

ESCUELA SUPERIOR DE INGENIERÍA Y TECNOLOGÍA

TRABAJO DE FIN DE GRADO

Grado en Ingeniería Electrónica Industrial y Automática

TÍTULO:

**DISEÑO E IMPLEMENTACIÓN DE UNA IMPRESORA
PARA LA REALIZACIÓN DE PLACAS DE CIRCUITOS
IMPRESOS (PCB) BASADA EN GRABADOR LÁSER.**

Alumno/a: Elena Velázquez Navarro.

Tutores: Manuel Rodríguez Valido
David Hernández Expósito.

Julio 2017

Agradecimientos

En primer lugar, quiero agradecer a mis tutores, Manuel Rodríguez Valido y David Hernández Expósito, por conocimientos y consejos, sin los cuales el desarrollo de este proyecto no hubiese sido posible.

También a Ramón Pérez, por su disposición ante cualquier problema y por introducirme en el mundo de las impresoras 3D.

A todo el servicio de electrónica, gracias por su amabilidad y su colaboración al facilitarme los recursos necesarios en todo momento.

A mis padres y mi hermana, por comprenderme y apoyarme durante estos cuatro años, haciendo que cada día sea mejor persona. También a mis abuelos, por impulsarme siempre a alcanzar nuevas metas.

Y, por último, a mis compañeros y amigos, en especial a Evelyn, David, Julio, Sergio, Iván, Carlos y Adrián, quienes me han animado en los momentos más difíciles, llegando a ser parte esencial de mi vida.

ÍNDICE GENERAL

Memoria.....	3
Capítulo 1: Objetivo.....	12
Capítulo 2: Marco teórico.....	16
Capítulo 3: Descripción del hardware y del software.....	23
Capítulo 4: Diseño de un sistema de soporte y adaptación del láser.....	37
Capítulo 5: Circuito de control.....	46
Capítulo 6: Puesta en marcha.	58
Capítulo 7: Pruebas y resultados.	69
Capítulo 8: Presupuesto.	83
Capítulo 9: Conclusiones y futuras mejoras.	86
Bibliografía.....	89
Anexos.....	90



ESCUELA SUPERIOR DE INGENIERÍA Y TECNOLOGÍA

MEMORIA

TRABAJO DE FIN DE GRADO

Grado en Ingeniería Electrónica Industrial y Automática

TÍTULO:

**DISEÑO E IMPLEMENTACIÓN DE UNA IMPRESORA
PARA REALIZACIÓN DE PLACAS DE CIRCUITOS
IMPRESOS (PCB) BASADA EN GRABADOR LÁSER.**

Alumno/a: Elena Velázquez Navarro.
Tutores: Manuel Rodríguez Valido
David Hernández Expósito.

Julio 2017

ÍNDICE DE LA MEMORIA

Resumen	10
Abstract.....	11
Capítulo 1: Objetivo.	
1. Contexto.....	13
2. Objetivo principal.....	14
3. Organización del proyecto.	14
Capítulo 2: Marco teórico.	
1. ¿Qué es una PCB?	17
1.1. Partes que la componen.	17
1.2. Tipos de PCB.....	17
1.3. Elementos de una PCB.	18
2. Métodos de fabricación.	20
2.1. Fabricación por ataque químico.	20
2.2. Fabricación en seco.....	21
3. Reglas de diseño.	21
Capítulo 3: Descripción hardware y software del sistema.	
1. Introducción.....	24
2. Hardware.	24
2.1. Impresora 3D RepRap.	24
2.2. Ventilador externo.....	29
2.3. Diodo láser.	29
2.4. Circuito de control (“Driver Circuit”).	30
2.5. Fuente de alimentación.	31
2.6. Placas fotosensibles positivas.	31
3. Software.....	31
3.1. IDE Arduino.	32
3.2. Repetier Firmware.	32
3.3. Repetier-Host.....	33
3.4. KiCad.	34
3.5. FlatCAM.....	34
3.6. Inkscape.	35

3.7. FreeCAD.	35
Capítulo 4: Diseño de un sistema de soporte y adaptación del láser.	
1. Introducción.	38
2. Sistema óptico.	38
2.2. Lente convergente.	38
3. Diseño de la pieza de soporte.	41
4. Proceso de impresión.	43
4.1. Material: PLA.	43
4.2. Exportación del fichero STL.	43
4.3. Creación de capas.	43
4.4. Impresión 3D.	45
Capítulo 5: Circuito de control.	
1. Introducción.	47
2. Funcionamiento.	47
3. Señal PWM en Arduino.	48
4. Pruebas de la intensidad.	49
4.1. Proyecto en Arduino IDE.	50
4.2. Resultados.	50
5. Conexión del láser a la placa RAMPS 1.4.	53
6. Circuito adaptador de señal.	53
6.2. Esquemático.	54
6.3. Montaje en placa de prototipo.	56
Capítulo 6: Puesta en marcha.	
1. Introducción.	59
2. Configuración del Firmware Repetier.	59
2.1. Caracterización de la impresora.	59
2.2. Motores y sensores. Resolución.	59
2.3. Control del láser.	60
2.4. Ventilador externo.	61
3. Correcciones mecánicas.	61
4. Área de impresión.	61
4.1. Creación de la plantilla.	62
4.2. Sistema de referencia.	63
5. Proceso de diseño e insolación de PCBs.	64

5.1. Diseño del esquemático y Layout.	64
5.2. Trayectoria del láser.....	65
5.3. Obtención de G-Code.....	66
5.4. Insolación.	68
 Capítulo 7: Pruebas y resultados.	
1. Introducción.....	70
2. Intensidad.	70
2.1. Modificación de la velocidad de trazado.	71
2.2. Número de pasadas.	72
3. Resolución de los ejes y separación entre líneas.....	73
3.1. Separación entre líneas sobre cartón.	74
3.2. Separación entre líneas sobre placa fotosensible.	75
4. PCB de prueba.....	76
5. Resultados.....	78
5.1. Resultados de insolación.....	78
5.2. Resultados de serigrafía.....	81
 Capítulo 8: Presupuesto.	
1. Introducción.....	84
2. Coste de los materiales.....	84
3. Coste de la mano de obra.	85
4. Presupuesto total del proyecto.	85
 Capítulo 9: Conclusiones y futuras mejoras.	
1. Conclusiones.....	87
2. Futuras mejoras.	87
3. Opinión personal.....	88

ÍNDICE DE FIGURAS

Figura 1.1. Ejemplo de fotolito.....	13
Figura 2.1. Partes de una PCB.	19
Figura 3.1. Impresora 3D Prusa i3.	25
Figura 3.2. Motor NEMA 17.	26
Figura 3.3. Placa Arduino Mega 2560.	27
Figura 3.4. Conexiones de la placa RAMPS 1.4 para impresora 3D.	28
Figura 3.5. Ventilador externo.	29
Figura 3.6. Disipador de calor y circuito de control del láser.	30
Figura 3.7. Circuito de control del láser.	30
Figura 3.8. Software utilizado.	31
Figura 3.9. Función del firmware.	32
Figura 3.10. Entorno de <i>Repetier-Host</i>	34
Figura 4.1. Lente convergente.	39
Figura 4.2. Lente. Vista desde arriba.	40
Figura 4.3. Lente. Vista frontal.	40
Figura 4.4. Base de la pieza de soporte del láser.	41
Figura 4.5. Vista axonométrica de la pieza de soporte.....	42
Figura 4.6. Pieza de soporte en <i>Repetier-Host</i>	44
Figura 4.7. Creación de capas de la pieza de soporte.....	44
Figura 4.8. Posición final de la pieza de soporte del láser.....	45
Figura 5.1. Circuito de control del láser.	48
Figura 5.2. Señal PWM con distintos ciclos de trabajo.	49
Figura 5.3. Programa en IDE Arduino. Prueba de intensidad.	50
Figura 5.4. Gráfica de intensidad consumida e intensidad del láser.	51
Figura 5.5. Ciclo de trabajo igual al 8%.....	52
Figura 5.6. Ciclo de trabajo igual al 20%.....	52
Figura 5.7. Ciclo de trabajo igual al 50%.....	52
Figura 5.8. Esquema del circuito adaptador de señal.	53

Figura 5.9. Esquemático del bloque de alimentación.	54
Figura 5.10. Esquemático del bloque reductor de amplitud.	55
Figura 5.11. Esquemático de la puerta NOT.	56
Figura 5.12. Captura del osciloscopio. Señal pin 9 y señal adaptada.	56
Figura 5.13. Montaje final del circuito adaptador de señal.	57
Figura 6.1. Captura de la plantilla: Área de impresión.	62
Figura 6.2. Sistema de referencia.	63
Figura 6.3. Proceso de insolación.	64
Figura 6.4. Forma de calcular la trayectoria del láser.	66
Figura 6.5. Captura de la herramienta <i>Laser Tool</i>	67
Figura 7.1. Prueba de intensidad sin lente.	71
Figura 7.2. Prueba de intensidad con lente.	71
Figura 7.3. Prueba de velocidad de trazado con la lente.	72
Figura 7.4. Prueba de número de pasadas.	72
Figura 7.5. Prueba de resolución de los ejes y separación entre líneas.....	73
Figura 7.6. Resultado de la prueba de resolución de los ejes y separación entre líneas.....	74
Figura 7.7. Prueba de separación entre líneas sobre cartón.....	74
Figura 7.8. Prueba separación entre líneas sobre placa fotosensible.....	75
Figura 7.9. Resultado. Separación entre líneas sobre placa fotosensible.....	76
Figura 7.10. Fotolito de la PCB de prueba.....	77
Figura 7.11. Resultado de los cuadrados con dos pasadas.....	78
Figura 7.12. Resultado de la PCB de prueba con lente.....	79
Figura 7.13. Resultado de la PCB de prueba sin lente. Diámetro: 0,5 mm.....	80
Figura 7.14. Resultado de la PCB de prueba sin lente. Diámetro: 0,3 mm.....	81
Figura 7.15. Resultado de serigrafía con texto.....	81
Figura 7.16. Resultado de serigrafía con las huellas de los componentes.....	82

ÍNDICE DE LAS TABLAS

Tabla 5.1. Variación de intensidad según el ciclo de trabajo.....	51
Tabla 6.1. Cálculo del número de pasos por milímetro de los motores.....	60
Tabla 7.1. Resultados de la separación mínima del eje y.	75
Tabla 7.2. Componentes de la PCB de prueba.	77
Tabla 7.3. Tiempos de fabricación. Cuadrados.	78
Tabla 7.4. Parámetros para la PCB de prueba con lente.	79
Tabla 7.5. Tiempos de fabricación. PCB de prueba con diámetro igual a 0,5 mm	80
Tabla 8.1. Coste de los materiales.	84
Tabla 8.2. Coste de la mano de obra.	85
Tabla 8.3. Presupuesto total del proyecto.	85

Resumen

El presente trabajo consiste en la adaptación de una impresora 3D para la insolación de placas de circuitos impresos. De esta manera, pretende ser una alternativa más automatizada del proceso tradicional de transferir el diseño mediante un fotolito y una insoladora.

Por un lado, la insolación de las placas será posible gracias al uso de un diodo láser que emite una luz azul-violeta. Por esta razón, se sustituyó el extrusor de la máquina por una pieza que sujetará al disipador de calor del láser, así efectúa el mismo tipo de desplazamientos que el extrusor en la impresión en 3D.

Finalmente, se ha diseñado una PCB de prueba para comprobar el funcionamiento del sistema en el entorno real, consiguiendo la correcta insolación de una placa fotosensible. No obstante, con los resultados obtenidos se pudo detectar que el prototipo actual presenta algunas carencias que podrán ser corregidas en el futuro.

Abstract

This project consists of a modification of a 3D printer to expose printed circuit boards to the light. Therefore, it could be a more automated option to manufacturing PCBs than the traditional process, which is based on the photolith film and the contact copier.

On the one hand, the design will be engraved in the board using a laser diode, which emits blue-violet light. For that reason, the extruder of the machine was replaced by a plastic piece that contains the laser heat sink. The result is that the laser makes the same movements as the extruder during 3D printing.

Finally, it has been designed a PCB test to verify the proper operation of the system in a real environment, concluding that the photolithography of the photosensitive board was successful. Nevertheless, the results show that the system presents some failures that could be resolved in the future.

Capítulo 1: Objetivo.

1. Contexto.

La fabricación de placas de circuitos impresos (*Printed Circuit Board o PCBs*) para prototipado es una tarea imprescindible en el diseño de sistemas electrónicos. Existen dos métodos principales de fabricación: fabricación por ataque químico y fabricación en seco. El método más usado debido a su sencillez, es el revelado por ataque químico, el cual se caracteriza por utilizar un fotolito para transferir el diseño a la placa mediante la insolación con luz ultravioleta. En cambio, el método en seco consiste en retirar el material sobrante con una máquina CNC (*Computer Numerical Control*), dejando solo el cobre en las zonas de conducción.

Por un lado, las fresadoras CNC presentan una serie de problemas. En primer lugar, su precio es relativamente elevado y la resolución obtenida es baja si se compara con otros métodos de fabricación. Además, se suelen usar brocas, que resultan ser altamente frágiles cuanto menor es su diámetro.

Por otra parte, la técnica por ataque químico depende de distintas variables como, por ejemplo, la calidad del fotolito, el proceso de iluminación y el espesor de la película fotosensible. Siendo todas ellas mejorables empleando procesos y equipos de alto coste. Desde el punto de vista de los residuos, los fotolitos son un fungible que pierde su utilidad cuando el procedimiento ha terminado.

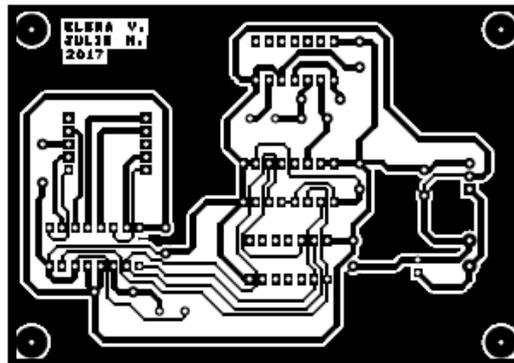


Figura 1.1. Ejemplo de fotolito.

En consecuencia, se aprovecha el uso extendido de las impresoras 3D para automatizar distintos procesos de manera económica, simplemente sustituyendo el cabezal por un soldador, láser, etc.

2. Objetivo principal.

El objetivo de este trabajo se centra en el diseño e implementación de una impresora para la realización de placas de circuitos impresos basada en grabador láser. En concreto, se pretende automatizar el proceso de insolación de placas fotosensibles para reducir los inconvenientes derivados de la impresión del fotolito.

Para conseguirlo se propone el uso de un láser como cabezal de una impresora 3D y a partir de un fichero Gerber, dónde están las características del circuito, se dirige la máquina para insolar el diseño sobre el soporte de fabricación.

El diseño de un circuito impreso no es más que un conjunto de formas geométricas que se han obtenido mediante una herramienta CAD. Normalmente, el fichero de salida de esta herramienta contiene las coordenadas de las pistas, y es interpretado por el firmware para controlar los movimientos en el plano.

3. Organización del proyecto.

El proyecto se encuentra organizado por capítulos dónde se explicarán los dispositivos empleados y el desarrollo del software que ha sido necesario para la ejecución del mismo. Además, se mostrarán los resultados obtenidos y como ha sido el proceso de pruebas para conseguir insolar correctamente una PCB.

Por lo tanto, la memoria del trabajo se divide de la siguiente manera:

- **Capítulo 2: Marco teórico.** Teoría acerca de las placas de circuitos impresos y los métodos de fabricación de las mismas.
- **Capítulo 3: Descripción hardware y software del sistema.** Se explican los dispositivos, elementos y programas que han sido utilizados.
- **Capítulo 4: Circuito de control.** Funcionamiento del circuito que controla el láser mediante el uso de una señal PWM.
- **Capítulo 5: Diseño de un sistema de soporte y adaptación del láser.** Diseño de la pieza de soporte para el láser.

- **Capítulo 6: Puesta en marcha.** Configuración del software y del hardware para la realización de las pruebas.
- **Capítulo 7: Pruebas y resultados.** Explicación de las pruebas realizadas para obtener los resultados.
- **Capítulo 8: Conclusiones y futuras mejoras.** Conclusiones del proyecto y aspectos mejorables en el futuro.

Capítulo 2: Marco teórico.

1. ¿Qué es una PCB?

Las Placas de Circuitos Impresos o PCB son un soporte físico para componentes electrónicos que permite establecer la conexión eléctrica entre los mismos. Habitualmente, están compuestas por varias capas conductoras, separadas unas de otras por capas de material aislante y pegadas entre sí.

1.1. Partes que la componen.

- Sustrato o material base.

Es una lámina rígida o flexible que puede ser de distintos materiales dependiendo de la aplicación y del entorno en el que se encontrará la PCB. En el caso de que se necesite gran rigidez, se utilizarán las resinas fenolíticas rígidas impregnadas con papel y la resina Epoxy impregnada con fibra de vidrio. Mientras que, si interesa que sea flexible, se usará el Film de “Mylar”, “teflón” o poliamidas como sustrato.

- Lámina conductora.

La lámina conductora debe ser de cobre electrolítico debido a su elevada conductividad y su resistencia a la corrosión. Además, puede ser de distintos espesores, siendo más anchas las empleadas para la alimentación.

- Adhesivo.

El adhesivo debe poseer las características determinadas para no alterar la placa base y no permitir la separación de las láminas de cobre y el sustrato. Igualmente, debe soportar la acción de los disolventes y reactivos que son utilizados en los procesos de fabricación.

1.2. Tipos de PCB.

Existen distintos tipos de Placas de Circuitos Impresos según el número de capas que contengan. En general, las placas tienen dos superficies o caras externas: una cara de componentes y una cara de pistas, también denominada cara de soldadura.

De esta forma, cuando una placa tiene una cara de pistas se llama “Placa de una cara” o SSB (*Single-Sided Board*). En cambio, aquellas placas que permiten trazar pistas tanto en la cara de componentes como la de soldadura se denominan “Placas de doble cara” o DSB (*Double-Sided Boards*).

A medida que aumenta el número de componentes a colocar, también aumenta el número de pistas a trazar por lo que las dos caras de trabajo son insuficientes. Por lo tanto, para evitar aumentar el tamaño de la PCB, se desarrolla la tecnología Multicapa o MLB (*Multilayer Board*). La cual consiste en crear capas compactas compuestas por una lámina de cobre, sustrato y adhesivo, una sobre otra.

1.3. Elementos de una PCB.

La mayoría de las placas de circuitos impresos poseen los siguientes elementos.

- Pistas o 'nets'.

Son los elementos que establecen la conexión eléctrica entre dos puntos de la PCB. Las características principales que definen a una pista son: la temperatura y la corriente, y a partir de ellas se puede establecer el ancho de las mismas.

- Pad.

Se utilizan superficies de cobre o pad que permite la soldadura de los pines de los componentes con el cobre de la PCB. Se utilizan distintas formas de pad dependiendo del tipo de tecnología de los componentes: *THD (Through-hole technology)* o *SMD (Surface-mount technology)*.

El pad para un componente THD se caracteriza por contener un orificio en el centro del mismo, en el cual se inserta uno de los pines del dispositivo. Por otro lado, los pads correspondientes a la tecnología SMD tienen un área cuadrada o rectangular. Pero en este caso, no poseen ninguna perforación puesto que los componentes de este tipo son soldados directamente al pad.

- Nodos térmicos.

Los nodos térmicos permiten la conexión de positivo y negativo de cualquier capa a los planos de alimentación y masa respectivamente.

- Taladros o “drills”.

Los taladros son destinados generalmente a la inserción de los componentes de tecnología THD. También se suelen hacer cuatro perforaciones, una en cada esquina, con un diámetro mayor para facilitar la sujeción de la placa.

- Vías.

Las pistas de diferentes capas se pueden conectar eléctricamente mediante vías, que son taladros metalizados que establecen una conexión eléctrica. Por otra parte, nunca deben ser utilizados para insertar componentes.

Dependiendo del tipo de capas que unan estos elementos reciben una denominación diferente, por ejemplo, si conecta las dos capas externas entre sí se denominará vía pasante. Mientras que si las pistas se encuentran en capas internas será una vía enterrada. Y, por último, si conecta una externa con una o varias internas será vía ciega [1].

