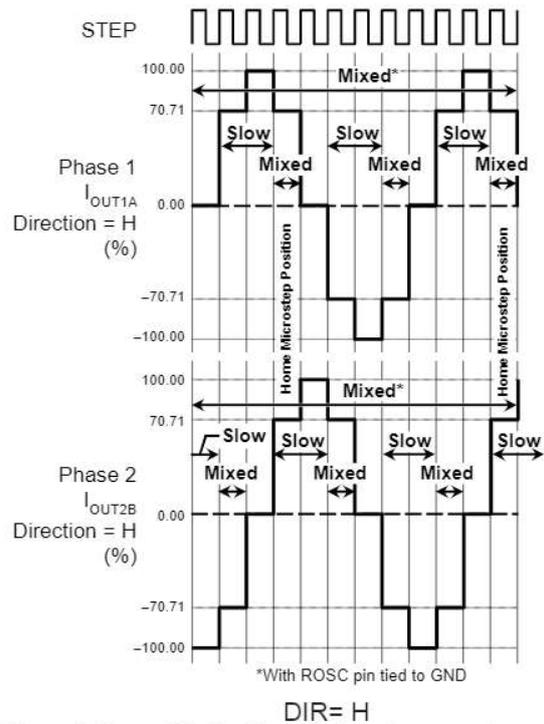


Figure 9. Decay Modes for Half-Step Increments

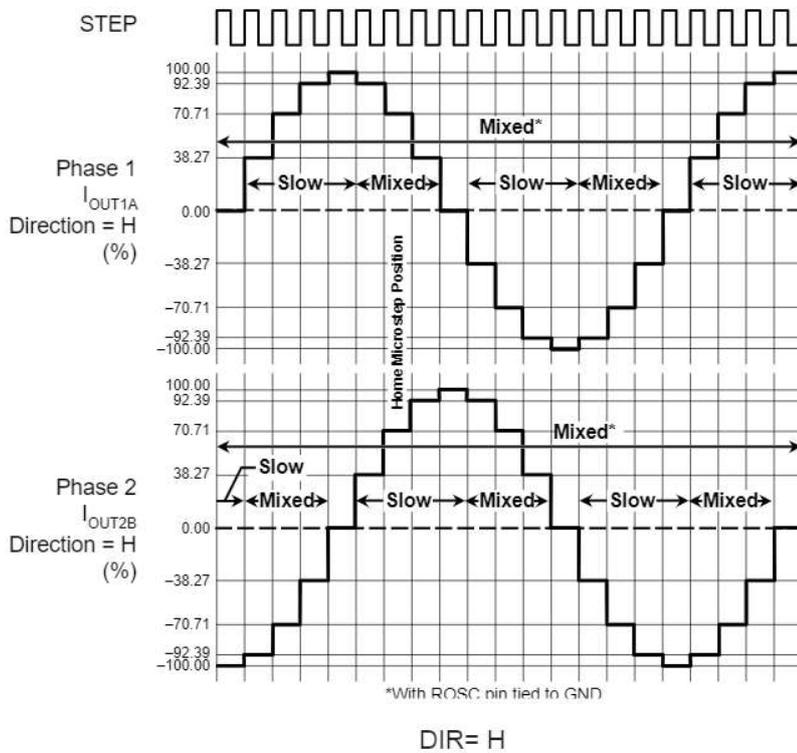


Figure 10. Decay Modes for Quarter-Step Increments

A4988

DMOS Microstepping Driver with Translator and Overcurrent Protection

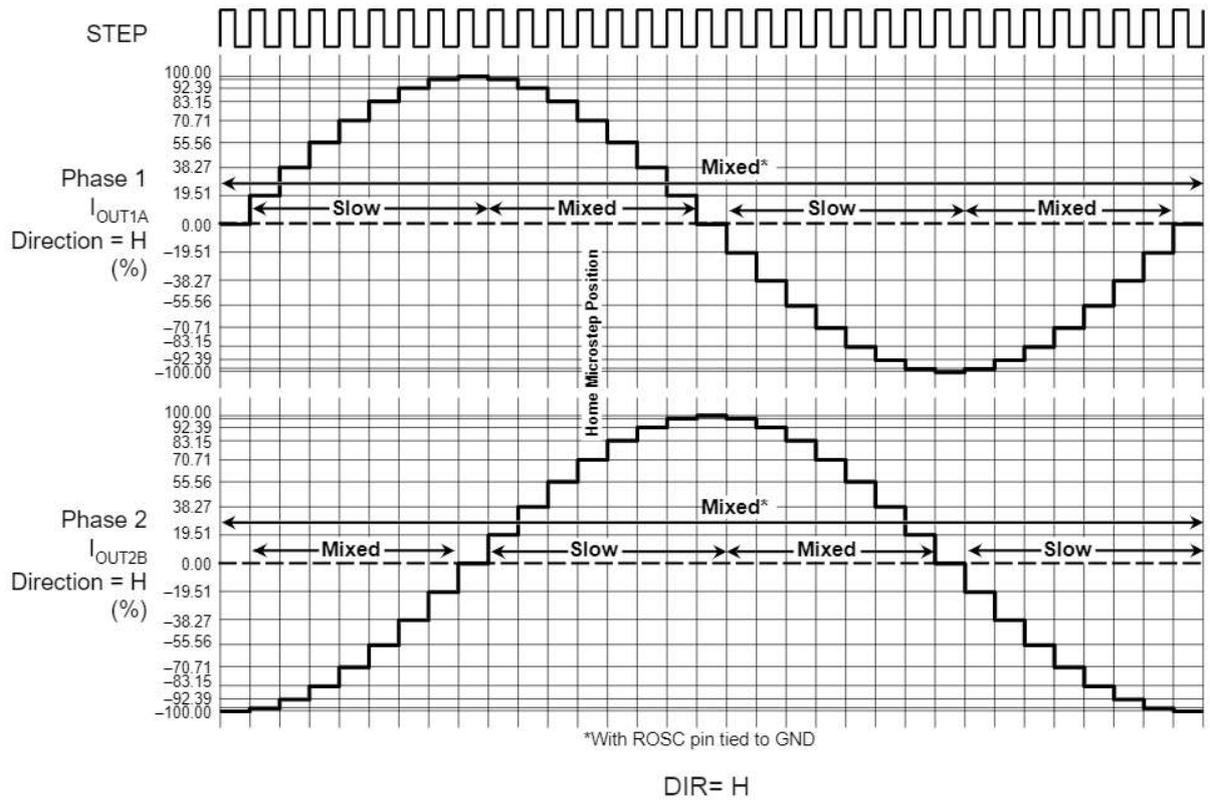


Figure 11. Decay Modes for Eighth-Step Increments

A4988

DMOS Microstepping Driver with Translator and Overcurrent Protection

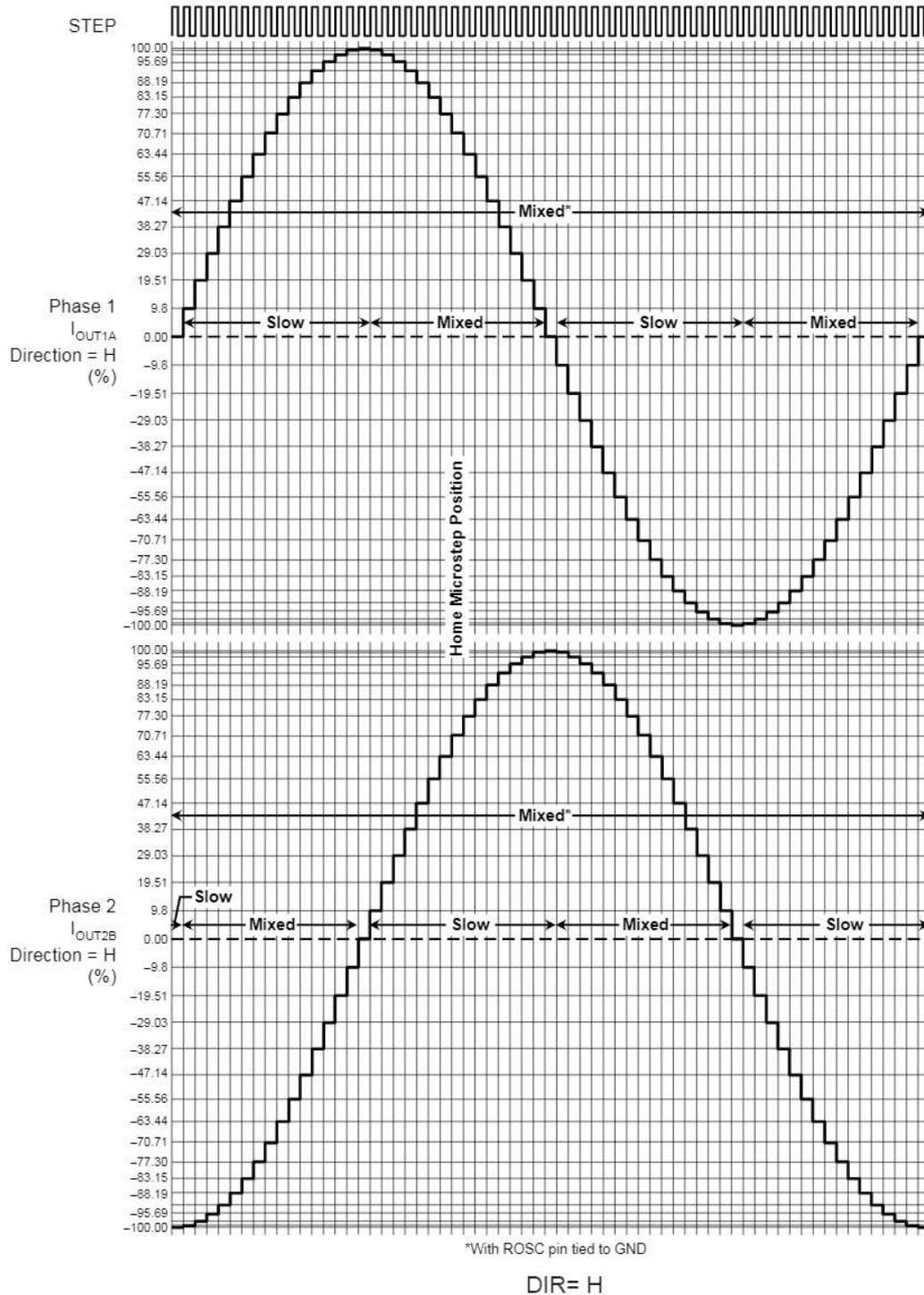


Figure 12. Decay Modes for Sixteenth-Step Increments

A4988***DMOS Microstepping Driver with Translator
and Overcurrent Protection***

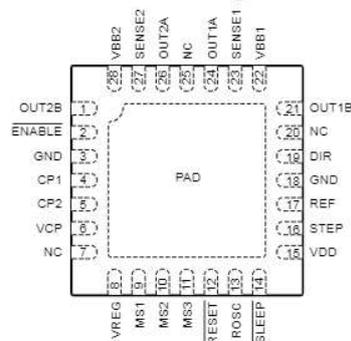
Table 2. Step Sequencing Settings

Home microstep position at Step Angle 45°; DIR = H

Full Step #	Half Step #	1/4 Step #	1/8 Step #	1/16 Step #	Phase 1 Current [% I _{tripMax} (%)	Phase 2 Current [% I _{tripMax} (%)	Step Angle (°)	Full Step #	Half Step #	1/4 Step #	1/8 Step #	1/16 Step #	Phase 1 Current [% I _{tripMax} (%)	Phase 2 Current [% I _{tripMax} (%)	Step Angle (°)	
	1	1	2	1	100.00	0.00	0.0									
				2	99.52	9.80	5.6		5	9	17	33	-100.00	0.00	180.0	
				3	98.08	19.51	11.3					34	-99.52	-9.80	185.6	
			2	4	95.69	29.03	16.9				18	35	-98.08	-19.51	191.3	
				5	92.39	38.27	22.5					36	-95.69	-29.03	196.9	
		2	3	6	88.19	47.14	28.1			10	19	37	-92.39	-38.27	202.5	
				7	83.15	55.56	33.8					38	-88.19	-47.14	208.1	
			4	8	77.30	63.44	39.4					39	-83.15	-55.56	213.8	
				9	70.71	70.71	45.0					40	-77.30	-63.44	219.4	
1	2	3	5	10	63.44	77.30	50.6	3	6	11	21	41	-70.71	-70.71	225.0	
				11	55.56	83.15	56.3					42	-63.44	-77.30	230.6	
			6	12	47.14	88.19	61.9					22	43	-55.56	-83.15	236.3
				13	38.27	92.39	67.5					44	-47.14	-88.19	241.9	
		4	7	14	29.03	95.69	73.1			12	23	45	-38.27	-92.39	247.5	
				15	19.51	98.08	78.8					46	-29.03	-95.69	253.1	
			8	16	9.80	99.52	84.4				24	47	-19.51	-98.08	258.8	