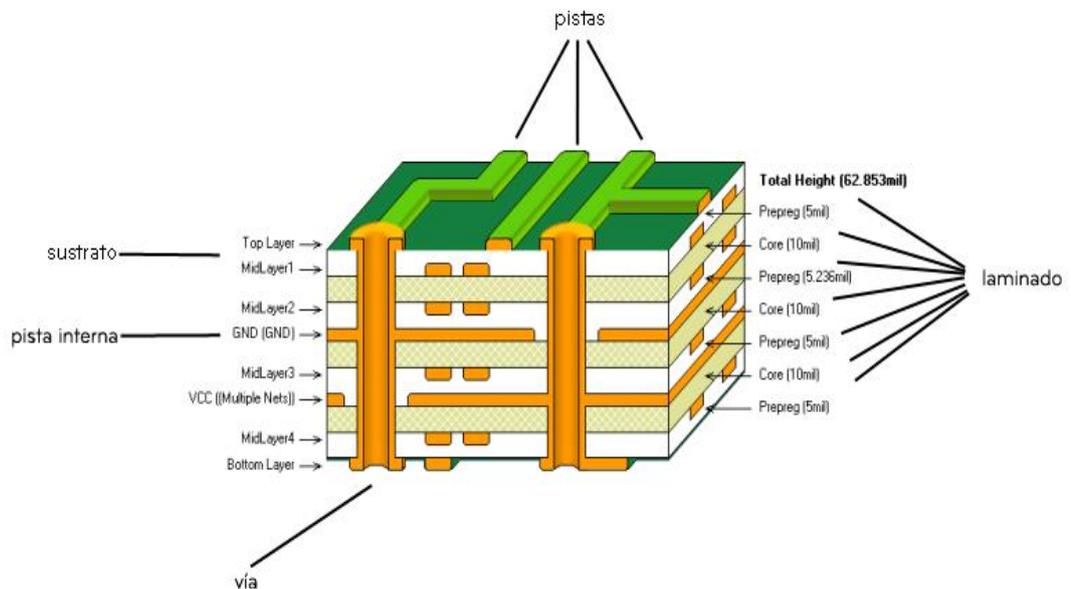


Figura 2.1. Partes de una PCB [16].

2. Métodos de fabricación.

Entre las múltiples técnicas de fabricación de que existen, destaca la fabricación por ataque químico y en seco, por ser las más comunes en la producción no automatizada de PCBs.

2.1. Fabricación por ataque químico.

La fabricación por ataque químico o el método de insolación está formado por un conjunto de procesos que derivan en la creación de PCBs: diseño por software, insolación, revelado, ataque químico, taladrado y soldadura.

- Diseño por software.

En primer lugar, se diseña el esquemático, que es el circuito electrónico que se desea implementar y las correspondientes conexiones, mediante herramientas CAD como *Eagle* o *KiCad*. Luego, se genera el Layout donde se colocan las “footprint” (“huellas”) de los dispositivos y se organizan de la manera óptima para crear las pistas de acuerdo a las reglas de diseño.

- Insolación.

Una vez terminado el diseño, se obtiene el fotolito, que es una transparencia del patrón en negativo, es decir, las pistas y las áreas de cobre aparecerán en negro y el resto permanecerá transparente. De esta forma, llegará la luz UV (“Ultravioleta”) a las zonas donde se pretende eliminar el cobre.

Este tipo de proceso será adecuado para placas fotosensibles, las cuales poseen una capa de cobre cubierta por una resina fotosensible. Durante la insolación, esta resina es debilitada en las zonas en las que el fotolito deje pasar la luz.

- Revelado.

Tras la insolación, se introduce la placa en una disolución de sosa cáustica. Se agita el recipiente o la propia PCB para facilitar el proceso de eliminación de la resina fotosensible.

- Ataque químico.

Cuando se observa el cobre en las zonas que habían sido expuestas a la luz UV, se aplica una disolución de ácido. De esta manera, se elimina el cobre solo en las zonas sin la película fotosensible.

- Taladrado y soldadura.

Una vez obtenemos el resultado final del atacado químico, podemos realizar los taladros pertinentes. Luego, se van colocando y soldando los componentes. La soldadura puede ser por fusión, donde se requiere de una fuente de calor y de un material de aportación, el cual se fundirá cuando alcance su temperatura de fusión.

Normalmente, se emplea estaño como material de aportación y un soldador para llevarlo a su punto de fusión, así como un desoldador para corregir errores. De manera que la soldadura debe cubrir todo el pad formando un cono entre la base del mismo y la patilla del componente.

Una vez finalizado este proceso, se aplican sprays y lacas para proteger a la placa de la corrosión y aislarla eléctricamente.

2.2. Fabricación en seco.

Durante la fabricación en seco se aíslan las pistas del resto del cobre circulante mediante el fresado de su contorno lateral. Este sistema no utiliza fotolitografía ni ácidos, emplea en su lugar, microfresadoras o dispositivos láser y sistemas CAD/CAM que procesan la información de los archivos Gerber y Excellon. Estos tipos de ficheros son generados en el mismo programa en el que se diseña la colocación real de los componentes.

La secuencia de trabajo en este proceso será: fresado de la cara de soldadura y taladrado de la misma. En caso de hacerse para una placa de doble cara, se realiza lo mismo para la cara contraria [1].

3. Reglas de diseño.

Las reglas de diseño son recomendaciones para la creación de placas de circuitos impresos. A continuación, se enumerarán las más importantes que se deben llevar a cabo.

1. La separación general entre componentes debe ser de 0,5 mm, aunque depende del tipo de encapsulado. Por lo que no está permitido el solapamiento de componentes.
2. La distribución de los componentes debe ser sencilla, colocándolos paralelos al borde de la placa, es decir, horizontales o verticales. Permitiendo identificar su código, valor, etc. Aquellos que pertenezcan a un mismo grupo, se montarán en el mismo sentido.
3. Los conectores deben colocarse en el borde de la PCB para facilitar la conexión y desconexión salvo imposiciones de tipo mecánico.
4. Los circuitos integrados se situarán de forma ordenada y si es posible, equidistantes. Su dirección longitudinal deberá coincidir con la dirección longitudinal de la PCB y todos tendrán su polaridad orientada en el mismo sentido. El resto de componentes se distribuirán alrededor de los CI.
5. Dejar una distancia mínima de 2 mm entre el borde y los componentes, para que la PCB pueda ser sujeta en las mesas de montaje.
6. Los indicadores de señal, como los LED's, deben estar visibles. Igualmente, los elementos ajustables como los potenciómetros deben ser accesibles.
7. Se debe prever la fijación de los radiadores para disipar el calor, los cuales no aparecerán en el diseño.
8. Se debe minimizar la longitud de las "pistas críticas" si las hay.
9. Los CI más rápidos estarán cerca del conector de alimentación y su condensador de desacoplo a la distancia más corta posible de su patilla de alimentación.
10. Los CI que reciban señales de entrada estarán junto al conector correspondiente, y lo mismo con las señales de salida.
11. En el caso de componentes SMD en la capa "bottom" solo se suelen colocar resistencias, condensadores y los encapsulados SO.
12. Las pistas deben ser horizontales, verticales o formando ángulos de 45°. Además, los giros no deben ser de 90 °, se suavizarán con ángulos de 135 ° [1].

Capítulo 3:

Descripción hardware y software del sistema.

1. Introducción.

Un sistema de estas características necesita un software y un hardware que lo soporte. Como se ha mencionado anteriormente, se sustituirá el cabezal de la impresora 3D por un diodo láser. Por lo tanto, esta máquina será la base y soporte mecánico para alcanzar los objetivos del proyecto.

Con respecto al hardware, cabe señalar que la impresora cuenta con la placa de Arduino que se encarga del control general de la misma. Al él se conecta la placa RAMPS que convierte el código generado por las aplicaciones, en instrucciones dirigidas a los motores paso a paso. También, se resalta la importancia de los sensores final de carrera, sin los cuales no sería posible la automatización del sistema.

Por otro lado, es preciso el uso de software tanto para el control de la impresora como para el diseño de PCBs a alto nivel. El control se realiza mediante el firmware *Repetier*, que ha sido modificado de acuerdo a los parámetros de la impresora. En cuanto a las aplicaciones software, se ha seleccionado *KiCad* para el diseño de placas de circuitos impresos, *Inkscape* para la generación de archivos G-code y, por último, *Repetier-Host* para el proceso de insolación.

2. Hardware.

Como ya se ha mencionado en el apartado anterior, para la correcta realización del trabajo se han empleado los siguientes dispositivos:

- Impresora 3D RepRap.
- Ventilador externo.
- Diodo láser.
- Circuito de control.
- Fuente de alimentación.
- Placa fotosensible positiva.

2.1. Impresora 3D RepRap.

La impresora 3D RepRap se caracteriza por ser la primera máquina auto-replicante de uso general. Se utiliza el término auto-replicante porque es capaz de fabricar una copia de sí misma de forma autónoma, es decir, puede imprimir los objetos plásticos que la conforman. De esta

manera, con tan solo una máquina de este tipo (junto a los elementos eléctricos y electrónicos necesarios) se pueden crear infinitas impresoras. RepRap está disponible bajo licencia GNU GPL, la cual permite que pueda copiarse, estudiarse, distribuirse y mejorarse tanto el diseño como el código fuente [2].

En particular, la impresora empleada en este proyecto es una modificación del modelo “Prusa i3” que posee una estructura sencilla con cuatro motores paso a paso, uno para el eje x , otro para el eje y , mientras que los dos últimos son empleados para elevar o bajar el eje z . Además, se emplean tres sensores final de carrera, uno para la posición mínima de cada eje.

Por otro lado, cabe destacar que el movimiento de la impresora se realiza mediante la traslación del carro del extrusor a lo largo de los ejes x y z , a la vez que se traslada la base de madera en el eje y . Para esta aplicación tan solo se usarán los ejes que conforman las dos dimensiones, los cuales emplean correas dentadas. Sin embargo, el eje z , tan solo es empleado para colocar el láser a la distancia adecuada de la base, haciendo uso de dos tornillos de métrica cinco M5. También aparecen rodamientos lineales, los cuales permiten el deslizamiento de las distintas piezas sobre las varillas lisas o como elemento que permite el giro de los tornillos.

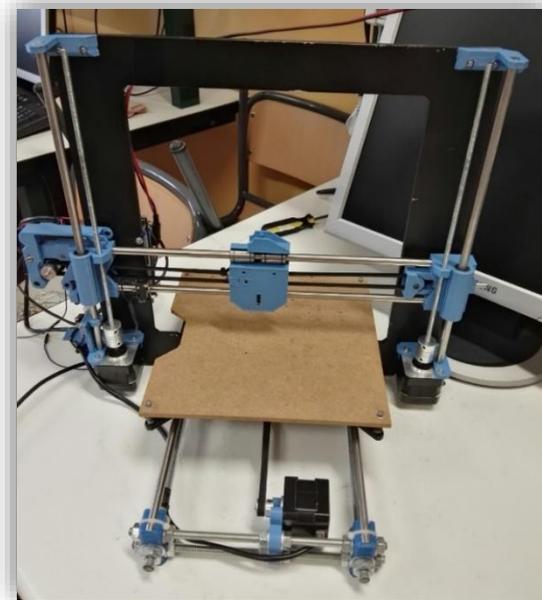


Figura 3.1. Impresora 3D Prusa i3.

- Motor NEMA 17.

Dentro de la impresora Prusa i3 podemos encontrar cuatro motores NEMA los cuales están muy extendidos en el uso de este tipo de impresoras 3D debido a su robustez, a la fuerza que es capaz de soportar, la facilidad para encontrarlo y su buen precio. Este tipo de motor es paso a paso y compuesto por dos bobinas, por lo que posee cuatro cables de salida, uno para cada bobina. Teniendo en cuenta el orden Negro-Verde-Rojo-Azul se pueden conectar en un sentido, o en sentido contrario y de esta manera se puede invertir el giro.

Esta categoría de motores se utiliza en aplicaciones donde lo más importante es conseguir una posición exacta, puesto que posee una resolución de 1,8 grados por paso, es decir, un paso en el motor significa que el eje gira 1,8 grados. De esta forma, para una vuelta completa tiene que avanzar 200 pasos [3].



Figura 3.2. Motor NEMA 17 [3].

- Finales de carrera.

Para poder controlar los movimientos de los ejes, se hace uso de tres sensores final de carrera, uno para cada eje. Estos son los encargados de determinar donde se encuentra la posición mínima y, por tanto, el “Home” total de la impresora. Este modelo de sensor puede ser configurado como normalmente abierto NO, o normalmente cerrado NC. Para este caso en particular, operan como normalmente cerrado. De manera la señal permanecerá en nivel alto hasta que se alcance el final de recorrido, cuando se abrirá el circuito. Esta característica será importante a la hora de configurar el firmware de la impresora.

- Arduino Mega 2560.

Arduino Mega es el encargado de establecer la comunicación entre el ordenador y la máquina. Además, es capaz de interpretar el G-code mediante la instalación del firmware.

Este dispositivo es un microcontrolador basado en el ATmega1280. Posee distintos tipos de entradas y salidas, de las cuales destacan los 54 pines de entrada y de salida, 16 entradas analógicas y 4 puertos serie UARTs. También, posee un conector USB el cual puede ser conectado a un ordenador y que este opere como fuente de alimentación. Igualmente, se puede utilizar una fuente externa que proporcione 5V de tensión en continua. La memoria de este dispositivo es de 128KB de tipo flash para almacenar el código, 8KB de SRAM y 4KB de EEPROM.

Por otro lado, proporciona cuatro hardware UART para TTL de 5V de comunicación en serie. También se puede usar un FTDI FT232RL en los canales de la placa, uno de ellos por USB y los controladores FTDI proporcionan un puerto COM virtual para el software del ordenador [4].

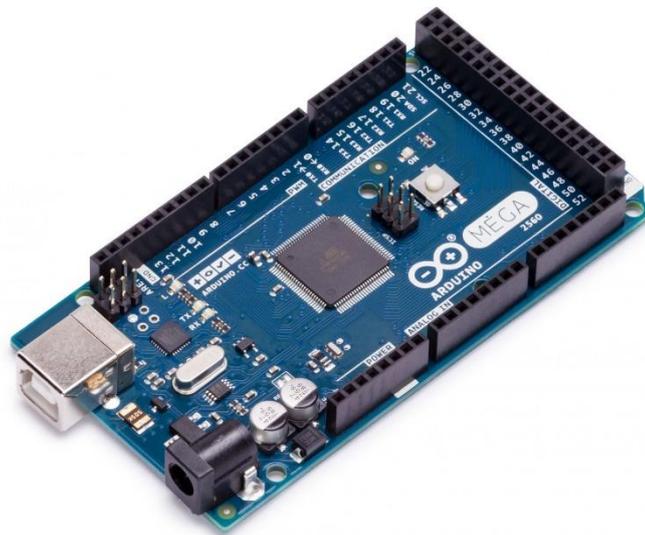


Figura 3.3. Placa Arduino Mega 2560 [17].

- Placa RAMPS 1.4.

RAMPS (*RepRap Arduino Mega Pololu Shield*) es una placa adicional a Arduino que, gracias a los pololus, traduce las órdenes digitales programadas a través del ordenador, en órdenes por pasos para los motores de la impresora. De manera que protege a la placa de Arduino de los elementos de potencia.

En la siguiente imagen se muestran las conexiones que se establecen en este tipo de impresoras destinadas a la elaboración de piezas 3D. En los capítulos posteriores se mencionarán las modificaciones realizadas para la creación de placas de circuito impreso [5].

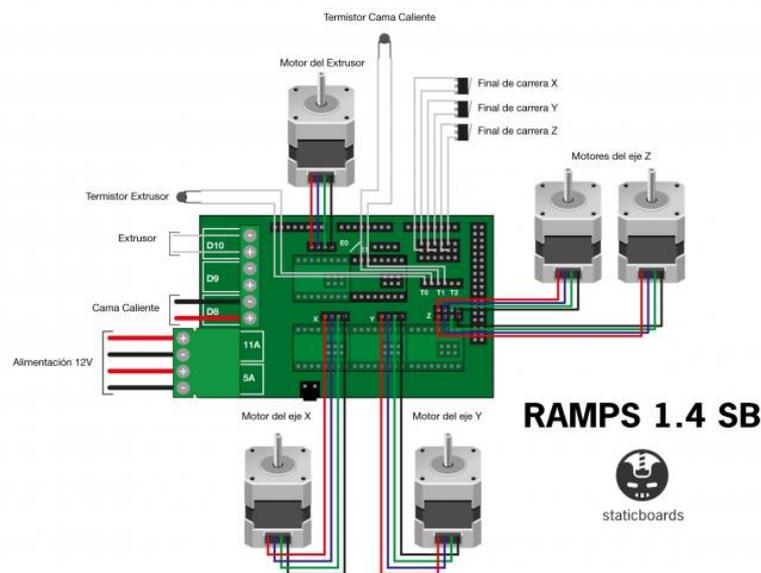


Figura 3.4. Conexiones de la placa RAMPS 1.4 para impresora 3D [5].

- Pololu stepper driver.

El Pololu es un driver o controlador de motores paso a paso. Poseen un potenciómetro y mediante el giro del mismo se consigue modificar la corriente que llega a cada motor.

Por lo tanto, existe un driver de este tipo para cada uno de los ejes con una precisión de hasta 1/16 pasos, es decir, cada paso puede ser dividido en 16 micropasos. Dicha precisión depende del montaje mecánico de la impresora y no de la electrónica empleada, por lo que la máquina debe estar correctamente calibrada.

Este módulo está compuesto principalmente por un circuito integrado, el cual se calienta rápidamente cuando la impresora comienza a moverse debido a la potencia que consumen. Es por ello que se coloca un disipador sobre el mismo, pero esto resulta insuficiente si no existe un ventilador para proporcionar aire frío al circuito y por esta razón se conecta un ventilador externo [3].

2.2. Ventilador externo.

Es un ventilador colocado sobre un soporte de plástico diseñado por una impresora 3D cuya finalidad es que circule aire frío por la placa RAMPS y así evitar un sobrecalentamiento de los drivers. Además, al colocarse sobre toda la placa, se genera una refrigeración total de la misma.



Figura 3.5. Ventilador externo.

2.3. Diodo láser.

El diodo láser es el encargado de insolar la placa fotosensible mediante un haz con una longitud de onda de 405 nm, por lo que se puede apreciar su color azul-violeta. Este tipo de luz es perjudicial para la visión y la piel, por lo que siempre se debe llevar gafas protectoras para evitar tanto el contacto directo con el haz luminoso como la reflexión del mismo sobre cualquier superficie.

El láser está contenido dentro del disipador de calor, el cual, gracias al ventilador, es capaz de mantener la temperatura de trabajo del diodo. Además, este módulo posee una lente ajustable para focalizar la luz y de esta manera aumentar o disminuir el diámetro del spot. El módulo encargado de proporcionar la corriente necesaria para encender y apagar el láser se describe a continuación.



Figura 3.6. Disipador de calor y circuito de control del láser.

2.4. Circuito de control (“Driver Circuit”).

El circuito de control del láser es una fuente de corriente ajustada por una señal TTL. Adicionalmente, proporciona la alimentación del ventilador que se encuentra en el disipador de calor. Este dispositivo será descrito con más detalle en los capítulos posteriores.



Figura 3.7. Circuito de control del láser.

2.5. Fuente de alimentación.

Este elemento es necesario para alimentar la placa RAMPS con 12V. Se debe tener en cuenta que la fuente no tenga un límite de intensidad bajo, ya que cada motor consume 400 miliamperios aproximadamente y se usarán dos de ellos, por lo que consumirá 800 miliamperios tan solo en los movimientos en dos dimensiones.

2.6. Placas fotosensibles positivas.

Las placas fotosensibles son aquellas que poseen una película fotosensible sobre la capa de cobre. Este material se caracteriza por ser sensible a la luz, especialmente a la luz UV, debilitando o fortaleciéndose cuando recibe este tipo de radiación (dependiendo si es positiva o negativa). En este caso se han utilizado placas positivas en las cuales se debilitan las zonas que son alcanzadas por los rayos UV, facilitando así la eliminación del cobre.

3. Software.

El software utilizado durante el desarrollo del proyecto puede dividirse en tres partes: el relacionado con el control de la impresora, tanto para generar y cargar el firmware en Arduino como para enviar el GCODE. Por otro lado, el relacionado con el diseño de las PCBs y el post-procesado para generar el G-code. Y, por último, la aplicación empleada para el diseño de la pieza de soporte. En los próximos apartados se explicarán con detalle cada uno de los entornos.