				17	0.00	100.00	90.0					48	-9.80	-99.52	264.4	
	3	5	9	18	-9.80	99.52	95.6		7	13	25	49	0.00	-100.00	270.0	
				19	-19.51	98.08	101.3					50	9.80	-99.52	275.6	
			10	20	-29.03	95.69	106.9				26	51	19.51	-98.08	281.3	
				21	-38.27	92.39	112.5					52	29.03	-95.69	286.9	
		6	11	22	-47.14	88.19	118.1			14	27	53	38.27	-92.39	292.5	
				23	-55.56	83.15	123.8					54	47.14	-88.19	298.1	
			12	24	-63.44	77.30	129.4				28	55	55.56	-83.15	303.8	
				25	-70.71	70.71	135.0					56	63.44	-77.30	309.4	
2	4	7	13	26	-77.30	63.44	140.6	4	8	15	29	57	70.71	-70.71	315.0	
				27	-83.15	55.56	146.3					58	77.30	-63.44	320.6	
			14	28	-88.19	47.14	151.9				30	59	83.15	-55.56	326.3	
				29	-92.39	38.27	157.5					60	88.19	-47.14	331.9	
		8	15	30	-95.69	29.03	163.1			16	31	61	92.39	-38.27	337.5	
				31	-98.08	19.51	168.8					62	95.69	-29.03	343.1	
			16	32	-99.52	9.80	174.4				32	63	98.08	-19.51	348.8	
												64	99.52	-9.80	354.4	

A4988***DMOS Microstepping Driver with Translator
and Overcurrent Protection***

Pin-out Diagram



Terminal List Table

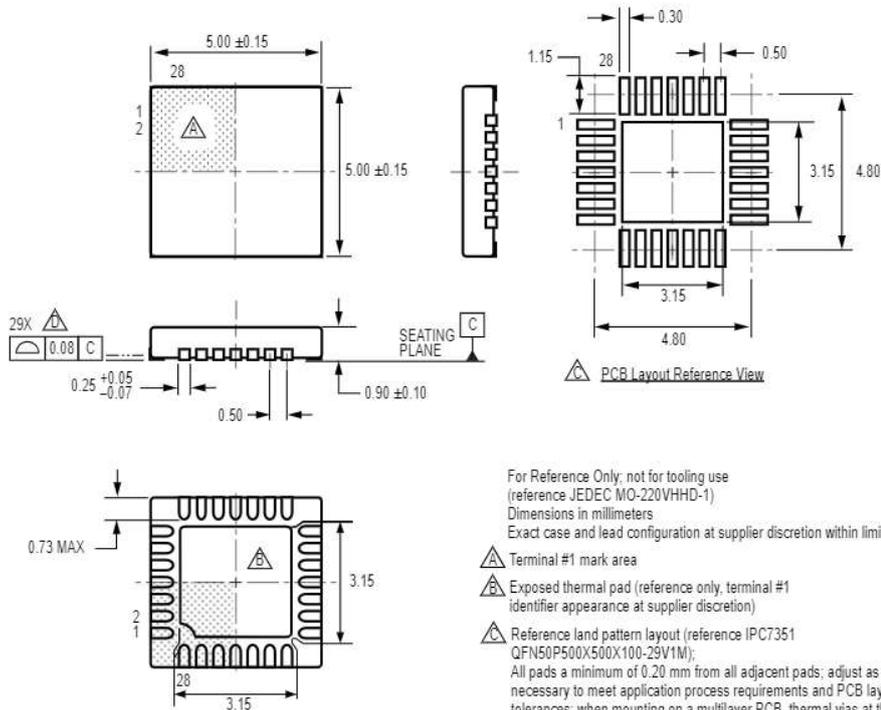
Name	Number	Description
CP1	4	Charge pump capacitor terminal
CP2	5	Charge pump capacitor terminal
VCP	6	Reservoir capacitor terminal
VREG	8	Regulator decoupling terminal
MS1	9	Logic input
MS2	10	Logic input
MS3	11	Logic input
$\overline{\text{RESET}}$	12	Logic input
ROSC	13	Timing set
$\overline{\text{SLEEP}}$	14	Logic input
VDD	15	Logic supply
STEP	16	Logic input
REF	17	G_m reference voltage input
GND	3, 18	Ground*
DIR	19	Logic input
OUT1B	21	DMOS Full Bridge 1 Output B
VBB1	22	Load supply
SENSE1	23	Sense resistor terminal for Bridge 1
OUT1A	24	DMOS Full Bridge 1 Output A
OUT2A	26	DMOS Full Bridge 2 Output A
SENSE2	27	Sense resistor terminal for Bridge 2
VBB2	28	Load supply
OUT2B	1	DMOS Full Bridge 2 Output B
$\overline{\text{ENABLE}}$	2	Logic input
NC	7, 20, 25	No connection
PAD	–	Exposed pad for enhanced thermal dissipation*

*The GND pins must be tied together externally by connecting to the PAD ground plane under the device.

A4988

DMOS Microstepping Driver with Translator and Overcurrent Protection

ET Package, 28-Pin QFN with Exposed Thermal Pad



For Reference Only; not for tooling use
 (reference JEDEC MO-220VHHD-1)
 Dimensions in millimeters
 Exact case and lead configuration at supplier discretion within limits shown

- △ Terminal #1 mark area
- △ Exposed thermal pad (reference only, terminal #1 identifier appearance at supplier discretion)
- △ Reference land pattern layout (reference IPC7351 QFN50P500X500X100-29V1M); All pads a minimum of 0.20 mm from all adjacent pads; adjust as necessary to meet application process requirements and PCB layout tolerances; when mounting on a multilayer PCB, thermal vias at the exposed thermal pad land can improve thermal dissipation (reference EIA/JEDEC Standard JESD51-5)
- △ Coplanarity includes exposed thermal pad and terminals

Copyright ©2009-2010, Allegro MicroSystems, Inc.

The products described here are manufactured under one or more U.S. patents or U.S. patents pending.

Allegro MicroSystems, Inc. reserves the right to make, from time to time, such departures from the detail specifications as may be required to permit improvements in the performance, reliability, or manufacturability of its products. Before placing an order, the user is cautioned to verify that the information being relied upon is current.

Allegro's products are not to be used in life support devices or systems, if a failure of an Allegro product can reasonably be expected to cause the failure of that life support device or system, or to affect the safety or effectiveness of that device or system.