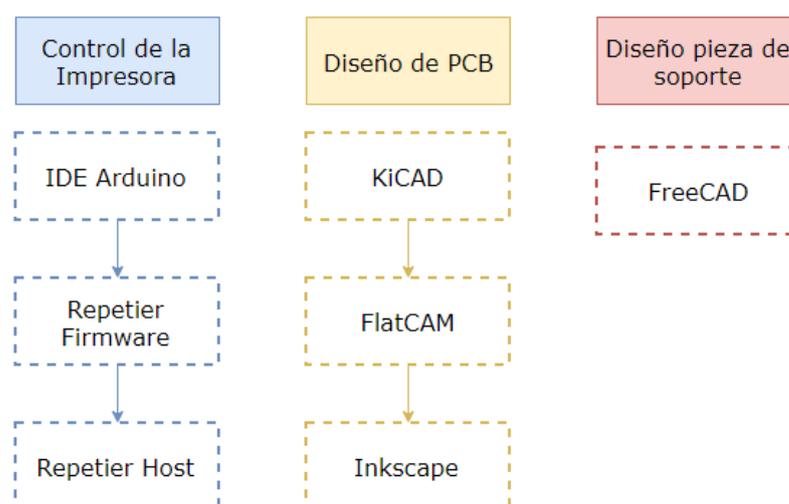


Figura 3.8. Software utilizado.

3.1. IDE Arduino.

IDE Arduino es utilizado para modificar y cargar el firmware *Repetier* en la memoria EEPROM de la placa de Arduino Mega 2560. Asimismo, ha sido útil durante el periodo de pruebas para comprobar el funcionamiento de distintas variables.

En general, *IDE Arduino* es un entorno sencillo que debe instalarse en un PC que permite editar, compilar y cargar programas a la placa a través del puerto serie. La pantalla principal de este programa se divide en: barra de herramientas rápidas, editor de texto y consola de mensajes. Los archivos elaborados en esta aplicación se denominan “bocetos” (“sketches”).

- Estructura.

La estructura de los bocetos elaborados en el software de Arduino se divide en dos funciones principales: `setup()` y `loop()`. En el primero de ellos se inicializan las variables y se configuran los pines como entradas o salidas mediante el comando “pinMode”. Esta función solo se ejecutará una vez, después de cada subida del programa. Por otra parte, en `loop()` se escribe el código que se ejecutará cíclicamente.

- Funciones principales.

Existen dos funciones principales que se utilizan para leer y escribir datos sobre esas variables. Los comandos dependen del tipo de dato que procesan, ya que son distintos para los pines digitales que para los analógicos [6].

3.2. Repetier Firmware.

El firmware es un conjunto de instrucciones de un programa informático, las cuales fijan una lógica que ejerce el control de los circuitos. Se podría decir que es el encargado de traducir las instrucciones codificadas en G-code para producir las acciones de la impresora.

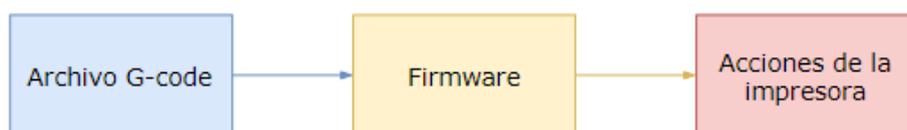


Figura 3.9. Función del firmware.

Durante la ejecución del trabajo se analizaron distintos tipos de firmware, entre los cuales destacaron: *Marlin* y *Repetier*. Ambos han sido creados para funcionar con la placa base Arduino 2560 conectada a la placa RAMPS 1.4, como es el caso de la impresora Prusa i3. Sin embargo, en el firmware no se podía modificar el pin al que se asociaba el diodo láser. Por este motivo, se optó por el firmware *Repetier*, cuyo fabricante pone a disposición una herramienta de configuración.

La herramienta que propone *Repetier* supone una ventaja respecto al resto de software de este tipo porque, a través de una interfaz sencilla, se van completando pasos con la configuración del sistema y finalmente se obtienen los archivos correspondientes al Firmware. Para cargarlo en el sistema tan solo será necesario subirlo a través del entorno de Arduino.

3.3. Repetier-Host.

Repetier-Host es el software que se encarga de cargar los ficheros GCODE generados en *Inkscape*, y enviar los comandos a la impresora para producir el movimiento de la misma. Desde este programa se puede modificar los parámetros característicos de la impresora como el puerto serie al que está conectado y la velocidad de transmisión de los datos. También es posible editar el código G del fichero desde este entorno, así como enviar los códigos directamente, sin necesidad de cargar un archivo.

Aunque la función principal de *Repetier-Host* es la creación de un diseño en dos o tres dimensiones, también puede ser utilizado con el control manual, que permite mover cada eje de forma independiente el número de milímetros que se desee. Así como realizar el “homing”, es decir, el movimiento hacia la posición mínima de cada eje. Por otro lado, para evitar posibles desperfectos cuando se produce un error, existe un botón de parada de emergencia, el cual permite la parada total del sistema.

En definitiva, este programa ha sido utilizado como interfaz para enviar los G-codes correspondientes al diseño de PCBs. Al mismo tiempo, fue usado para el post-procesado de la pieza de soporte y para luego enviarle el diseño a la impresora 3D.

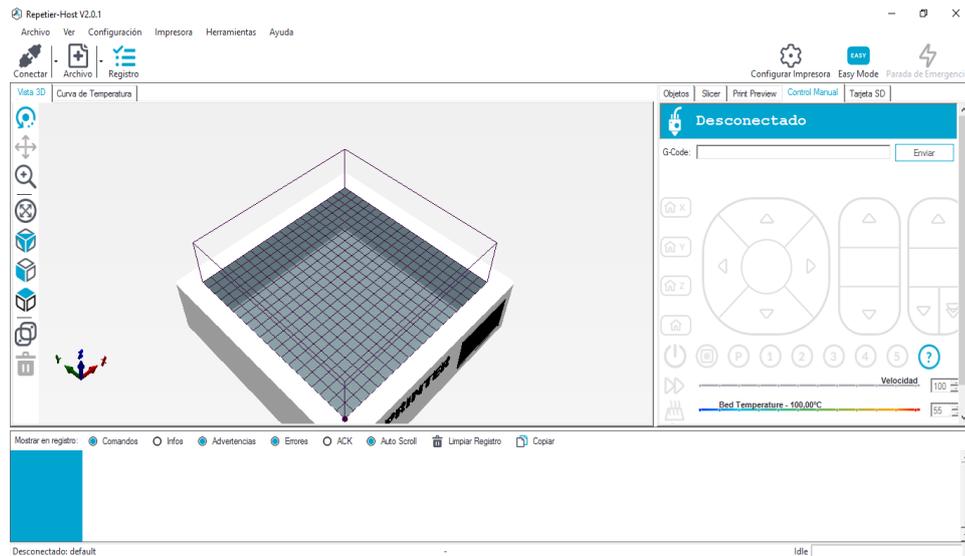


Figura 3.10. Entorno de Repetier-Host.

3.4. KiCad.

KiCad es un software de código abierto que se empleará para la creación de archivos Gerbers, los cuales contienen la información necesaria para efectuar la insolación. De este modo, el proceso comienza con el diseño del circuito y luego se trazan las pistas para establecer las conexiones eléctricas.

En general se trata de una aplicación sencilla, pero a la vez lo suficientemente potente para esta aplicación. Además, dispone de una amplia librería de huellas de componentes que se asociarán a los dispositivos colocados en el esquemático.

3.5. FlatCAM.

FlatCAM es un programa destinado a la preparación de proyectos para la fabricación de PCB con una máquina CNC [7]. En este proyecto se usa para generar la geometría de la ruta que seguirá el láser y posteriormente exportarla como archivo SVG. A pesar de que esta aplicación es capaz de generar directamente el G-code, este no es compatible con *Repetier-Host*.

Para las máquinas CNC, se establece el diámetro de la herramienta y de forma automática calcula la distancia a la que se debe crear la trayectoria para que el ancho de la pista se mantenga tras el paso de la fresadora. En este caso, el diámetro del spot del láser imita al de la fresa.

3.6. Inkscape.

Inkscape es un editor de gráficos vectoriales de código abierto con herramientas de dibujo, de formas o textos. También es capaz de importar imágenes y manipularlas, es decir, cambiar su tamaño, moverlas, rotarlas, agruparlas, etc. El objetivo principal de este programa es ser una herramienta de dibujo potente y cómoda, totalmente compatible con estándares XML y SVG [8]. Uno de los aspectos destacados de *Inkscape* es que se pueden añadir nuevas extensiones o *plug-ins*, y en este proyecto ha sido necesario añadir una de ellas para la generación del G-code a partir de una imagen.

- *Inkscape Laser Tool Plug-In*.

Laser Tool Plug-In es un complemento para la aplicación *Inkscape* para la generación de G-codes. En particular, permite establecer los comandos que encenderán y apagarán el láser, la velocidad de desplazamiento con el láser apagado y la del mismo cuando se encuentra encendido. También se define la potencia a través de un número de 0-255 que modificará el ancho de pulso de la señal PWM (*Pulse Width Modulation*) para controlar el láser.

En resumen, la función principal de *Inkscape* en este proyecto será la importación de cualquier archivo SVG y la colocación del mismo en el área de trabajo. Igualmente, se trabajará sobre la imagen para obtener el código G de la trayectoria.

3.7. FreeCAD.

FreeCAD es un modelador 3D paramétrico, es decir, permite modificar de forma sencilla el diseño regresando dentro del historial del modelo y cambiando sus parámetros. Está principalmente dirigido a la ingeniería mecánica y el diseño de productos.

Este programa es una multiplataforma que se caracteriza por ser de código abierto, por lo que cualquier persona puede acceder a él, personalizarlo, programarlo y subir a la red los resultados. Consecuentemente, va mejorando de forma continua y resulta altamente fiable. Entre sus cualidades destaca la capacidad de leer y escribir formatos de archivos libres tales como STEP, STL, BDX, DAE, etc [9].

FreeCAD es una herramienta que se adapta a las necesidades de este trabajo por diversas razones. En primer lugar, es un software gratuito, por lo que no requiere del pago de una licencia, como es el caso de otros programas CAD. Además, es relativamente fácil de utilizar, pero a la vez

tiene todas las herramientas necesarias, lo que permite una gran agilidad a la hora de crear los diseños 3D.

Por otro lado, admite la exportación de los ficheros con extensión STL, para la posterior impresión. Asimismo, es capaz de generar planos de las distintas vistas en 2D de las piezas en y poder exportarlos en PDF.

Capítulo 4:

Diseño de un sistema de soporte y adaptación del láser.

1. Introducción.

Ha sido necesario diseñar una pieza para que el disipador que contiene el diodo láser esté correctamente colocado en la impresora. Dicha pieza se colocará en el cabezal, en otras palabras, sobre el carro dónde habitualmente se sitúa el extrusor, para que pueda moverse a lo largo del eje x.

Por lo tanto, debe crearse de manera que el espacio entre el eje horizontal y el haz de luz sea el menor posible y que tenga los orificios necesarios para garantizar una correcta sujeción de la pieza. El diseño de la misma se realizará con el software libre *FreeCAD*, teniendo en cuenta las medidas reales del disipador y del carro del extrusor.

Por otra parte, además de las características ya mencionadas, la pieza deberá tener un soporte para colocar la lente. Asimismo, debe poder ajustarse manualmente la distancia de la misma en los ejes x e y, para determinar experimentalmente el centro de la lente y la distancia respecto al área de impresión.

2. Sistema óptico.

La pieza de soporte del disipador debe contener también un sistema adicional para colocar una lente. El propio disipador posee una pequeña lente de enfoque que permite modificar la dirección de los haces de luz que emite el láser. Sin embargo, no posee la precisión necesaria para enfocar todos los rayos en un solo punto. Es por ello que se hace uso una lente convergente, a una distancia adecuada del área de impresión, para disminuir el diámetro del spot.

2.2. Lente convergente.

Una lente óptica tiene la capacidad de refractar la luz y formar una imagen, dependiendo del tipo de lente. En este caso, se quiere que todos los rayos de luz que la atraviesan se concentren en un único punto. Por ello, se usará una lente convergente, en concreto, una planoconvexa.

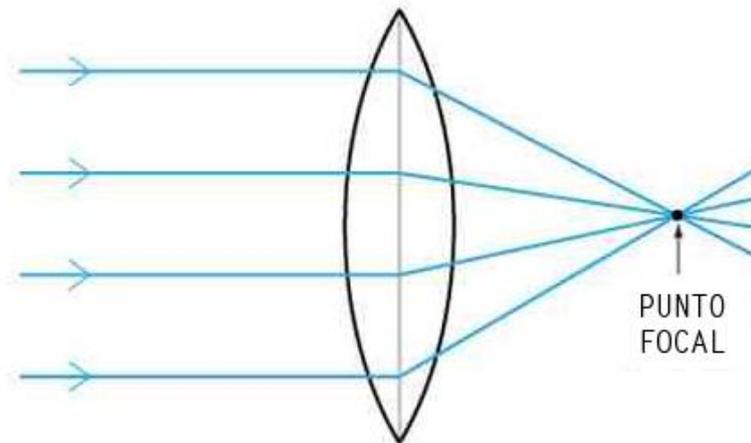


Figura 4.1. Lente convergente [18].

Las características generales de una lente son las siguientes:

- Foco: punto al que llegan todos los rayos refractados por la lente, en el caso de las de tipo convergente.
- Centro de curvatura: es el centro de las dos superficies esféricas por las que está compuesta cada lente.
- Eje óptico: línea horizontal que pasa por el foco y centro de curvatura.
- Centro óptico: centro geométrico de la lente.
- Distancia focal: distancia entre el centro óptico y el foco. Para las lentes convergentes, se define como la distancia desde el eje central de la lente hasta donde el punto en el que coinciden los rayos paralelos [10].

Por lo tanto, para obtener la imagen con el diámetro más pequeño posible se debe colimar el láser y luego situar el plano de trabajo en la distancia focal de la lente. El proceso de colimado se realiza de forma experimental, enfocando el láser hacia el infinito y girando el regulador que posee hasta que se observe un único punto, es decir, que todos los rayos sean paralelos.

De esta forma, la distancia entre el láser y la lente convergente no es relevante. No obstante, el soporte dedicado a la lente tendrá un mecanismo para poder subir o bajarla.

La lente utilizada en este proyecto tiene 2,5 cm de diámetro y una distancia focal de 5cm lo que significa que la distancia entre la lente y la zona de insolación de la impresora deberá ser de 5cm. Para su sujeción se dispone de un orificio con rosca que será útil para atornillarla a la pieza de soporte.



Figura 4.2. Lente. Vista desde arriba.

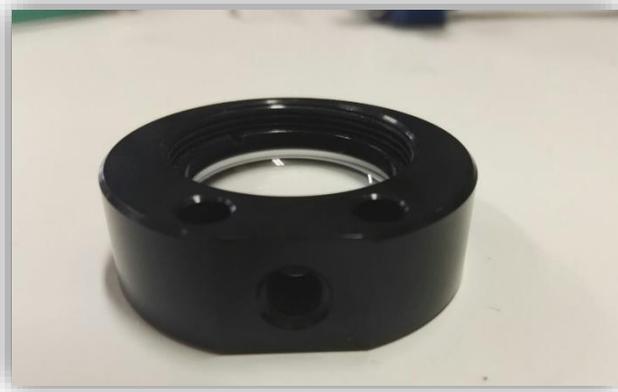


Figura 4.3. Lente. Vista frontal.

3. Diseño de la pieza de soporte.

El primer paso para realizar este proceso es tomar las medidas del carro del extrusor (pieza que está situado sobre el eje x) y del dissipador, así como el diámetro de los orificios de los mismos. El diseño se ha basado en una pieza ya creada para este tipo de aplicaciones que dispone de una base con dos pinzas que sujetarán el dissipador por ambos lados.

- Base.

Las dimensiones de la base serán de 70 x 58mm ya que el ancho y la parte inferior coincidirá con la de la pieza del extrusor, pero será más alta cubriendo donde se encuentran los rodamientos del eje horizontal.

Además, será de un grosor de 10mm porque deberá ser lo suficientemente ancha para contener orificios que no la atraviesen totalmente, con una profundidad de 5mm. Este es el caso de los huecos destinados a los tornillos que unirán la pieza de soporte con el carro del extrusor.

Asimismo, posee cuatro taladros pasantes gracias a los cuales se sostendrá el dissipador. Para estos últimos se ha creado un rebaje en forma de hexágono para que el cabezal del tornillo.

A continuación, se observa la vista de la planta de la base, los cuatro orificios situados verticalmente se corresponden con los del dissipador mientras que los dos restantes son los del carro del extrusor.

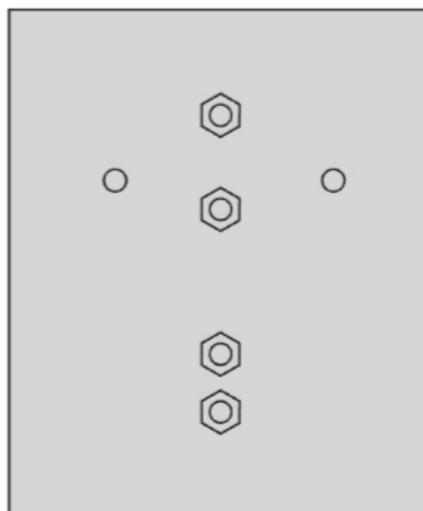


Figura 4.4. Base de la pieza de soporte del láser.

- Pinzas.

Sobre la base se crearon dos pinzas simétricas en forma de “L” para “abrazar” al disipador y así asegurar que este dispositivo permanecerá inmóvil. El alto de las pinzas será igual al ancho del disipador y en la parte superior existirá un hueco por el que pasará un tornillo lo suficientemente largo para conectar las dos pinzas a la vez.

- Soporte para la lente.

El soporte para la lente será una extensión de la base, pero con un ancho de la mitad ya que el elemento que contiene la lente no es concéntrico a ella y necesita 5 mm aproximadamente para que el eje del cristal esté alineado con el haz de luz del láser.

Esta parte adicional tendrá forma trapezoidal para garantizar una mayor rigidez a la hora de fabricarla con la impresora 3D. Igualmente, contendrá un agujero ovalado para poder deslizar el tornillo y, manualmente, ajustar la posición de la lente tanto en el eje horizontal como en el vertical.

Como se ha mencionado anteriormente, la distancia entre el disipador y la lente no influenciará en el diámetro de la imagen creada si el láser está previamente colimado. No obstante, se ha dejado un margen de 20 mm para mayor movilidad. La ranura con forma de óvalo se encuentra en el centro de la pieza con un radio de 3,1 mm.

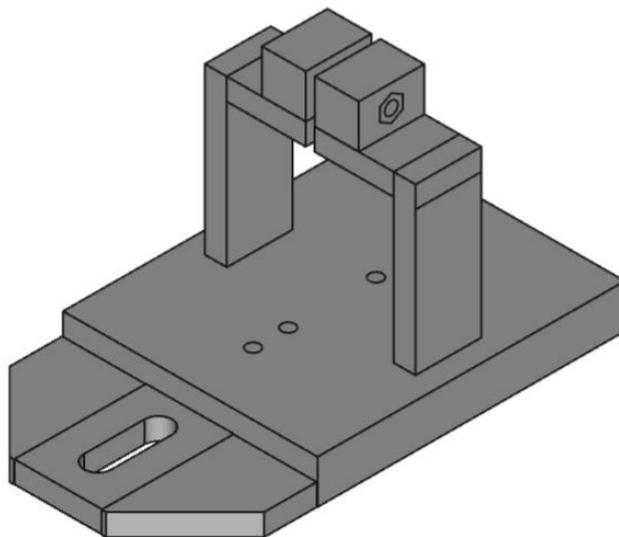


Figura 4.5. Vista axonométrica de la pieza de soporte.

4. Proceso de impresión.

Con respecto a la parte mecánica de la pieza, esta debe ser lo suficientemente resistente como para no deformarse por el calor producido por el láser. Una solución para prevenir este problema es utilizar PLA en el proceso de impresión.

4.1. Material: PLA.

El filamento PLA o Ácido Poliláctico es un polímero biodegradable que se utiliza para la impresión 3D. Entre sus principales características, destaca su gran dureza y rigidez, lo cual dificulta su deformación ante cargas [11]. No obstante, la cualidad por la que se ha seleccionado para este proyecto, es su elevado punto de fusión, siendo de 120 grados Celsius. De esta manera, se garantiza que el plástico mantendrá su forma ya que el láser no alcanzará temperaturas tan elevadas gracias al disipador y al ventilador que tiene incorporado.

4.2. Exportación del fichero STL.

Una vez terminado el diseño en *FreeCAD*, se deben tener visibles todos los elementos que deberán aparecer en la pieza final, ya que se exportará la imagen que esté en la pantalla en ese momento. En primer lugar, se debe garantizar que los orificios se han construido de forma correcta y que las dimensiones son las adecuadas.

Posteriormente, se guarda el proyecto que tiene extensión *.FCstd*, se despliegan los grupos y se seleccionan todos los elementos que conforman la pieza (la imagen de la pieza se volverá de color verde). Seguido de lo anterior, se podrá exportar como fichero STL que se encarga de definir la geometría de los objetos 3D excluyendo información como el color o la textura.