The information included herein is believed to be accurate and reliable. However, Allegro MicroSystems, Inc. assumes no responsibility for its use; nor for any infringement of patents or other rights of third parties which may result from its use.

For the latest version of this document, visit our website:

www.allegromicro.com



RAMPS 1.4

<http://reprap.org/wiki/ramps>



This is open hardware: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This hardware design is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

!!! Reversing input power, and/or inserting stepper drivers incorrectly will destroy electronics and cause fire hazard!!!

Test all electronics thoroughly before placing into service.

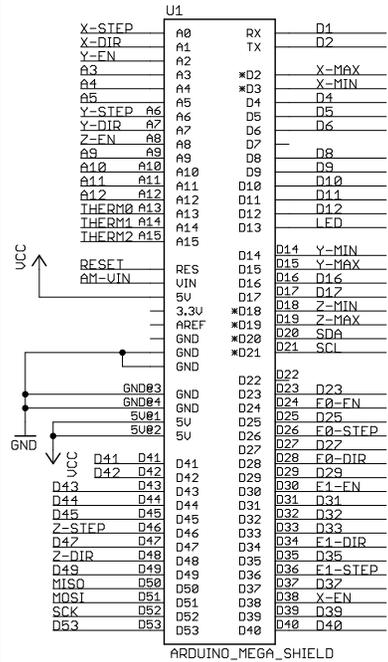
Do not leave power supplied to electronics unattended, or run machines unattended due to the risk of fire and malfunction.

This is NOT a toy and it contains small and sharp parts. Children can choke or suffocate by swallowing small objects. Keep all parts away from children and never leave printer/parts unattended.

Bill of Materials

Qty	Part Name	Value	Package
3	C1,C5,C8	10uF	153CLV-0405
1	C2	100nF	C0805
6	C3,C4,C6,C7,C9,C10	100uF	153CLV-0605
2	D1,D2	1N4004	DO41-10
1	F1	MFR500	MFR500
1	F2	MFR1100	MFR1100
1	LED1	green	CHIP-LED0805
3	LED2,LED3,LED4	red	CHIP-LED0805
3	Q1,Q2,Q3	STP55NF06L	TO220BV
5	R1,R7,R11,R21,R22	4.7K (1%)	R0805
8	R2,R3,R4,R5,R6,R8,R9,R10	100k	R0805
1	R12	1K	R0805
3	R23,R24,R26	1.8K	R0806
3	R13,R14,R15	10r	R0805
5	R16,R17,R18,R19,R20	10k	R0805
1	S1	B3F-3100	B3F-31XX
1	U\$2	282837-6	282837-6
1	X4	MSTBA4	MSTBA4
1		0.1" 18x2 pin	
8		0.1" 2x3 pin	
5		0.1" 8x1 pin	
2		0.1" 6x1 pin	
6		0.1" 4x1 pin	
2		0.1" 24x1 female	
4		0.1" 8x1 female	
15		0.1" 2 pin jumper	

MEGA Conn.



RAMPS 1.4 (RepRap Arduino Mega Pololu Shield)

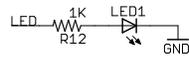
reprap.org/wiki/RAMPS1.4

Copyright 2011 Johnny Russell

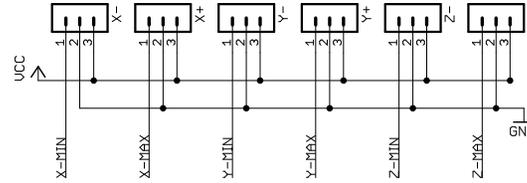
UltiMachine

GPL v3

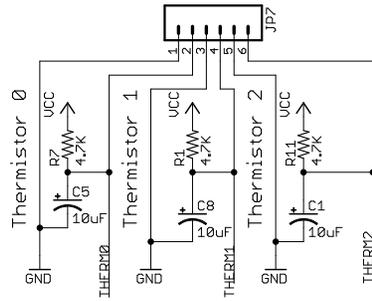
LED



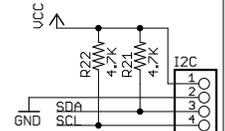
Endstops



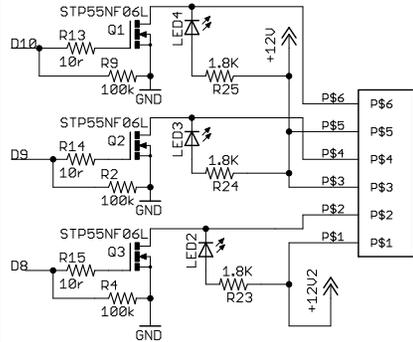
Thermistors



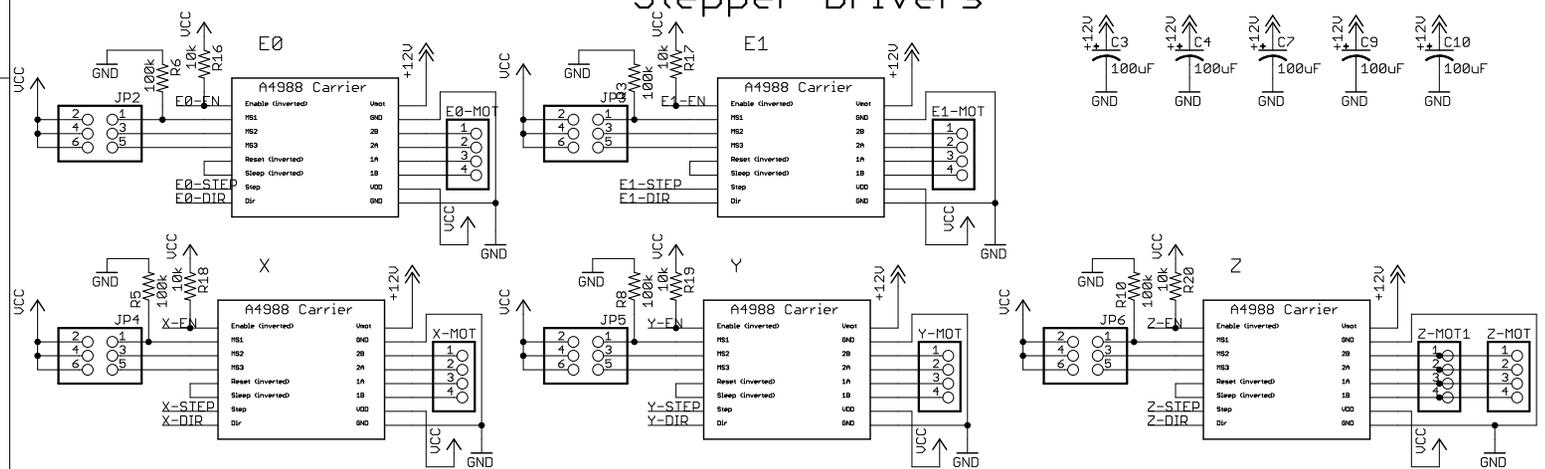
I2C



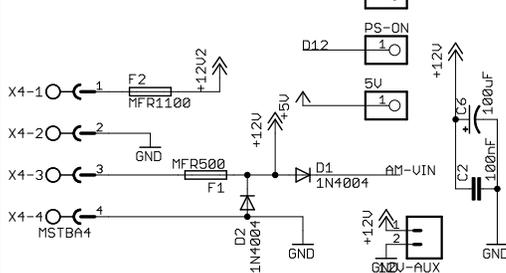
Heaters & Fans



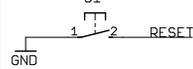
Stepper Drivers



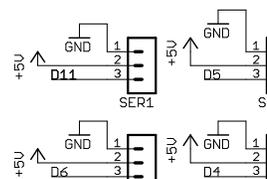
Power



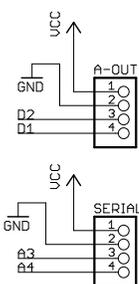
Reset



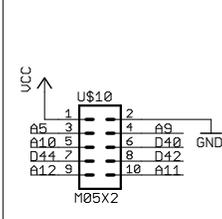
Servos



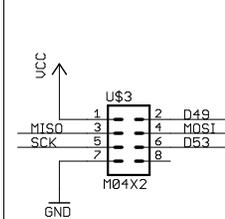
AUX-1



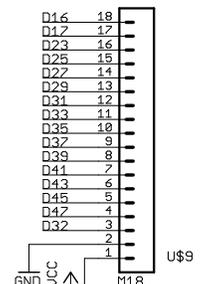
AUX-2



AUX-3



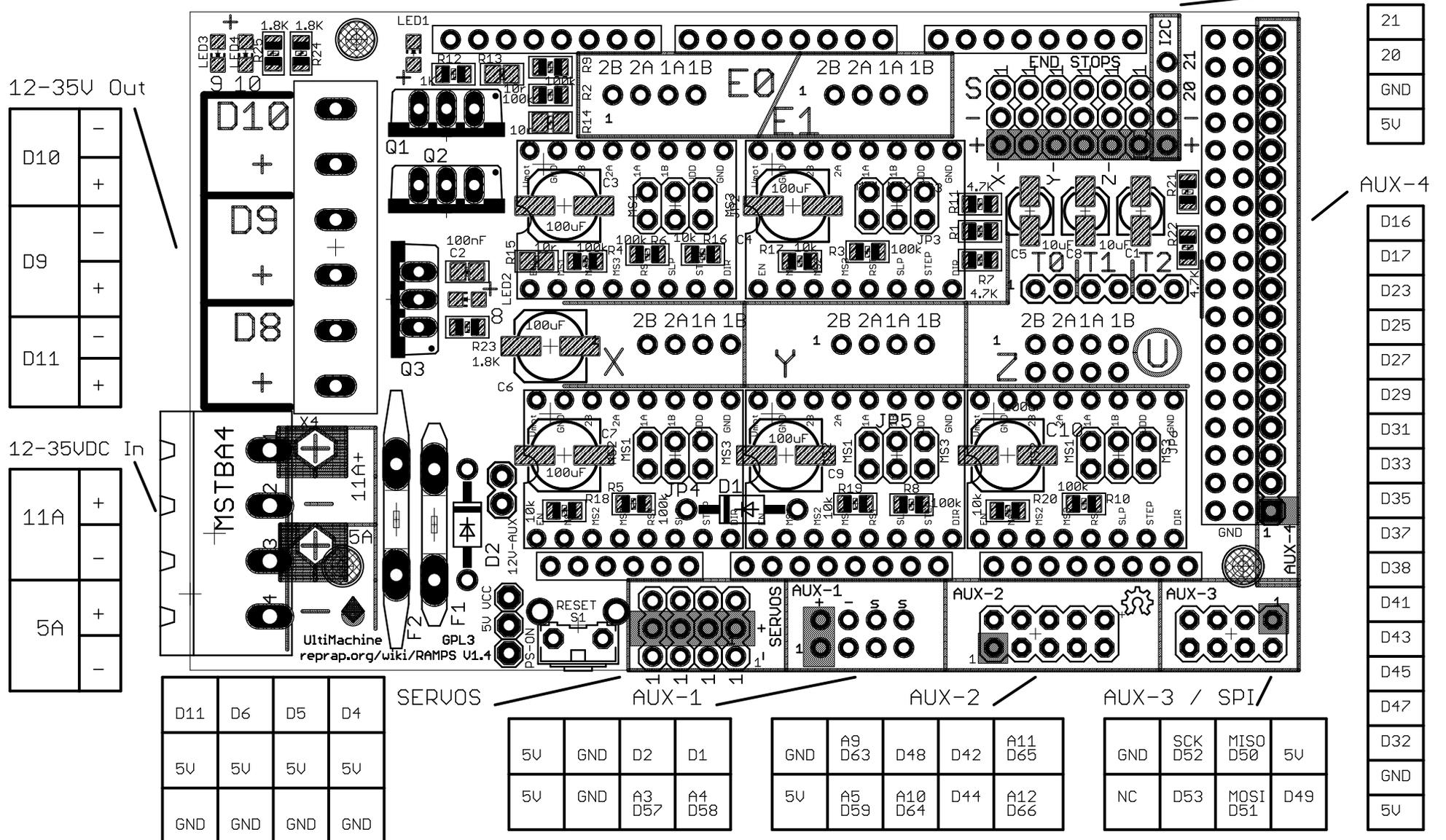
AUX-4

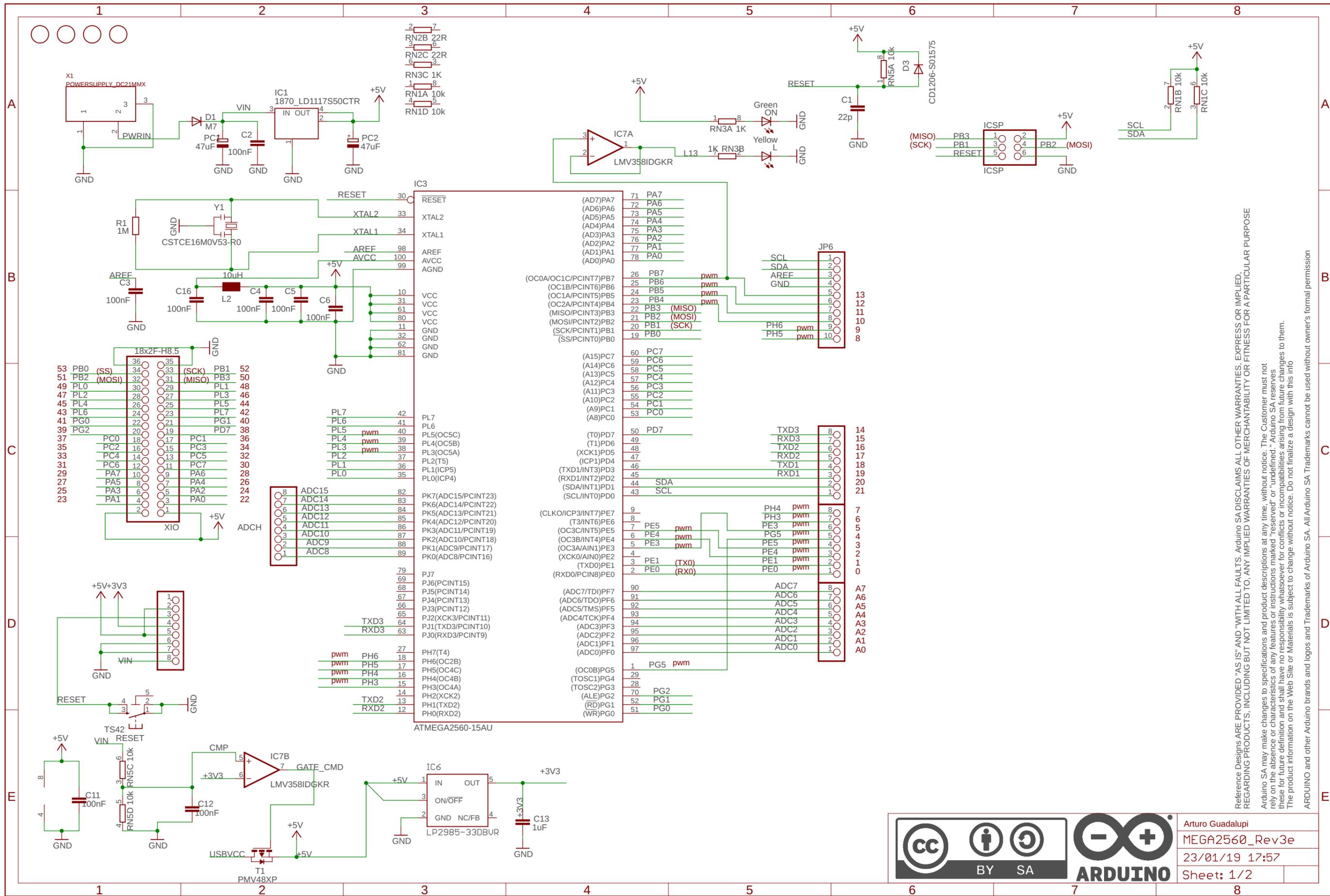


RAMPS 1.4 (RepRap Arduino MEGA Pololu Shield)

GPL v3

Reversing input power, and inserting stepper drivers incorrectly will destroy electronics.





Arturo Guadalupi
 MEGA2560_Rev3e
 23/01/19 17:57
 Sheet: 1/2



ATmega640/V-1280/V-1281/V-2560/V-2561/V

8-bit Microcontroller with 16/32/64KB In-System Programmable Flash

DATASHEET