4.3. Creación de capas.

El primer paso para imprimir la pieza será abrirla en el software *Repetier-Host*, en él podemos comprobar si el resultado es el esperado.

Al abrir el fichero STL aparece un error debido a que la parte horizontal de las pinzas no tienen un soporte para ser extruidas y el programa muestra un aviso. Por este motivo, se opta por una impresión al 40% de relleno con soporte en la pestaña de *Slicer*. De esta forma se generan las mallas por capas, necesarias para la construcción de las piezas.

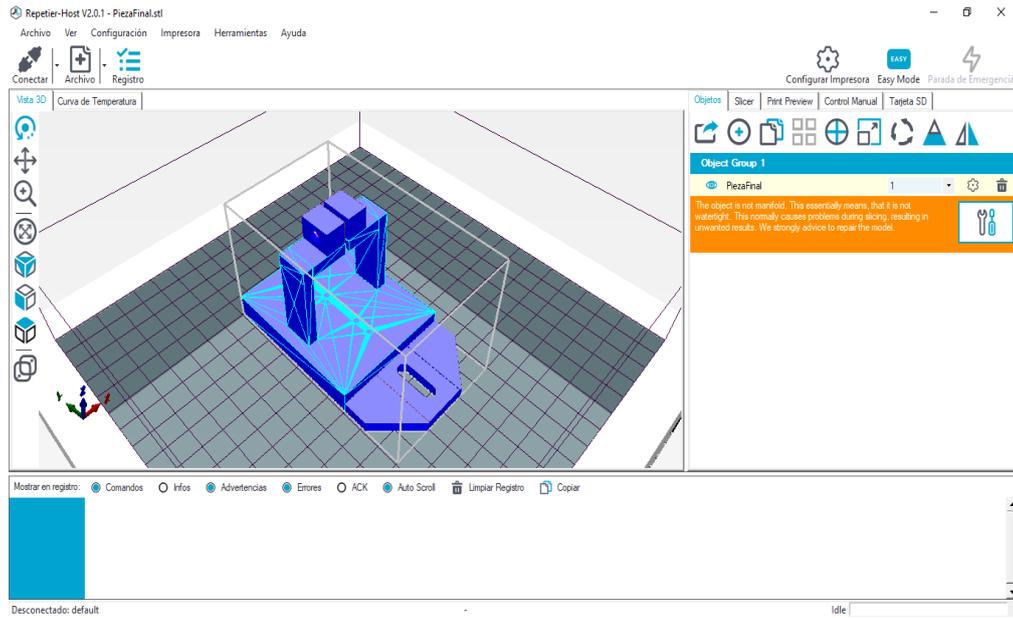


Figura 4.6. Pieza de soporte en Repetier-Host.

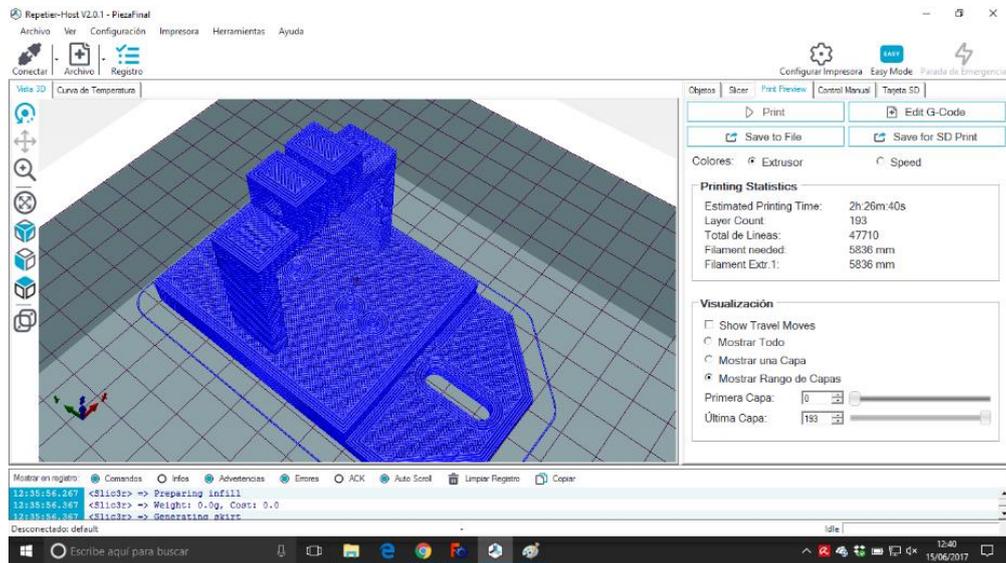


Figura 4.7. Creación de capas de la pieza de soporte.

4.4. Impresión 3D.

Previamente a la impresión se debe dar una capa de laca a la base de cristal de la impresora 3D, para asegurar que el PLA se pegará a la superficie. El siguiente paso es colocar la base en su lugar y el filamento del material seleccionado.

Esta impresora en concreto está configurada para que caliente el filamento a una temperatura de 215 grados Celsius y cuando el sensor alcanza dicho valor hace una prueba que consiste en: ir al origen de coordenadas y extruir un poco de material. Esto se realiza como una prueba previa a la impresión. Se retira el plástico de la prueba y comienza el proceso desde la capa más baja. La impresión de esta pieza tiene un tiempo aproximado de tres horas.

En último lugar, se debe retirar el plástico que forma parte del soporte y lijar los bordes que están en contacto con el disipador del láser para facilitar su colocación en la pieza. En la siguiente imagen se muestra la pieza colocada en el dispositivo final. Se puede apreciar como sujeta el láser y a la lente colimadora.

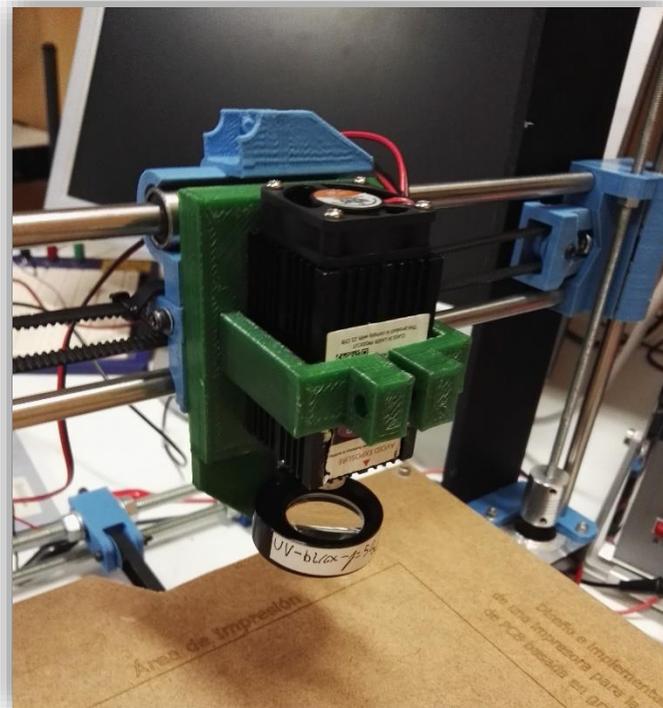


Figura 4.8. Posición final de la pieza de soporte del láser.

Capítulo 5: Circuito de Control.

1. Introducción.

La mayoría de circuitos integrados funcionan con tensiones de alimentación próximas a los 5V, es por ello que necesitan señales de excitación con valores de 0V y 5V. Ocurre lo mismo con el circuito que controla el diodo. De esta manera, se utiliza la modulación PWM para controlar la intensidad que circula por el láser, modificando el ciclo de trabajo de la misma. Por lo tanto, cuando la señal es nula, el láser permanecerá apagado. En cambio, cuando la señal presenta pulsos, se mostrará una luz parpadeante, volviéndose continua cuando el ciclo de trabajo es del 100%.

La señal PWM será generada en uno de los pines disponibles de la placa de Arduino, en concreto el 9, que en la impresión de figuras 3D se destina a la conexión del ventilador. Sin embargo, esta señal es de 12V, por lo que será necesario crear un circuito adaptador de señal para reducir la amplitud a 5V.

Por otro lado, se carece de información con respecto al diodo láser que se utiliza en el proyecto. No obstante, se ha buscado un modelo de láser con las mismas características principales (potencia igual a 500 milivatios y longitud de onda de 405 nm) para conocer el valor aproximado del resto de parámetros. En concreto, se ha tomado como referencia el HL40023MG.

2. Funcionamiento.

El circuito que controla el láser posee dos entradas independientes. En primer lugar, una entrada de alimentación de 12V y por otro lado la entrada para la señal TTL (en la que se aplicará la señal PWM). También posee dos salidas: una de 12 V para el ventilador, y otra denominada GRN para el láser.

Dentro del propio circuito, existe un puente rectificador de diodos que es útil en caso de que se introduzca una señal alterna de alimentación. Este elemento compuesto por cuatro diodos convierte la corriente alterna en continua ya que, a la salida del mismo, se encuentra el circuito integrado MC34063 que requiere este tipo de señales.

El MC34063 forma parte de un convertidor DC-DC para trabajar con menos tensión, es decir, reducir los 12VDC a la tensión en la que funcionan el resto de dispositivos (que suele ser 3,3V o 5V). También, hay dos condensadores que funcionan como desacoplo a los circuitos integrados, cuyo objetivo es desacoplar las señales CA de las CC, por lo que se encargan de eliminar el ruido.

Seguidamente, se encuentra un regulador para proporcionar la tensión necesaria para que el láser opere en condiciones de trabajo.

A continuación, se muestra una imagen del circuito donde aparecen la situación de las entradas (azul) y de salidas (rojo), así como los componentes que forman el circuito.

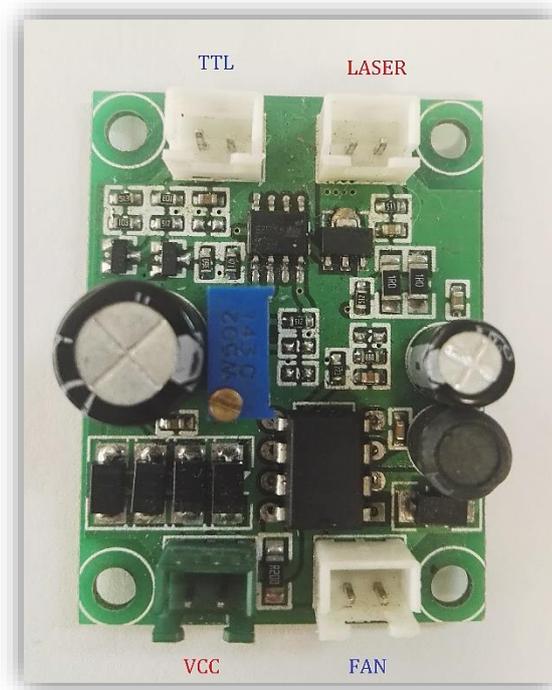


Figura 5.1. Circuito de control del láser.

3. Señal PWM en Arduino.

La señal PWM, o modulación por ancho de pulso, es utilizada en el siguiente proyecto para controlar la potencia del láser a través del circuito de control. Esta es generada en uno de los pines (del 0 al 13) de la placa Arduino Mega e introducida en la entrada TTL del circuito.

Para modificar el ancho de pulso respecto al periodo de la señal, se emplea el comando “`analogWrite(pin, value)`”, siendo ‘pin’ el número del pin destinados para PWM, y ‘value’ un número entre 0 y 255 que representa el porcentaje del ciclo de trabajo [12].

A continuación, se muestran los valores típicos que puede tener ‘value’ aunque puede introducirse cualquier valor dentro del rango.

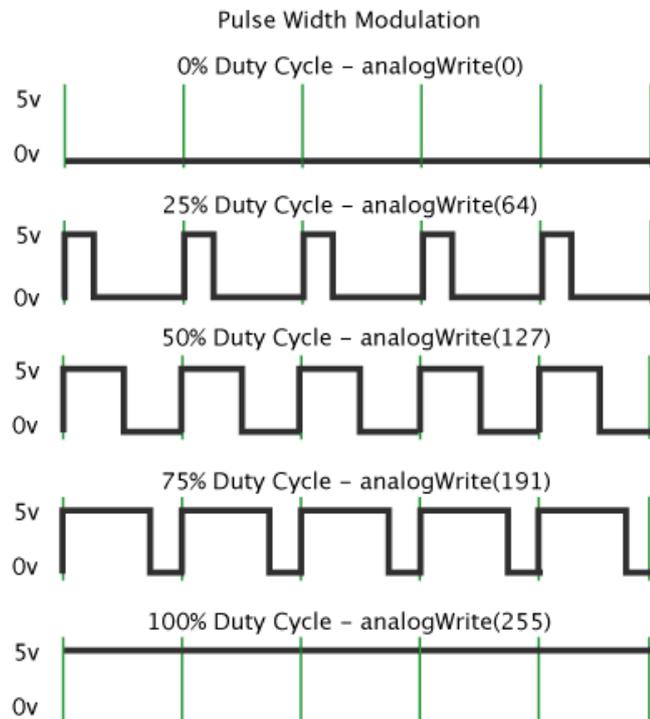


Figura 5.2. Señal PWM con distintos ciclos de trabajo [12].

De esta forma, el comando tan solo modifica el ancho de los pulsos y no la frecuencia de la señal, que por defecto es de 490 Hz aproximadamente. Para esta aplicación en concreto, el valor de la frecuencia será un problema que será mencionado más adelante.

4. Pruebas de la intensidad.

Como paso previo a la conexión del láser en la impresora, se realizó una prueba para determinar la intensidad que circula por él. Para ello, se conecta una resistencia de 1 Ohmio en serie con el láser, a la salida GRN del circuito. De esta manera se medirá la tensión en la resistencia, siendo directamente el valor de la intensidad en ese circuito (puesto que su valor es de 1 Ohmio).

Se seleccionará uno de los pines que están configurados como PWM y se irá modificando el valor del ciclo de trabajo para observar los cambios que se producen en la señal.

4.1. Proyecto en Arduino IDE.

Para realizar la prueba se ha creado un programa en Arduino que genere una señal PWM y desde el cual se pueda cambiar el valor del ciclo de trabajo (que es un valor entre 0 y 255). Por lo tanto, a medida que se aumenta el parámetro “ciclo”, aumenta el ancho de pulso de la señal e igualmente va aumentando la intensidad que llega al láser. Se tomarán valores inferiores a la mitad, es decir, menores a 127, para evitar que la corriente alcance un valor muy elevado.

En la siguiente imagen se muestra el código desarrollado en el entorno de Arduino para realizar la prueba de intensidad.

```
#include <PwmFrequency.h>

#define pin 6 //Pin de PWM con un valor máximo de 31250Hz.
#define ciclo 128 //Valor del ciclo de trabajo expresado de 0 a 255.

void setup() {

  pinMode(pin, OUTPUT); //Se define el pin como salida.

}

void loop() {

  analogWrite(pin, ciclo); //Se genera una señal PWM en el 'pin' con el ciclo de trabajo 'ciclo'.

// Ciclo de trabajo Valor
// 0%(siempre 0V) 0
// 25% 64
// 50% 128
// 75% 191
// 100%(siempre 5V) 255
```

Figura 5.3. Programa en IDE Arduino. Prueba de intensidad.

4.2. Resultados.

En la siguiente tabla, se muestran los resultados obtenidos al modificar el ciclo de trabajo. Se tiene en cuenta que el láser se enciende desde que le alcanza una intensidad de 7,50 miliamperios y que el ventilador consume en torno a 80 miliamperios.

Además, se representan en una gráfica los valores de la intensidad consumida por la fuente y la consumida por el láser.

Variable "ciclo"	Ciclo de Trabajo	Intensidad consumida (mA)	Intensidad láser (mA)
0	0	80	0,00
10	4%	80	7,50
20	8%	90	14,00
40	16%	100	27,70
50	20%	100	35,30
64	25%	110	45,70
70	27%	110	49,40
80	31%	120	54,30
90	35%	120	59,60
100	39%	130	66,80
110	43%	130	75,10
128	50%	150	95,50

Tabla 5.1. Variación de intensidad según el ciclo de trabajo.

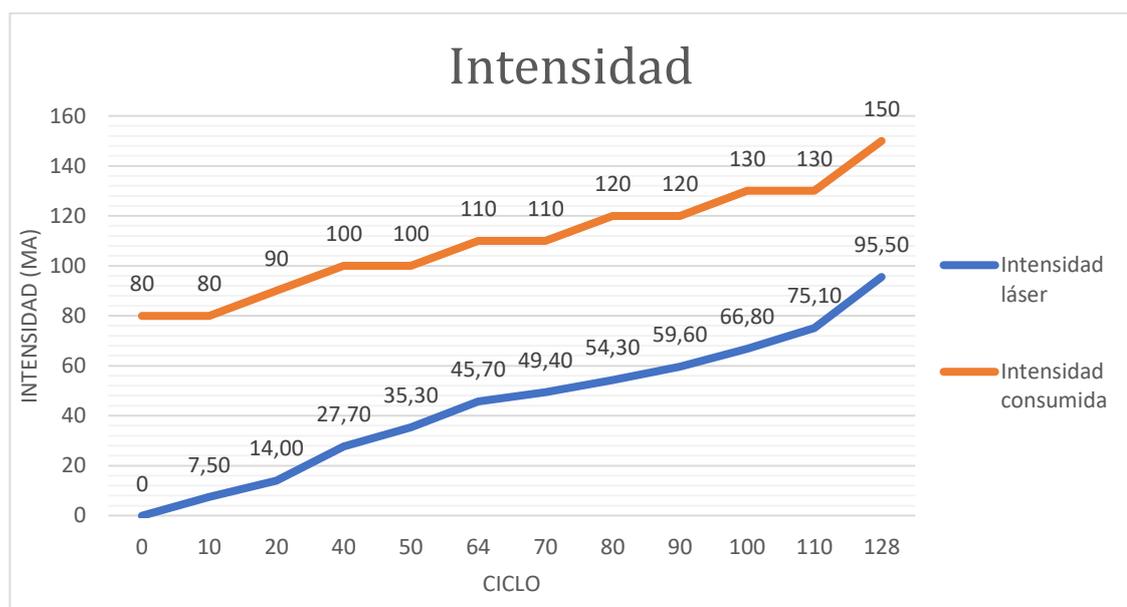


Figura 5.4. Gráfica de Intensidad consumida e intensidad del láser.

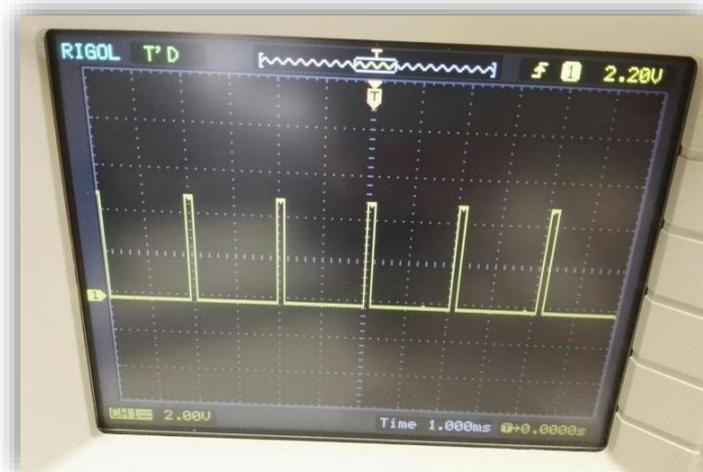


Figura 5.5. Ciclo de trabajo igual al 8%

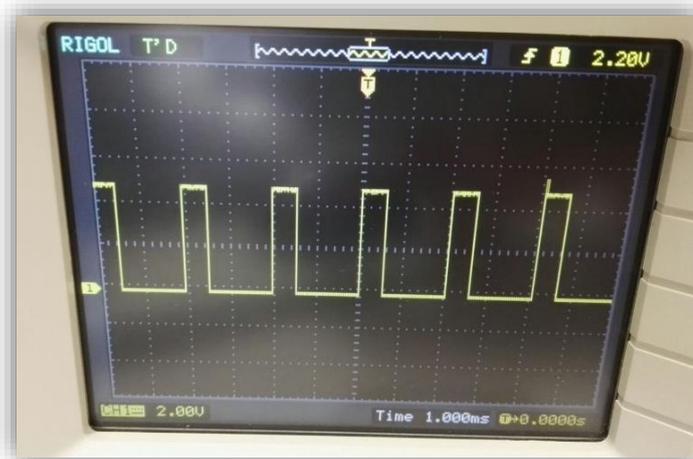


Figura 5.6. Ciclo de trabajo igual al 20%.



Figura 5.7. Ciclo de trabajo igual al 50%.

5. Conexión del láser a la placa RAMPS 1.4.

El circuito posee dos entradas: una de alimentación de 12V y otra de la señal TTL. Estas deberán estar conectadas a la impresora. En primer lugar, se utilizará la propia alimentación de la impresora para el circuito de control.