Features

- High Performance, Low Power AVR[®] 8-Bit Microcontroller
- Advanced RISC Architecture
 - 135 Powerful Instructions – Most Single Clock Cycle Execution
 - 32 × 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 16 MIPS Throughput at 16MHz
 - On-Chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
 - 64K/128K/256KBytes of In-System Self-Programmable Flash
 - 4Kbytes EEPROM
 - 8Kbytes Internal SRAM
 - Write/Erase Cycles:10,000 Flash/100,000 EEPROM
 - Data retention: 20 years at 85°C/ 100 years at 25°C
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
 - Endurance: Up to 64Kbytes Optional External Memory Space
- QTouch[®] library support
 - Capacitive touch buttons, sliders and wheels
 - QTouch and QMatrix acquisition
 - Up to 64 sense channels
- JTAG (IEEE[®] std. 1149.1 compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
 - Four 16-bit Timer/Counter with Separate Prescaler, Compare- and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Four 8-bit PWM Channels
 - Six/Twelve PWM Channels with Programmable Resolution from 2 to 16 Bits (ATmega1281/2561, ATmega640/1280/2560)
 - Output Compare Modulator
 - 8/16-channel, 10-bit ADC (ATmega1281/2561, ATmega640/1280/2560)
 - Two/Four Programmable Serial USART (ATmega1281/2561, ATmega640/1280/2560)
 - Master/Slave SPI Serial Interface
 - Byte Oriented 2-wire Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
 - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
 - 54/86 Programmable I/O Lines (ATmega1281/2561, ATmega640/1280/2560)
 - 64-pad QFN/MLF, 64-lead TQFP (ATmega1281/2561)
 - 100-lead TQFP, 100-ball CBGA (ATmega640/1280/2560)
 - RoHS/Fully Green
- Temperature Range:
 - -40°C to 85°C Industrial
- Ultra-Low Power Consumption
 - Active Mode: 1MHz, 1.8V: 500µA
 - Power-down Mode: 0.1µA at 1.8V
- Speed Grade:
 - ATmega640V/ATmega1280V/ATmega1281V:
 - 0 - 4MHz @ 1.8V - 5.5V, 0 - 8MHz @ 2.7V - 5.5V
 - ATmega2560V/ATmega2561V:
 - 0 - 2MHz @ 1.8V - 5.5V, 0 - 8MHz @ 2.7V - 5.5V
 - ATmega640/ATmega1280/ATmega1281:
 - 0 - 8MHz @ 2.7V - 5.5V, 0 - 16MHz @ 4.5V - 5.5V
 - ATmega2560/ATmega2561:
 - 0 - 16MHz @ 4.5V - 5.5V

<http://ww1.microchip.com/downloads/en/DeviceDoc/ATmega640-1280-1281-2560-2561-Datasheet-DS40002211A.pdf>

Codigo Marlin:

https://drive.google.com/drive/folders/1Wq7cB7BvMNF-IDWe6712X1sAIQa7LJiL?usp=drive_link