Por otro lado, es necesario generar y controlar una señal PWM a través de la placa RAMPS. Es por esta razón que se piensa en utilizar el pin 9, el cual está destinado inicialmente para encender y apagar el ventilador en las impresoras 3D. se selecciona este pin en particular, porque tiene asociados los G-codes M106 para encenderlo y M107 para apagarlo. Además, el comando M106 puede ir acompañado de S y dos dígitos que van desde 00 hasta 255 que se relacionan con la potencia del láser ("Laser Power"). Este parámetro puede ser configurado con la herramienta *Laser Tool* para generar el archivo GCODE. De este modo, se puede controlar la intensidad, como se realizó en la prueba anterior.

Sin embargo, el pin 9 tiene una tensión de salida de 12V y el circuito solo admite una señal de 5V de amplitud. Por ello se ha creado un circuito adicional que adapte la señal. Esto resulta ser una solución provisional puesto que no se ha podido conseguir conectar el circuito a un pin de 5V.

6. Circuito adaptador de señal.

El circuito adaptador de señal debe transformar una señal PWM con una amplitud de 12V en una con la misma forma, pero reduciendo la amplitud a 5V.

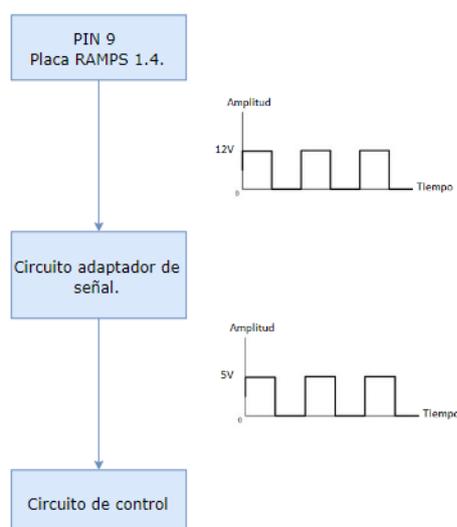


Figura 5.8. Esquema del circuito adaptador de señal.

Para conseguir el resultado esperado se ha empleado un transistor MOSFET (*Metal Oxide Semiconductor Field Effect Transistor*) y una puerta NOT que invertirá y aislará la señal. El transistor MOSFET posee tres terminales: Gate (G), Drain (D) y Source (S) y está basado en la estructura MOS. En concreto, se utiliza uno de tipo N en modo de enriquecimiento.

6.2. Esquemático.

Teniendo en cuenta lo anterior se crea el siguiente circuito, dónde el MOSFET funciona como interruptor y es alimentado a VCC. Esta tensión está proporcionada por un circuito regulador que transforma los 12V de entrada en 5V. Se recomienda colocar un condensador de 0,1 microfaradios cerámico para eliminar los ‘picos’ que puedan ser generados por la fuente de alimentación.

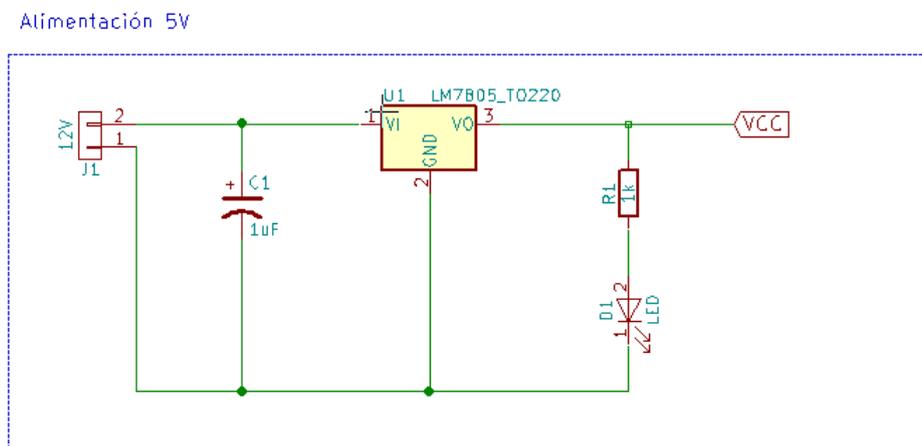


Figura 5.9. Esquemático del bloque de alimentación.

La parte del circuito encargado de reducir la amplitud funciona de la siguiente manera. Cuando la tensión de entrada (VIN) en la puerta (G) es cero, el dispositivo entra en corte y la tensión de salida (VOUT1) es igual a la tensión de alimentación (VCC). En cambio, cuando a la entrada se aplica una tensión alta o igual a VCC, la corriente de drenaje aumenta y se convierte en un valor constante que solo depende de VGS. Por lo tanto, el transistor se comporta como un interruptor cerrado y la tensión VOUT es nula.

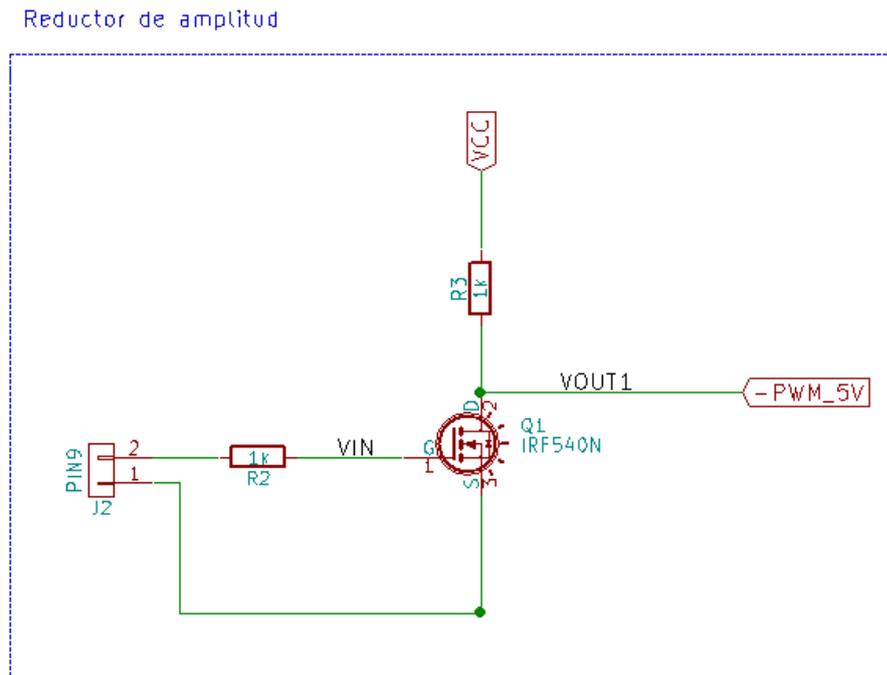


Figura 5.10. Esquemático del bloque reductor de amplitud.

Sin embargo, de esta manera se genera la señal con una amplitud de 5V, pero invertida. En otras palabras, cuando la señal que proviene del pin 9 está a nivel alto, VOUT1 está a nivel bajo y viceversa.

Asimismo, es necesario aislar este circuito del de control del láser ya que en caso contrario, el láser se enciende a máxima potencia y nunca llega a apagarse. Esto es debido a que la tecnología que emplea el regulador de tensión no es compatible con la del circuito y éste tan solo observa una señal distinta de cero a su entrada. Por este motivo, se emplea una puerta NOT con la cual volvemos a invertir la señal, para obtener la misma que en el pin 9 pero reduciendo la amplitud y al mismo tiempo permanece aislada.

En concreto, se utilizó un 74LS04, el cual es un circuito integrado de la familia TTL con seis puertas NOT. La entrada de cada una de ellas tiene una resistencia muy alta, lo que significa que no les afecta la parte del circuito donde se conecten. Este dispositivo provoca una reducción adicional de la amplitud, obteniendo una salida de 4V, lo cual no resulta un problema ya que esta tensión es suficiente para el sistema.

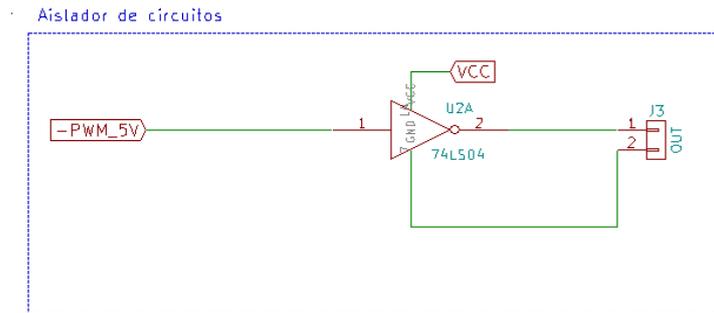


Figura 5.11. Esquemático de la puerta NOT.

6.3. Montaje en placa de prototipo.

El circuito creado es un añadido al circuito de control del láser, por lo que resulta ser una primera versión mejorable de la electrónica de la impresora. Por esta razón se ha decidido montar sobre una placa de prototipado de uso temporal, en concreto una de cintas o stripboard, dónde los orificios están conectados formando filas de material conductor.

El procedimiento de montaje consiste en buscar todos los componentes necesarios, colocarlos para elegir la distribución óptima, y por último soldarlos a la placa. Los dispositivos se han ido montando por bloques para comprobar el correcto funcionamiento. Adicionalmente, se ha añadido una salida de 12V que alimentará al circuito de control del láser.

Finalmente, se han conectado las entradas y las salidas y se ha verificado que la señal se salida del circuito es igual en forma a la de entrada, pero con una amplitud de 4V aproximadamente. La señal PWM de la placa RAMPS presenta ciertas oscilaciones cuando está a nivel bajo y estas son corregidas por el MOSFET, obteniendo así una señal más limpia.

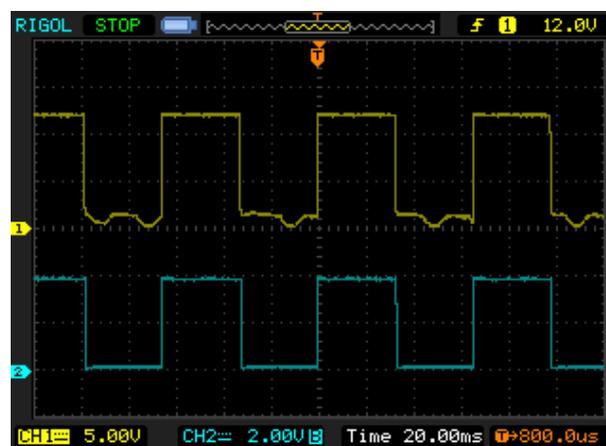


Figura 5.12. Captura del osciloscopio. Señal Pin 9 y señal adaptada.

Además, se ha tenido en cuenta que debe estar sujeta a la estructura de la impresora, por lo que su tamaño debe ser reducido. A continuación, se muestra el resultado de la placa de prototipado con el circuito adaptador de señal.

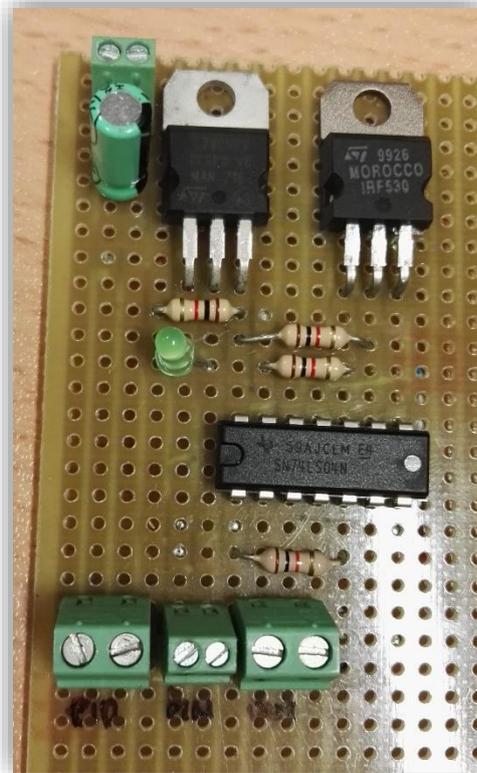


Figura 5.13. Montaje final del circuito adaptador de señal.

Capítulo 6: Puesta en Marcha

1. Introducción.

En este capítulo se explicarán los procesos realizados para poner en marcha la impresora 3D para la insolación de PCB. Previamente ha sido necesario configurar el firmware de *Repetier* para modificar las velocidades de los ejes y definir los pines a los que se conectará el circuito del láser y el ventilador externo.

Finalmente, se explicará de forma detallada el proceso completo de impresión, desde el diseño de las placas hasta los códigos que son enviados a la impresora.

2. Configuración del Firmware Repetier.

La configuración del firmware *Repetier* se puede hacer mediante una herramienta online que propone el fabricante, la cual facilita el proceso. A continuación, se mencionan las características que se han sido modificadas.

2.1. Caracterización de la impresora.

En la pestaña de configuración general se definirá el tipo de procesador, que en este caso es Arduino Mega a la que se conecta la placa base: RAMPS 1.4. Además, se establece el puerto serie por defecto y una velocidad de transmisión de datos de 250000 baudios.

Luego, se determinan los puntos mínimos de cada eje y la longitud máxima de cada uno de ellos. La zona de impresión de la impresora es cuadrada de 220x220 mm, a excepción de una cavidad en forma de trapecoide en uno de los lados. Por eso se establecen las dimensiones con un valor de 200 mm siendo ligeramente inferior al tamaño real.

Para la corrección del eje z no se realiza ningún cambio puesto que la insolación de las placas se realiza en dos dimensiones.

2.2. Motores y sensores. Resolución.

El eje horizontal y el vertical usan un motor paso a paso y correas para efectuar el movimiento. Por lo tanto, la resolución de cada eje dependerá del ángulo/paso del motor, el microstepping del driver, el paso de la correa y el número de dientes de la misma. El software *Repetier-Host* tiene una herramienta que permite el cálculo de los pasos por milímetro.

	Eje X	Eje Y
Ángulo/paso del motor	1.8	1.8
Microstepping del driver	1/16	1/16
Pasos de la correa	2 mm (GT2 Belt)	2 mm (GT2 Belt)
Número de dientes	20	20
Pasos por mm	80	80

Tabla 6.1. Cálculo del número de pasos por milímetro de los motores.

En el Firmware también se debe definir la velocidad máxima de desplazamiento y la velocidad del “homing” que será inferior para garantizar el contacto con el sensor. También, se establece la aceleración de desplazamiento y la de impresión, siendo la de desplazamiento superior para que el movimiento sea más rápido en los cambios de dirección.

Por otro lado, hay que tener en cuenta que la impresora solo dispone de tres sensores que están colocados en la posición mínima de cada eje. Los cuales están configurados como interruptores normalmente cerrados y conectados en el lugar correspondiente de la placa RAMPS, es decir, el del eje x en el pin 3, el eje y en el pin 14 y el eje z en el 18.

2.3. Control del láser.

El firmware de *Repetier* posee un apartado para añadir herramientas, como un láser o una fresadora. Sin embargo, este tipo de configuración solo soporta un todo/nada mediante el código M3 para encender y M5 para apagar y no permite la variación del ancho de pulso de la señal PWM.

En el proceso de insolación de PCB, la intensidad del haz de luz será determinante puesto que variará según el tipo de placa que se utilice. Por ello, el control del láser se realizará mediante el FAN_PIN que es el que se emplea en las impresoras 3D para encender, apagar y cambiar la velocidad del ventilador y conectándolo en el pin número 9. Este genera una señal PWM con una amplitud de 12V, pero el circuito adaptador de señal reducirá este valor para poder introducirla en el circuito de control.

2.4. Ventilador externo.

Como ya se ha mencionado, los drivers de los motores, colocados en la placa RAMPS 1.4, poseen un circuito integrado que tiende a calentarse. Para prevenir que sobrepase el límite térmico se coloca un ventilador externo que se conectará al pin asociado al FAN_BOARD_PIN. Este se caracteriza por estar activado mientras lo están los motores.

3. Correcciones mecánicas.

La impresora 3D fue construida como una réplica del modelo Prusa i3, pero nunca fue puesta a funcionar. Por este motivo, se han tenido que hacer una serie de modificaciones.

En primer lugar, el sensor final de carrera del eje Y se encontraba en una posición donde al accionador no presionaba correctamente el contacto móvil. Como solución a ello se rodó el sensor y se fijó el accionador con silicona caliente en el lugar adecuado.

Igualmente, cuando se enviaba el comando G28 (“homing”) el carro del extrusor iba en sentido negativo del eje x hasta tropezar con uno de los tornillos que sujetan a la placa de Arduino y nunca tocaba el sensor. Por lo tanto, se optó por rodar el final de carrera en sentido positivo del eje hasta una posición donde la pieza llegase sin problemas.

Por otro lado, las conexiones correspondientes al motor del eje y se encontraban en invertidas, es decir, se movía en sentido contrario. Para solucionarlo solo fue necesario cambiar el sentido en el que estaban conectadas las fases del motor.

Con respecto al ventilador externo, para que el aire estuviese dirigido hacia la placa, se colocó paralela a la RAMPS, lo cual es posible gracias al soporte. Este irá sujeto en la parte de la carcasa de la impresora, justo encima de la placa.

4. Área de impresión.

El siguiente paso tras configurar el firmware y realizar las modificaciones mecánicas, es determinar el área de impresión en el que se colocará la placa. Se debe tener en cuenta que el origen de coordenadas ha variado debido al grosor de la nueva pieza más la distancia de la misma al láser. Por ello, se realizaron varias pruebas experimentales para conocer el punto en *Inkscape* que coincida con el (0,0) de *Repetier Host*.

En primer lugar, se dibujó un cuadrado de 50x50mm, se colocó en el origen y se imprimió. De esta manera, se pudo deducir que la escala de *Inkscape* no se corresponde con la realidad. Es decir, 50 mm en el programa son 14 mm en la realidad, por lo que la escala real es 3,57 en el eje horizontal y en el vertical.

Con respecto al origen de coordenadas, con la escala por defecto, se genera en el punto inferior izquierdo de la página. Sin embargo, al modificar dicho valor, el programa lo coloca a -11,783 mm del punto anterior. La extensión lo crea de forma automática, por lo que no se puede modificar esta característica. Por esta razón, se crea una plantilla con el área de impresión, de manera que cualquier imagen que se genere dentro de este espacio, será impresa en el plano real.

4.1. Creación de la plantilla.

Se ha definido el área máxima de impresión como un espacio de 150x150mm. Se dibuja un cuadrado y se sitúa en el nuevo eje de coordenadas, en otras palabras, en el punto (0,-11,783). Se eliminan los bordes de la página, se aumenta el grosor de la silueta del cuadrado y se cambia su color para una mejor visualización. El cuadrado ha quedado guardado en la capa que genera el software por defecto, la cual se bloqueará para que no pueda ser modificada, por lo que para añadir un elemento se deberá crear una nueva capa. Finalmente, se guarda el archivo como una nueva plantilla de *Inkscape*. De esta forma, se obtiene una plantilla para importar cualquier tipo de imagen dentro del área de impresión.

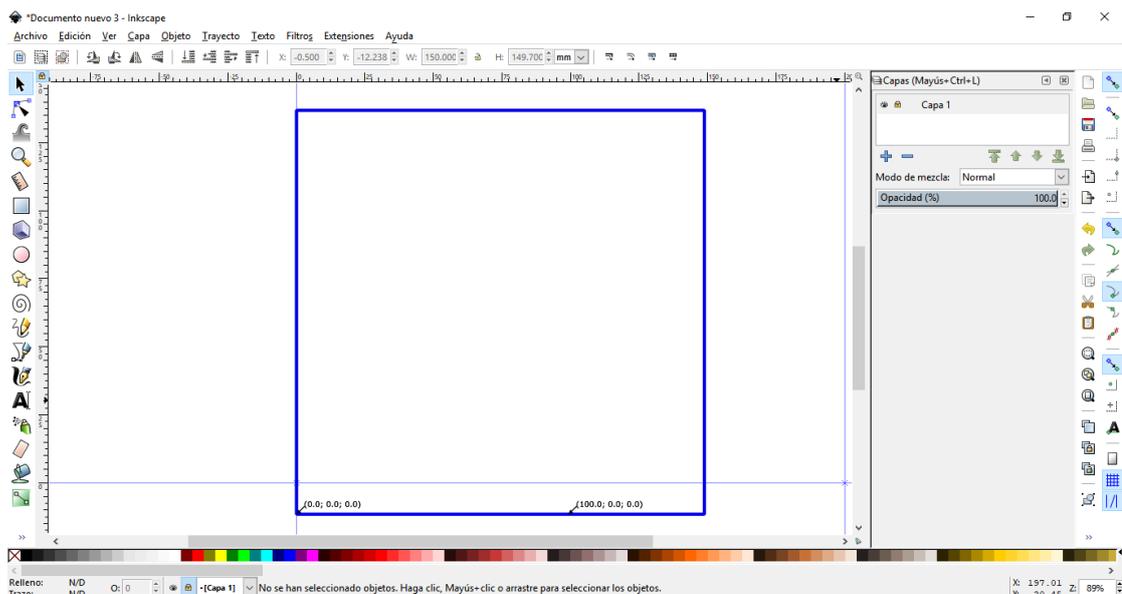


Figura 6.1. Captura de la plantilla: Área de impresión.

4.2. Sistema de referencia.

En el caso de que se pretenda crear una PCB de doble cara o se quiera realizar la serigrafía deberá existir un mecanismo que asegure la posición exacta de la placa. Por esta razón, se ha diseñado una pieza en forma de "L" con un centímetro de ancho que será situada sobre los ejes, coincidiendo el vértice de la pieza con el origen de coordenadas del área de impresión de *Inkscape*.

De esta forma, el proceso de fabricación sería el siguiente. Primero, se coloca la placa en el sistema de referencia, con la cara que se desea insolar hacia arriba y se envía el archivo de comandos a la impresora. Por lo tanto, se deberá crear el diseño en el lugar equivalente en el área de impresión.

Cuando el procedimiento de insolación haya terminado, si se desea insolar la cara posterior, solo es necesario darle la vuelta y colocarla en el mismo lugar. Se debe tener en cuenta que se debe aplicar espejo en la imagen para que coincida con la primera cara, y luego enviar los nuevos datos a la máquina. Para el caso de la serigrafía, se situará la imagen justo encima del circuito, en una nueva capa del programa, y así se podrá comprobar el emplazamiento de las huellas respecto a los componentes del circuito.

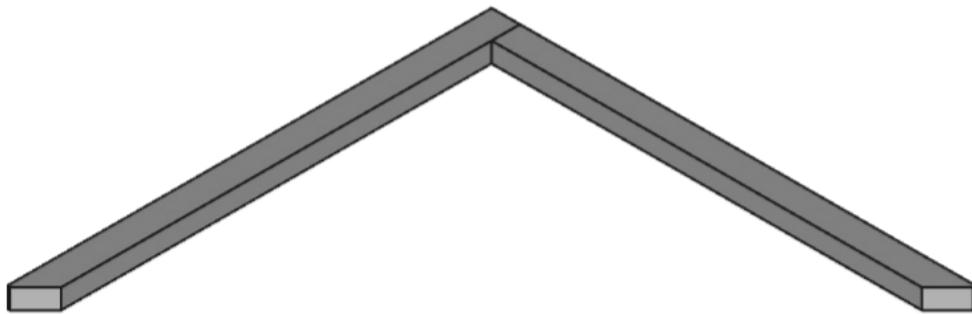


Figura 6.2. Sistema de referencia.

5. Proceso de diseño e insolación de PCBs.

El proceso de insolación comienza con el diseño del circuito, tras el cual se genera el fichero Gerber que contiene la información acerca del mismo. Luego, será necesario crear la trayectoria del láser y, por último, crear el fichero de código G. Este último será interpretado por el firmware para producir las acciones de la impresora.

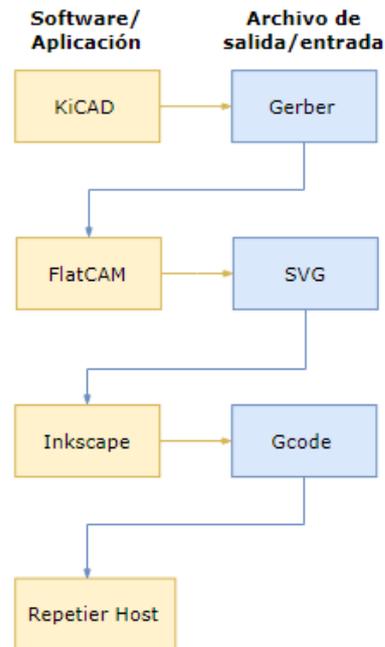


Figura 6.3. Proceso de insolación.

5.1. Diseño del esquemático y Layout.

El diseño de la PCB se ha realizado con el software *KiCad*, por lo que el primer paso consistirá en generar un nuevo proyecto y dentro del mismo abrir el editor de esquemas electrónicos.

- Dibujar esquema electrónico.

El esquema se creará en una página con un cajetín donde se puede escribir información acerca del diseño. Luego, se añaden los componentes y se conectan a través de un cable o mediante un bus. Además, se puede agregar la alimentación, tierra y etiquetas para conectar dos puntos de circuitos que se encuentran separados en el plano.

Cada dispositivo posee un valor y una referencia, esta última se enumerará de forma automática cuando se anoten todos los componentes. Una vez realizado esto, se

ejecutará un control de las reglas eléctricas. El siguiente paso es asociar cada elemento con el Footprint correspondiente, para ello se abre CvPcb. Se puede buscar por librerías, palabras o número de pines. Finalmente, se debe generar el netlist para *Pcbnew*.

- *Pcbnew*.

Al igual que en el esquemático, aquí se dispone de una página con cajetín configurable. En primer lugar, se recomienda definir la separación y el ancho de pista mínimo en las reglas de diseño. Seguidamente, se podrá importar el fichero netlist y de esta forma aparecerán todos los componentes conectados a través de un hilo fino, denominado *ratnest*. Se define el borde de la placa en la capa Edge.Cuts y se colocarán teniendo en cuenta las reglas de diseño. A continuación, se selecciona la capa en la que se quieren dibujar las pistas, las cuales podrán ser creadas de forma manual o automática. En la parte inferior se puede observar el número de conexiones restantes.

Generalmente, el siguiente paso sería generar un plano de tierra, pero en este caso no es necesario porque el cobre solo será eliminado de los bordes de las pistas y los pads. Por último, se realiza un chequeo de las reglas de diseño para asegurar que no se ha producido ningún error.

- Generación de archivo Gerber.

Una vez finalizada la PCB, se pueden generar archivos Gerber para cada capa. El proceso resulta sencillo, tan solo hay que seleccionar las capas y las opciones más adecuadas. *KiCad* también dispone de un visualizador de archivos de este tipo.

- Visión 3D.

Como elemento adicional, *Pcbnew* cuenta con un visualizador 3D que muestra una simulación del resultado final del circuito creado.

5.2. Trayectoria del láser.

El archivo Gerber es un listado de comandos que define la forma de las pistas y los pads. No obstante, el láser debe recorrer la silueta de estos elementos sin modificar su ancho. Por esta razón, se debe crear la trayectoria a una distancia igual al radio del spot.

El software *FlatCAM* es capaz de analizar el archivo Gerber y generar la ruta de insolación a partir de los siguientes parámetros: diámetro de la herramienta, d ; ancho o número de pasadas y la fracción del diámetro de la herramienta que se supera en cada pasada, θ . De esta manera se relaciona el área de cobre eliminada con el número de pasadas, puesto que, en cada una de ellas, el láser pasará a una distancia inferior al radio [14].

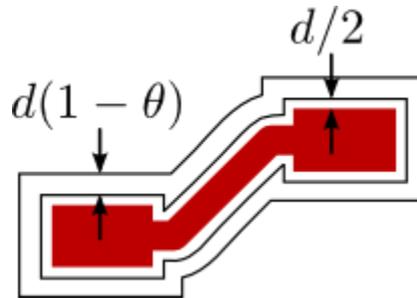


Figura 6.4. Forma de calcular la trayectoria del láser [14].

Una vez se definan los parámetros mencionados, se genera la geometría y se crea un archivo con extensión *btl_iso* por cada una de las pasadas. En caso de configurar más de pasada, se deberá exportar a formato de gráficos vectoriales cada uno los dibujos creados.

Aunque *FlatCAM* está diseñado para generar directamente el código G, lo hace mediante las instrucciones M3 y M5, que no son aplicables al láser ya que será controlado con los comandos M106 y M107.

5.3. Obtención de G-Code.

Para obtener el listado de comandos GCODE, se emplea el programa *Inkscape*, en concreto, la extensión creada por J Tech Photonics: *Laser Tool*.

El primer paso consiste en la importación del fichero SVG (*Scalable Vector Graphics*) dentro del área de impresión. Es importante no modificar el tamaño de la imagen ya que se corresponde con las dimensiones reales diseñadas en *KiCad*. Para poder observar la trayectoria de la insolación se eliminará el relleno de la imagen y se seleccionará un borde negro. Finalmente, se crea el trayecto a objeto y se aplica la herramienta para la generación del GCODE, donde se pueden modificar los siguientes parámetros:

- Laser ON Command: comando para encender el láser.
- Laser OFF Command: comando para apagar el láser.
- Travel Speed: velocidad de la máquina cuando el láser se encuentra apagado.
- Laser Speed: velocidad de la máquina cuando el láser está encendido.
- Laser Power: control de la señal PWM mediante la modificación del ancho del pulso.
- Power-On Delay: se encenderá el láser y esperará este tiempo para iniciar el movimiento.
- Passes: número de pasadas en las que el láser repetirá todo el trayecto.
- Pass Depth: el eje z se moverá esta distancia en cada pasada.
- Directory: directorio para guardar el archivo GCODE.
- Filename: nombre del archivo.
- All Units: unidades [15].

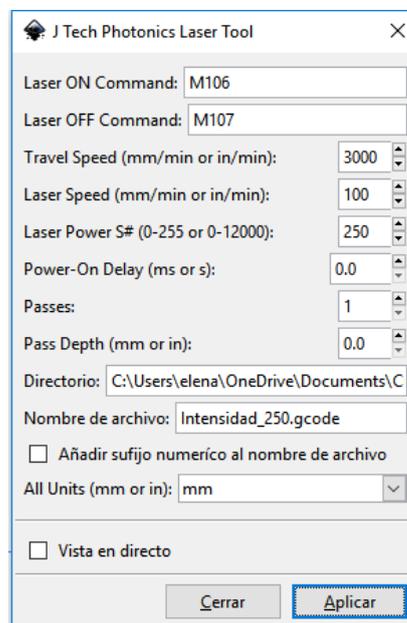


Figura 6.5. Captura de la herramienta Laser Tool.

5.4. Insolación.

Una vez generado la lista de G-code, que definen los movimientos que seguirá el láser y la configuración del mismo y se carga en el software Repetier. En este programa se puede visualizar el trayecto total y enviarlo a la impresora.

Antes de la impresión se deberá conectar *Repetier-Host* a la impresora. Para ello se introduce el puerto por el que está conectada la placa de Arduino y la velocidad de transmisión de datos. Además, durante el proceso de impresión se podrá ver el transcurso del mismo y el tiempo restante.

Capítulo 7: Pruebas y Resultados.

1. Introducción.

En este capítulo se describirán las pruebas realizadas con el láser tras ser colocado en la pieza de soporte que ha sido diseñada previamente. Gracias a la herramienta de *Inkscape*, se puede generar el código G de cualquier imagen, no solo de archivos Gerber. Por lo que se ha empleado esta cualidad para crear dibujos sencillos con los cuales se conocerá la velocidad, la intensidad y la posición adecuada del dispositivo. Además, se han utilizado distintos materiales para comprobar el efecto en cada uno de ellos antes de empezar a proyectar circuitos electrónicos en placas fotosensibles.

El objetivo final de las pruebas es establecer la configuración total del sistema para la insolación de PCB.

2. Intensidad.

Como se ha expuesto en capítulos anteriores, la intensidad del haz del láser depende del ciclo de trabajo de la señal PWM, el cual es modificado a través del comando M106. Este valor que va desde 0 hasta 255 es configurado con el parámetro "Laser Power" disponible en la extensión *Laser Tool*. De esta forma, cuando mayor es el valor, mayor es la intensidad.

En primera instancia, se desconoce el valor adecuado para realizar la insolación de las placas. Por este motivo, se lleva a cabo una prueba compuesta por cuadrados y texto, dónde se genera un fichero GCODE por cada cuadrado junto al título correspondiente.

Esta prueba se realiza primero sin la lente, con valores desde 90 (ciclo de trabajo: 35,29%) hasta 210 (ciclo de trabajo: 82,35%). Del mismo modo se ejecuta, pero con la lente situada en la pieza de soporte, pero ahora alcanzando valores más elevados.

Como resultado de la prueba, se observa que con valores inferiores a 150 se producen líneas discontinuas. Esto es debido a que la señal que llega al láser es la señal PWM generada. De esta manera, cuanto más alto es el ciclo de trabajo, más unidos están los puntos llegando a formar líneas continuas.

Por otro lado, comparando los resultados obtenidos, no se aprecian diferencias si el proceso se ha realizado con la lente o sin ella.

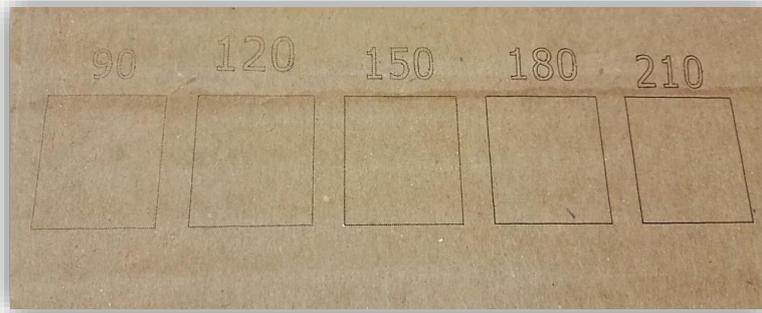


Figura 7.1. Prueba de intensidad sin lente.

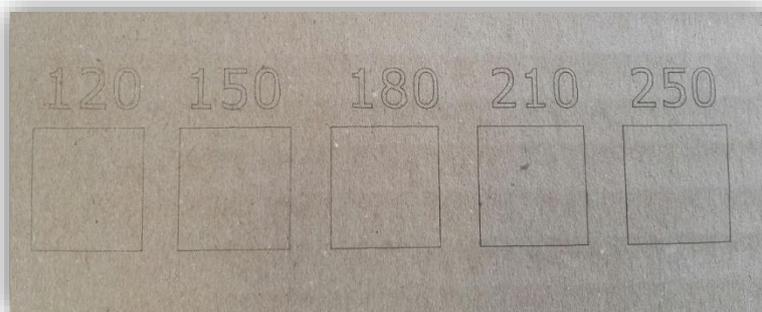


Figura 7.2. Prueba de intensidad con lente.

Para evitar los trazos discontinuos con ciclos de trabajo bajos, se han pensado varias posibles soluciones, teniendo en cuenta que no se puede modificar fácilmente la frecuencia de salida del pin de Arduino. Por lo tanto, se pretende reducir la velocidad o aumentar el número de pasadas para unir lo máximo posible los puntos generados.

2.1. Modificación de la velocidad de trazado.

Para obtener líneas continuas con ancho de pulsos más bajos, se pensó en reducir la velocidad. Para ello se realizó una prueba que consistió en escribir “Velocidad” y el número en milímetros por minuto para cada ensayo. Igual que en el caso anterior, fue necesario generar un archivo G-code por cada velocidad. En este caso, se efectuó utilizando la lente y sobre una cartulina azul para poder ver los detalles con claridad.

Con velocidades inferiores a 100 mm/min no se perciben cambios ni en el resultado mostrado ni en los movimientos que realiza el láser. En cambio, cuando aumenta la rapidez se comienzan a

ver discontinuidades hasta no dejar marca, como es el vaso de 1000 mm/min. Por lo que la velocidad no podrá ser una alternativa a intensidades elevadas.

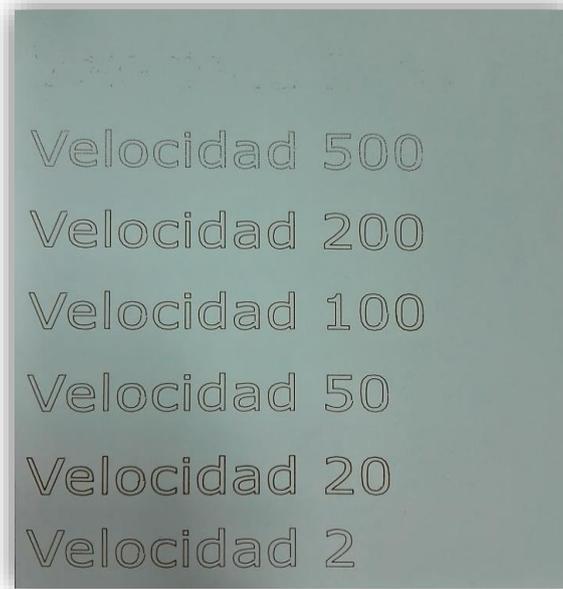


Figura 7.3. Prueba de velocidad de trazado con la lente.

2.2. Número de pasadas.

Con la herramienta *Laser Tool* se puede configurar las veces que se quiere que el láser realice la trayectoria completa. Por consiguiente, cuanto mayor sea el número de pasadas, más marcado quedará el dibujo. Sin embargo, con valores bajos de intensidad, cada pasada coloca el punto justo en el mismo lugar. Por lo que, tampoco es una solución al problema de discontinuidad.

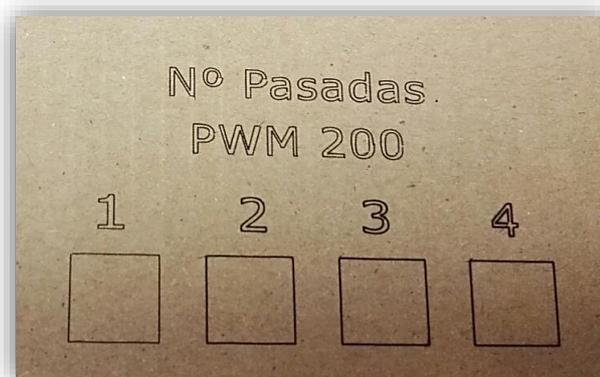


Figura 7.4. Prueba de número de pasadas.

3. Resolución de los ejes y separación entre líneas.

Un aspecto a tener en cuenta para la insulación de placas de circuitos impresos, es la distancia mínima entre pistas que es capaz de producir el láser. Para este caso, esta distancia coincidirá con el ancho mínimo de las pistas, puesto que el láser solo se mueve por el contorno de la imagen. También es importante conocer la resolución del eje horizontal y del vertical. En principio, debería ser el mismo puesto que tanto los motores como las correas son iguales.

Antes de aplicar el haz de luz sobre una placa, se ha creado un diseño con distintas separaciones entre líneas horizontales y verticales que serán dibujadas sobre una cartulina. Las cuales, están separadas entre ellas por valores que van desde 2 hasta 0,1 milímetros (colocándose el número correspondiente en la parte superior)

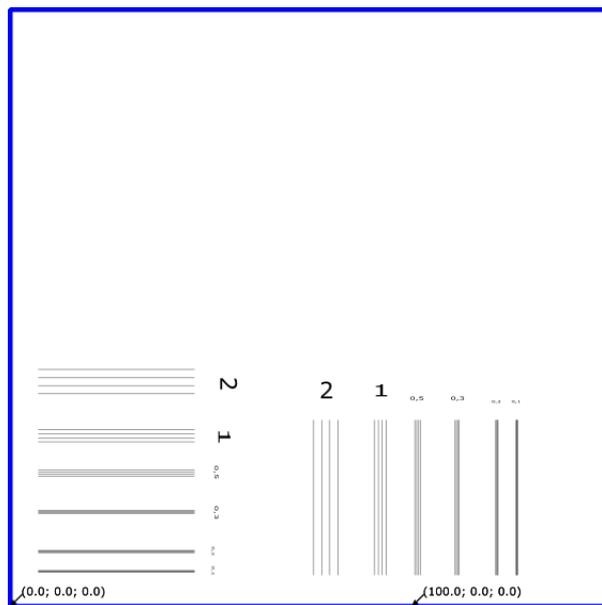


Figura 7.5. Prueba de resolución de los ejes y separación entre líneas.

Tras la impresión con el láser, se puede observar que las líneas verticales son rectas mientras que las realizadas por el eje x poseen algunas curvas. Por lo tanto, el eje y tiene una mayor resolución en comparación con el eje horizontal. Los problemas producidos se deben posiblemente a un mal ajuste mecánico.

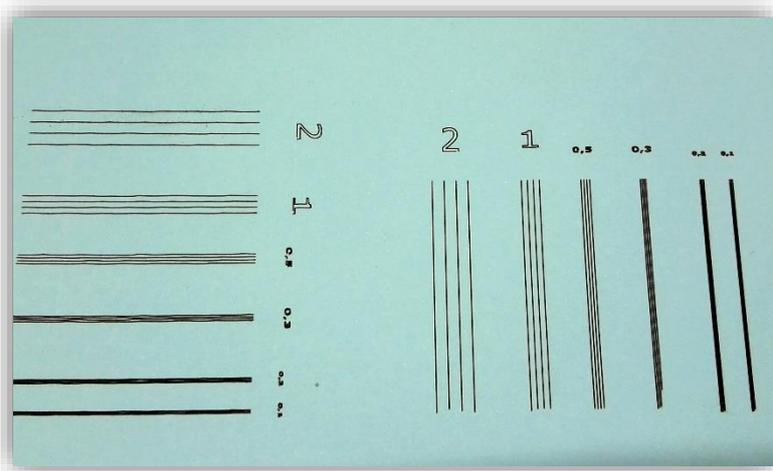


Figura 7.6. Resultado de la prueba de resolución de los ejes y separación entre líneas.

3.1. Separación entre líneas sobre cartón.

Con respecto al eje vertical, se ha repetido el diseño anterior para saber la separación mínima entre las líneas rectas.

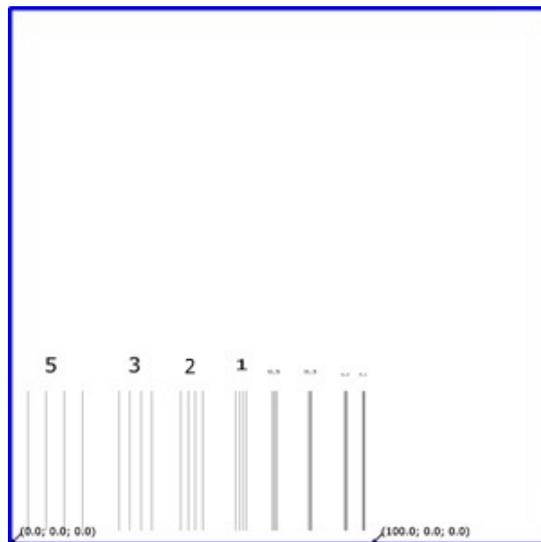


Figura 7.7. Prueba de separación entre líneas sobre cartón.

El dibujo se repitió sobre cartón sin la lente y con ella, modificando el valor de la intensidad y/o la distancia respecto a la base de la impresora. Sabiendo que, con la lente, la altura debe ser de 5 centímetros, se determinó la distancia mínima para distinguir todas las líneas. A continuación, se muestra una tabla con las variaciones realizadas y los resultados de las mismas.

Valor PWM	Distancia eje z (mm)	Separación mín. (mm)
Sin lente		
250	80	0,5
240	80	0,3
230	80	0,3
220	80	0,3
220	54	0,3
Con lente		
220	50	0,3
230	50	0,3
240	50	0,3

Tabla 7.1. Resultados de la separación mínima del eje y.

3.2. Separación entre líneas sobre placa fotosensible.

Una vez empleado el láser sobre diversas superficies, es el momento de aplicarlo a placas fotosensibles. Se ha seleccionado solo las separaciones desde 1 hasta 0,1 milímetros, ya que con las pruebas con el cartón se pudo ver que el diámetro del spot del láser es lo suficientemente pequeño como para hacerlo sin problema. Además, se han añadido otras formas geométricas, como se ve en la siguiente imagen.

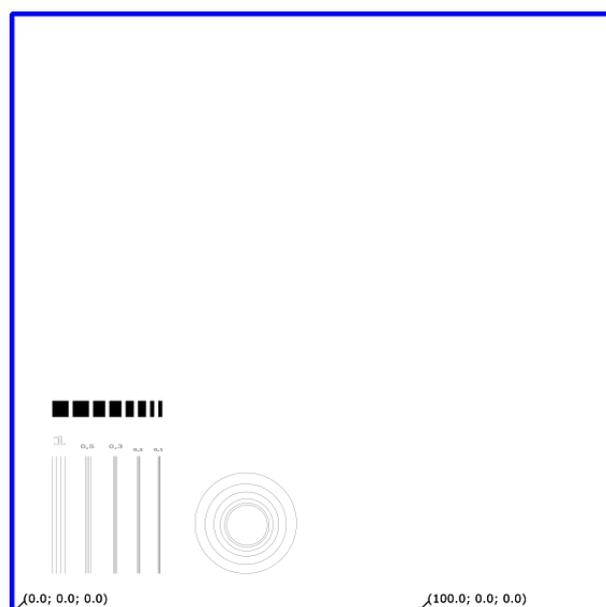


Figura 7.8. Diseño de separación entre líneas sobre placa fotosensible.

Para poder observar con claridad los resultados se ha revelado la placa sumergiéndola en una disolución de revelador de placas positivas durante un minuto aproximadamente. Cuando se aprecia el color característico del cobre en las zonas insoladas, se retira de la disolución.



Figura 7.9. Resultado. Separación entre líneas sobre placa fotosensible.

4. PCB de prueba.

Para comprobar si la intensidad del láser y el tiempo de insolación son suficientes para la creación de PCBs se ha diseñado un circuito sencillo que se denominará a partir de ahora como “PCB de prueba”.

Aunque las placas parezcan insoladas tras el revelado, se sabrá si el proceso ha tenido éxito tras el atacado, pues es cuando se elimina el cobre de manera que las pistas queden aisladas. El atacado consiste en introducir la placa revelada, en una disolución de ácido clorhídrico (HCL). El tiempo que tarda en retirarse el cobre depende del tipo de placa.

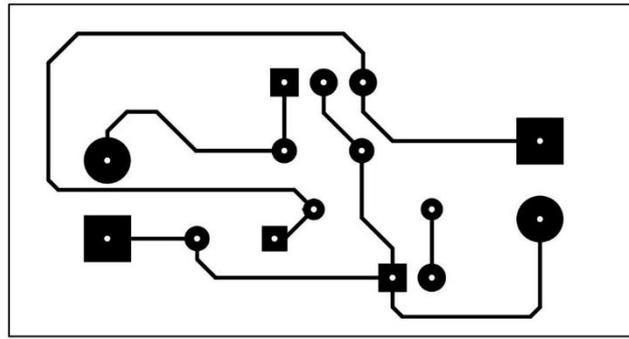


Figura 7.10. Fotolito de la PCB de prueba.

Con respecto a los componentes, se han empleado los siguientes:

Referencia	Valor	Tipo de componente	Footprint
C1	1uF	Condensador	Capacitors_THT:C_Disc_D4.7mm_W2.5mm_P5.00mm
C2	100nF	Condensador	Capacitors_THT:CP_Radial_Tantal_D4.5mm_P5.00mm
D1	-	LED	LEDs:LED_D3.0mm
J1	-	Conector	Connectors:bornier2
J2	-	Conector	Connectors:bornier2
R1	1kΩ	Resistencia	Resistors_THT:R_Axial_DIN0204_L3.6mm_D1.6mm_P7.62mm_Horizontal
U1	-	Regulador LM7805CT	TO_SOT_Packages_THT:TO-220_Vertical

Tabla 7.2. Componentes de la PCB de prueba.

El siguiente paso tras comprobar que no hay errores, es la creación de la placa. La distribución de los componentes y el trazado de las pistas se realizaron de acuerdo teniendo en cuenta las reglas de diseño. Para este circuito se han creado las pistas en la capa Bottom.

En definitiva, este circuito será la base de las pruebas en placas fotosensibles. Se modificarán valores como la intensidad, el número de pasadas o el tipo de placa, para definir la forma óptima de insolación con este sistema.

5. Resultados.

En este apartado se muestran los resultados obtenidos tras realizar el proceso completo de fabricación. En primer lugar, se realiza una prueba para comprobar que se aislarán las pistas sin la lente. Para ello, se han creado tres cuadrados concéntricos, primero con una única pasada y luego con dos (con un ciclo de trabajo de 100%).

5.1. Resultados de insolación.

Cabe destacar que el tiempo estimado de impresión que propone *Repetier-Host* no coincide con el tiempo real. En la siguiente tabla se muestran los tiempos de cada proceso.

Nº pasadas	Tiempo estimado insolado	Tiempo real insolado	Tiempo revelado	Tiempo atacado
1	2' 3 "	45 "	2 '	15 '
2	4' 6 "	1' 27 "	1 '	15 '

Tabla 7.3. Tiempos de fabricación. Cuadrados.

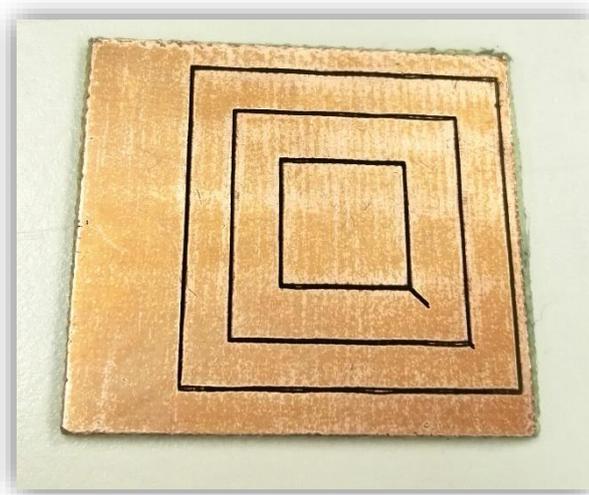


Figura 7.11. Resultado de los cuadrados tras dos pasadas.

Tras una pasada, las líneas horizontales provocan un mal resultado dejando las zonas de cobre no aisladas. Sin embargo, con una segunda repetición, se repasan todas las líneas y se produce el aislamiento esperado.

Además, en el proceso de revelado tan solo se ha aplicado revelador de placas positivas disuelto en agua. Por lo que, para este tipo de placas (con film protector blanco) no es necesario usar sosa cáustica (NaOH).

Las siguientes pruebas se ejecutarán con el diseño del circuito regulador de tensión, y hay que tener en cuenta tres variables que determinarán el resultado: el valor del ciclo de trabajo de la señal PWM, el ancho de la pista creado en *KiCad*, el diámetro que se ha configurado en *FlatCAM* y el tiempo de insolado. Este último parámetro está condicionado por el tipo de placa, para las pruebas se ha utilizado una con un tiempo en torno a los dos minutos.

Por consiguiente, se genera el circuito de prueba sobre una placa con resina fotosensible y con la lente, con los siguientes parámetros:

Valor del ciclo de trabajo de la señal PWM.	100%
Ancho de pista.	0,25 mm
Diámetro.	0,2 mm
Tiempo de insolación.	2 ' 24 "

Tabla 7.4. Parámetros para la PCB de prueba con lente.

Tras el revelado ya se aprecia que la zona central ha quedado más expuesta a la luz y al aplicar el ácido se verifica que el resultado no es el esperado. Esto es producido porque la lente refleja la luz del láser creando una huella que permanece sobre el medio de la placa.

Por lo tanto, la lente no proporciona una mejor resolución y produce problemas en la insolación. Igualmente, el ácido ha atacado completamente a las pistas, por lo que se debe escoger un diámetro del spot mayor a 0,2 mm.

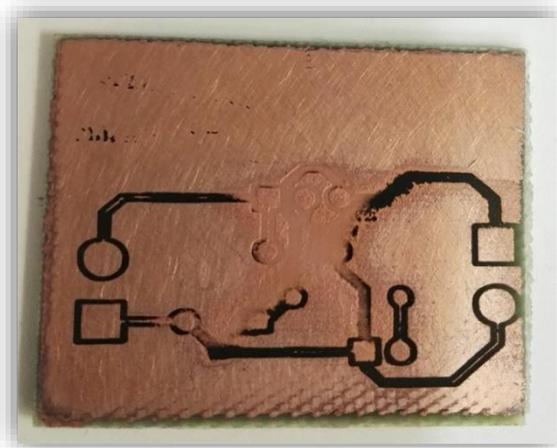


Figura 7.11. Resultado de la PCB de prueba con lente.

Teniendo en cuenta los errores anteriores, se ha insolado el mismo circuito, pero en este caso sin la lente. Para este caso, se ha modificado el diámetro para garantizar la creación de las pistas.

Nº pasadas	Tiempo estimado insolado	Tiempo real insolado	Tiempo revelado	Tiempo atacado
2	7 ' 34 "	2' 24"	4 ' 20 "	17 '

Tabla 7.5. Tiempos de fabricación. PCB de prueba con diámetro igual a 0,5 mm.

De esta manera, se pueden visualizar claramente el dibujo que forman las pistas. También, se ha comprobado la conectividad de las mismas y el aislamiento respecto al resto de material conductor. No obstante, las pistas tienen un ancho real de 0,75 mm, por lo que se puede afirmar que se ha escogido un diámetro mayor que el real.

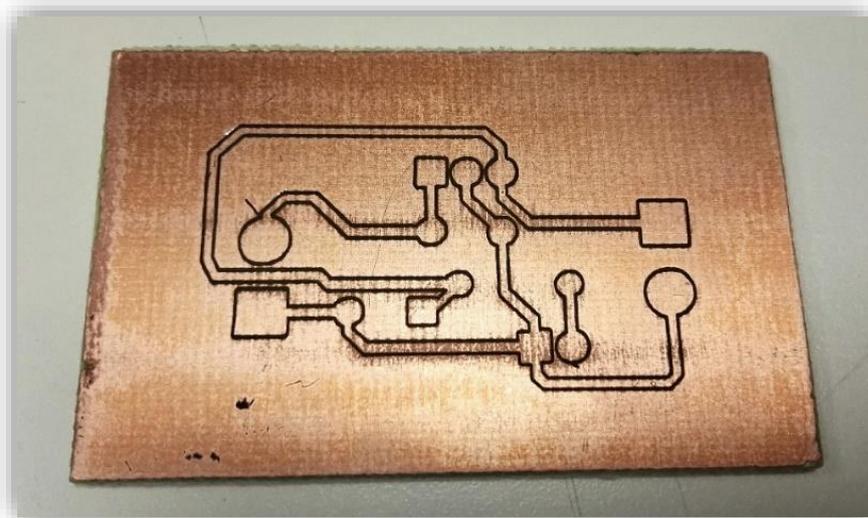


Figura 7.12. Resultado de la PCB de prueba sin lente. Diámetro 0,5 mm.

La última prueba se ejecuta para mostrar las diferencias que se producen al reducir el diámetro a 0,3 mm, manteniendo el valor del resto de los parámetros. Ahora, el ancho real de la pista se reduce a 0,5 mm aproximadamente y sigue existiendo aislamiento entre las diferentes pistas.

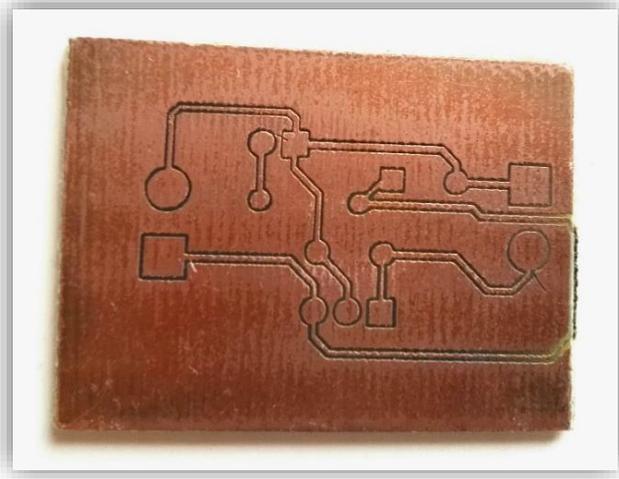


Figura 7.13. Resultado de la PCB de prueba. Diámetro 0,3 mm.

5.2. Resultados de serigrafía.

Durante la ejecución de las pruebas anteriores, se crearon tanto figuras geométricas como letras que el láser fue capaz de realizar. Por este motivo, es posible generar la serigrafía en la parte posterior de las placas de una sola cara.

El procedimiento es sencillo, tras importar el dibujo de las pistas en *Inkscape* se bloquea la capa en la que se encuentra el mismo y se crea otra. En ella, se podrá escribir cualquier texto o incluso importar la capa de serigrafía generado en *KiCad* y colocarla sobre el circuito. En la siguiente imagen se muestra un ejemplo del resultado.



Figura 7.13. Resultado de serigrafía con texto.

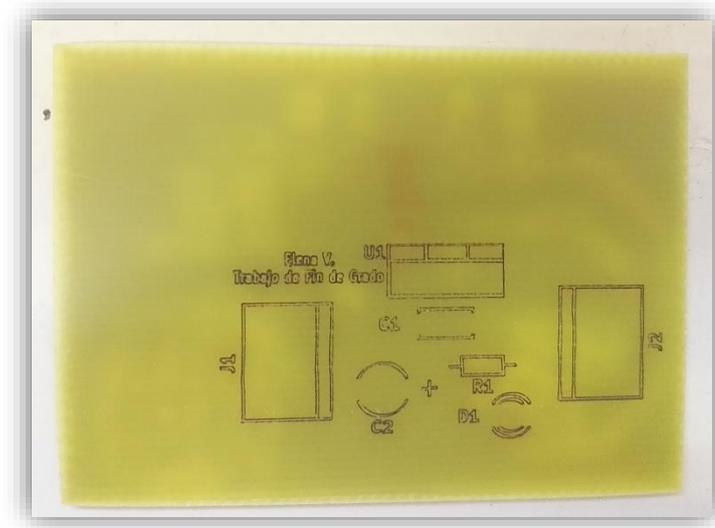


Figura 7.14. Resultado de serigrafía con las huellas de los componentes.

Capítulo 8: Presupuesto.

1. Introducción.

En el siguiente capítulo se detallarán los precios de los materiales utilizados para la implementación del proyecto, el coste relacionado con la mano de obra y, por último, la suma de los anteriores para determinar el presupuesto total del trabajo.

2. Coste de los materiales.

Descripción	Cantidad (ud)	Coste Unitario (€/ud)	Coste total (€)
Kit Prusa i3	1	498,90	498,90
Arduino Mega 2560	1	42,98	42,98
RAMPS 1.4.	1	59,50	59,50
Fuente de alimentación	1	24,95	24,95
Cables	1	12,00	12,00
Diodo láser, circuito de control y gafas protectoras	1	26,98	26,98
Placa fotosensible positiva	1	2,21	2,21
Placa perforada cintas	1	5,50	5,50
Resistencias	4	0,10	0,40
Led indicador	1	0,50	0,50
Regulador de tensión (7805)	1	0,39	0,39
Condensador electrolítico	1	0,25	0,25
Puerta NOT (74LS04)	1	0,64	0,64
MOSFET (IRF530)	1	0,93	0,93
Conector	4	0,35	1,40
Tornillos	10	0,20	2,00
Lente convergente	1	20,00	20,00
Filamento PLA	1	17,95	17,95
Revelador placa positiva	1	3,26	3,26
Atacador de PCB	1	4,40	4,40
Coste total de materiales			725,14

Tabla 9.1. Coste de los materiales.

3. Coste de la mano de obra.

Concepto	Cantidad (h)	Coste Unitario (€/h)	Coste total (€)
Tiempo de configuración	80	20	1600
Tiempo de diseño 3D	40	20	800
Tiempo realización de pruebas	140	20	2800
Tiempo de documentación	50	20	1000
Coste total de mano de obra			6200

Tabla 9.2. Coste de la mano de obra.

4. Presupuesto total del proyecto.

Concepto	Coste total (€)
Coste total materiales	725,14
Coste total de mano de obra	6200
Coste total del proyecto	6925,14

Tabla 9.3. Presupuesto total del proyecto.

Capítulo 9: Conclusiones y futuras mejoras.

1. Conclusiones.

En base a los resultados alcanzados se han obtenido las siguientes conclusiones:

- El sistema implementado para insolar placas fotosensibles de forma automática funciona, aunque se han detectado algunos aspectos sujetos a mejora.
- La lente utilizada aparentemente no presenta ninguna ventaja para este tipo de aplicación, puesto que no reduce el diámetro del spot. Este problema puede ser causado por diferentes motivos: que el láser no se encuentre correctamente colimado, que el haz del mismo no coincida con el centro de la lente o que la placa no se encuentre exactamente en el foco.
- Las líneas generadas por el movimiento del eje horizontal no son totalmente rectas a causa del ruido mecánico. Por lo que, se colocará el circuito en el área de impresión, de tal manera que la mayoría de las líneas sean verticales.
- Para realizar la insolación de forma correcta, el ciclo de trabajo de la señal PWM es el máximo (100%) y el número de pasadas para asegurar el aislamiento de las pistas debe ser de al menos 2.

Actualmente, este sistema no resulta ser una alternativa cómoda con respecto al método tradicional. Sin embargo, tras solventar los problemas mencionados, se podrá obtener una resolución mayor, que la lograda con un fotolito, en un tiempo de insolado relativamente corto.

2. Futuras mejoras.

El presente proyecto es un primer prototipo con vistas a futuras mejoras.

- Colimar correctamente el diodo láser y rediseñar el soporte para la lente para ajustarla milimétricamente en las tres dimensiones.
- El problema del eje horizontal para dibujar líneas rectas, podría solucionarse cambiando el tipo de correa que produce el movimiento o modificando la sujeción del cabezal de la impresora.
- El pin que controla el láser podría asociarse a uno que genere una señal PWM de 5V y así reducir la electrónica de la impresora.

3. Opinión personal.

Desde mi punto de vista, este proyecto me ha permitido conocer el desarrollo actual las impresoras 3D y las múltiples aplicaciones que tienen. También, ha sido una oportunidad para aplicar algunos de los conocimientos adquiridos en el grado, como la creación de figuras en 3D, el diseño de circuitos electrónicos, etc. En general, he podido llevar a cabo el objetivo marcado inicialmente, a pesar de los problemas presentados.

Bibliografía

- [1] https://campusvirtual.ull.es/1617/pluginfile.php/318846/mod_resource/content/5/temario/tema3/DYTECI_Tema3a.pdf
- [2] <http://reprap.org/wiki/RepRap/es>
- [3] <https://www.zonamaker.com/impresion-3d/crea-impresora/3-electronica-crea-imp>
- [4] <https://www.arduino.cc/en/Main/arduinoBoardMega>
- [5] <https://electronilab.co/tienda/shield-ramps-1-4-para-impresora-3d-reprap-prusa-mendel/>
- [6] <https://www.arduino.cc/en/Guide/Environment>
- [7] <http://flatcam.org/manual/introduction.html>
- [8] <https://inkscape.org/en/about/overview/>
- [9] https://www.freecadweb.org/?lang=es_ES
- [10] <https://es.wikibooks.org/wiki/F%C3%ADsica/%C3%93ptica/Lentes>
- [11] <http://www.eis.uva.es/~biopolimeros/alberto/pla.htm>
- [12] <https://playground.arduino.cc/ArduinoNotebookTraduccion/Appendix3>
- [13] https://es.wikipedia.org/wiki/Transistor_de_efecto_de_campo_metal-%C3%B3xido-semiconductor#Funcionamiento
- [14] <http://flatcam.org/manual/objectreference.html>
- [15] https://jtechphotonics.com/?page_id=1980
- [16] <http://www.jycircuitboard.com/news/8-layer-pcb-stack-up-guidelines-115.html>
- [17] <https://store.arduino.cc/arduino-mega-2560-rev3>
- [18] <https://mind42.com/public/dcba44fc-7240-4213-ad49-d10888b4359f>
- [19] <https://polaridad.es/que-es-g-code/>



ESCUELA SUPERIOR DE INGENIERÍA Y TECNOLOGÍA

ANEXOS

TRABAJO DE FIN DE GRADO

Grado en Ingeniería Electrónica Industrial y Automática

TÍTULO:

**DISEÑO E IMPLEMENTACIÓN DE UNA IMPRESORA
PARA REALIZACIÓN DE PLACAS DE CIRCUITOS
IMPRESOS (PCB) BASADA EN GRABADOR LÁSER.**

Alumno/a: Elena Velázquez Navarro.

Tutores: Manuel Rodríguez Valido
David Hernández Expósito.

Julio 2017

ÍNDICE ANEXOS

Anexo I: G-code.....	92
1. Descripción del G-code.	93
1.2. Códigos principales.	93
1.3. Otros comandos.....	93
1.4. Fichero G-code generado por <i>Laser Tool</i>	95
Anexo II: RAMPS 1.4.....	97
1. Conexiones con Arduino Mega 2560.	98
2. Esquemático RAMPS 1.4.....	98
Anexo III: Planos.	99
Anexo IV: Circuito adaptador de señal. Esquemático..	102
Anexo V: PCB de prueba. Esquemático, PCB, fotolito y visión 3D.	104
PCB.....	106
Fotolito.....	106
Visión 3D de la cara TOP.....	107
Visión 3D de la cara BOTTOM.....	107
Anexo VI: Datasheet diodo láser modelo HL40023MG.....	108

Anexo I: G-code.

1. Descripción del G-code.

El G-code es un lenguaje de descripción de operaciones para máquinas CNC que puede ser usado también como lenguaje de programación para controlar dispositivos. Se almacena en formato texto, es decir, se puede leer y modificar con un editor de texto plano [19]. Sin embargo, lo más habitual es generarlo y visualizarlo desde una aplicación de modelado y/o fabricación 3D, como por ejemplo *Repetier-Host*.

Los códigos G son conocidos también como códigos preparatorios y se caracterizan por comenzar por la letra “G” y van acompañados por dos dígitos que van desde el 00 hasta el 99 y cada uno de ellos tiene una función determinada.

En este trabajo fue necesario conocer los códigos principales para entender el fichero que genera la herramienta *Laser Tool* de *Inkscape* y para enviar códigos directamente a la máquina desde *Repetier*.

1.2. Códigos principales.

Código	Descripción
G00	Interpolación lineal rápida.
G01	Interpolación lineal a la velocidad programada en el registro F.
G21	Comienzo de uso de unidades métricas.
G28	Volver al home de la máquina.
G71	Unidad de datos expresados en milímetros.
G90	Desplazamiento en modo absoluto.
G98	Velocidad de alimentación (unidades/min).

Tabla I.1. Códigos G principales.

1.3. Otros comandos.

Generalmente, los códigos anteriores van acompañados de otras letras para definir parámetros que se relacionan con el movimiento. En la siguiente tabla se muestran los utilizados en este proyecto y su descripción.

Comando	Descripción
X	Comando para el eje x.
Y	Comando para el eje y.
Z	Comando para el eje z.
R	Radio desde el centro especificado.
F	Tasa de alimentación.
S	Velocidad de giro.
T	Número de herramienta.
M	Funciones Misceláneas

Tabla I.2. Comandos.

Las funciones Misceláneas destacan por definir acciones concretas de la impresora 3D y conforman un propio código, por lo que no van seguidas de un código G. En este caso, la letra M va acompañada por tres dígitos que van desde el 000 hasta el 999. Sin embargo, solo algunos de ellos son utilizados en la impresora láser.

Código	Descripción
M17	Encender motores.
M18	Apagar motores.
M106	Encender ventilador.
M107	Apagar ventilador.
M114	Devuelve la posición actual del cabezal.
M120	Guarda la posición del cabezal.
M220	Modificar la velocidad de impresión.
M502	Cargar los valores por defecto (codificados en el firmware) en la memoria.

Tabla I.3. Códigos M principales.

1.4. Fichero G-code generado por *Laser Tool*.

Como se ha mencionado en capítulos anteriores, la extensión *Laser Tool* de *Inkscape* es el encargado de generar estos comandos para la trayectoria seleccionada. Independientemente del tipo de imagen que se procese, el archivo GCODE siempre tiene una estructura determinada como se muestra a continuación.

En primer lugar, se garantiza que el láser estará apagado antes de comenzar el proceso. Se establece el valor de la señal PWM en 0.

```
M107 S0
```

Se configura el desplazamiento en modo absoluto.

```
G90
```

Se expresan todas las unidades en milímetros.

```
G21
```

En la siguiente línea se establece la velocidad de trayectoria del láser cuando permanece apagado. En este caso es de 3000 mm/min.

```
G1 F3000
```

Se mueve la impresora a las coordenadas señaladas en milímetros, que será el primer punto dónde comenzará la impresión.

```
G1 X11.1965 Y44.4341
```

G4 permite parar el proceso durante un tiempo determinado y en este caso se corresponde con el tiempo de retraso en comenzar la impresión, que es configurado en *Inkscape*. Para *Repetier*, el tiempo se define con la letra P y el valor en milisegundos.

```
G4 P0
```

El siguiente paso es encender el láser, siendo 230 el valor de la señal PWM, es decir, el ciclo de trabajo será del 90%. Por lo tanto, en este ejemplo la señal permanecerá más tiempo en nivel alto que en nivel bajo.

```
M106 S230
```

Se vuelve a definir el tiempo de retraso en 0 milisegundos.

```
G4 P0
```

Luego, se define la velocidad del láser cuando está encendido. En este caso será de 100 mm/min.

```
G1 F100.000000
```

Una vez se enciende el láser, se envían las siguientes coordenadas de una trayectoria continua.

```
G1 X11.1965 Y44.4353
```

```
G1 X11.1965 Y44.4365
```

```
G1 X11.1997 Y44.5017
```

```
G1 X11.1999 Y44.5041
```

```
G1 X11.2002 Y44.5065
```

```
G1 X11.2098 Y44.5711
```

Cuando sea necesario apagar el láser debido a que se produce una discontinuidad en el trayecto, se repite el proceso desde el principio.

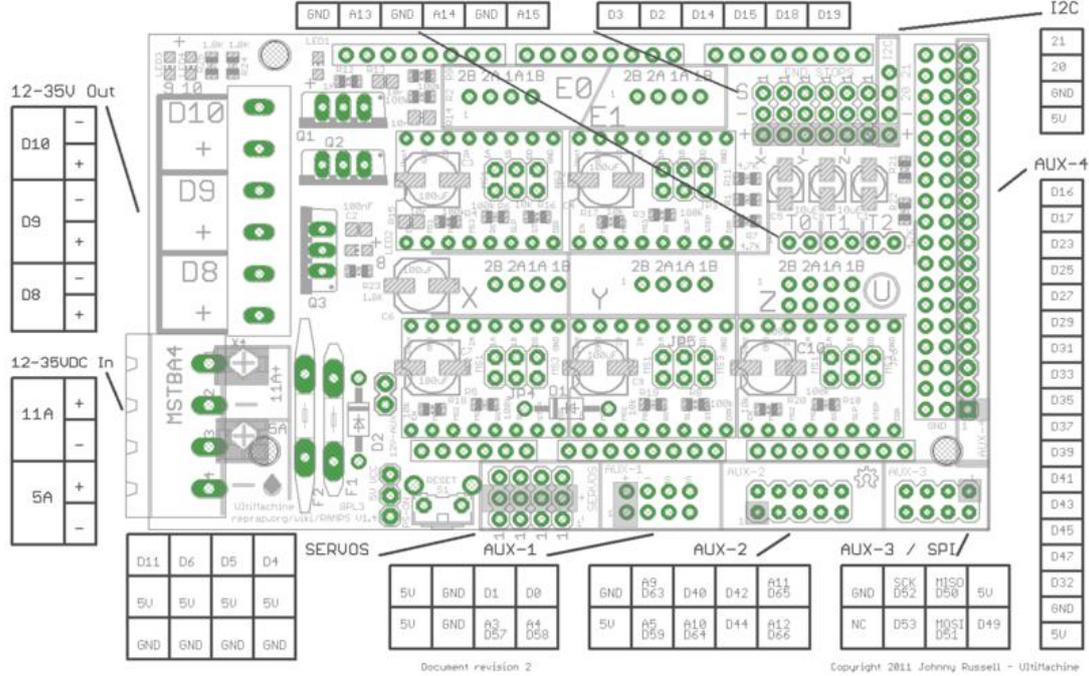
Anexo II: RAMPS 1.4.

1. Conexiones con Arduino Mega 2560.

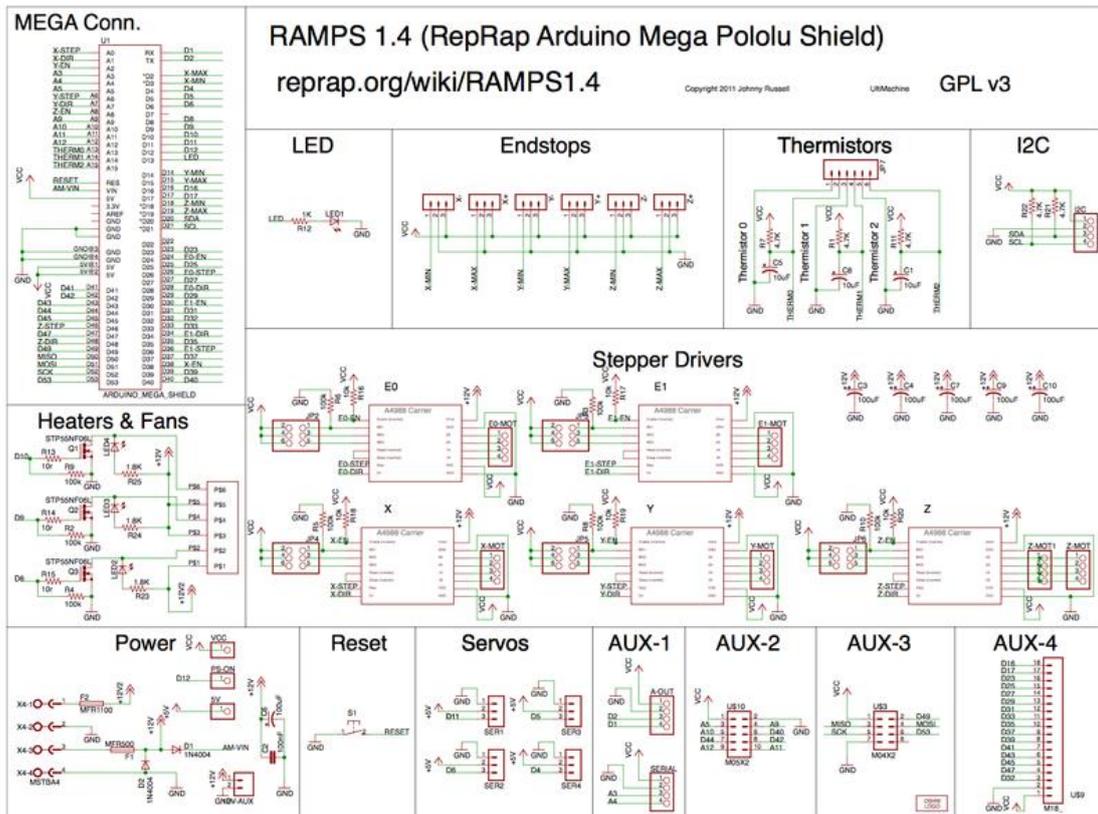
RAMPS 1.4 (RepRap Arduino MEGA Pololu Shield)
reprap.org/wiki/RAMPS1.4

GPL v3

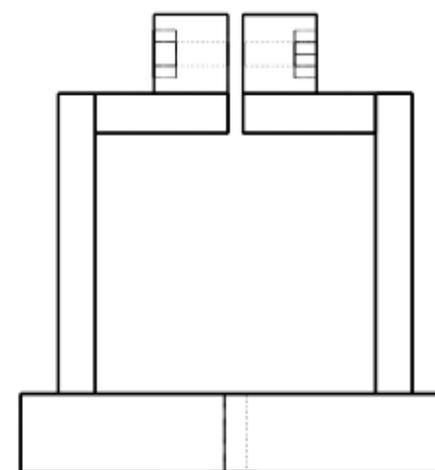
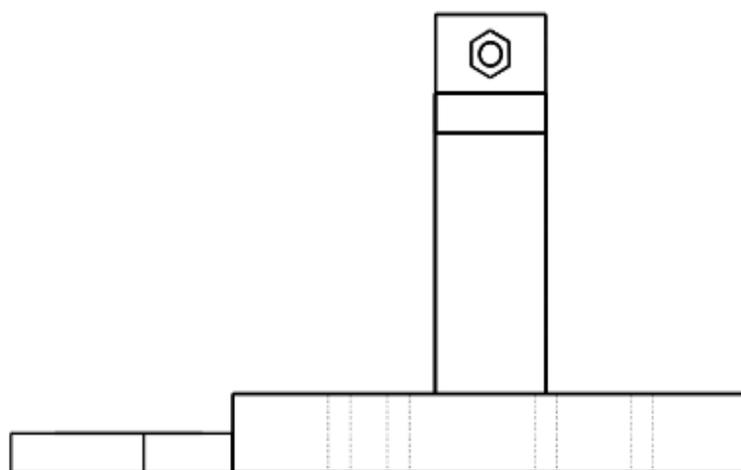
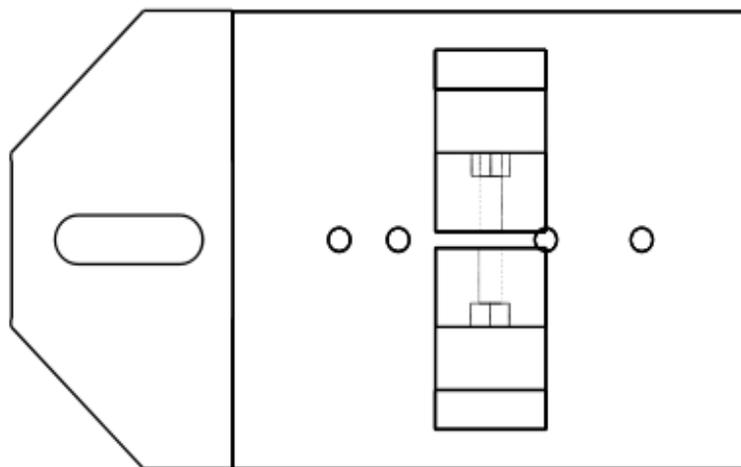
Reversing input power, and inserting stepper drivers incorrectly will destroy electronics.



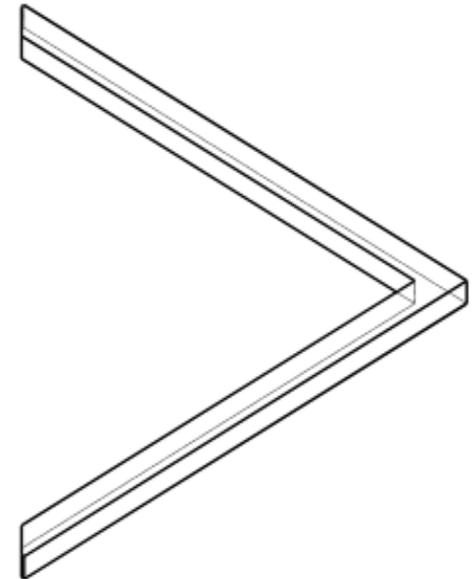
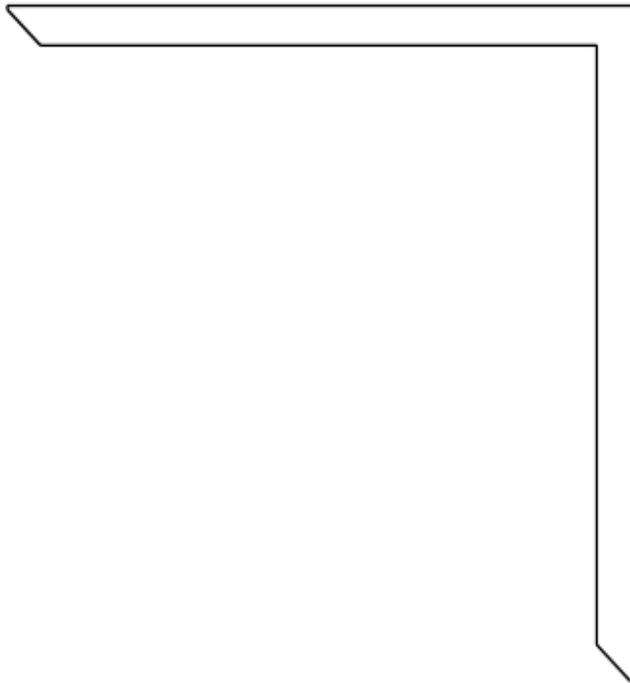
2. Esquemático RAMPS 1.



Anexo III: Planos.



Diseño e implementación de una impresora para la realización de PCBs basada en grabador láser.			
Fecha	Autor		TFG Grado en Ingeniería Electrónica, Industrial y Automática
28/06/2017	Elena Velázquez Navarro		
Modificación			
Id. s. Normas	UNE - EN - DIN		
Escala: 1:1	Pieza de soporte del láser.		Nº de Plano: 001



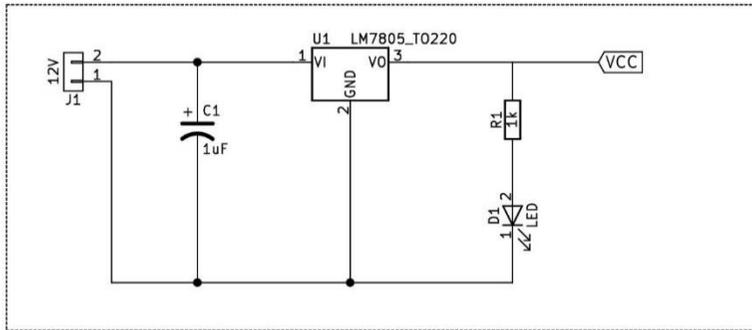
Diseño e implementación de una impresora para la realización de PCBs basada en grabador láser.

Fecha	Autor		TFG Grado en Ingeniería Electrónica, Industrial y Automática
28/06/2017	Elena Velázquez Navarro		
Modificación			
Id. s. Normas	UNE - EN - DIN		
Escala: 1:2	Sistema de referencia.		Nº de Plano: 002

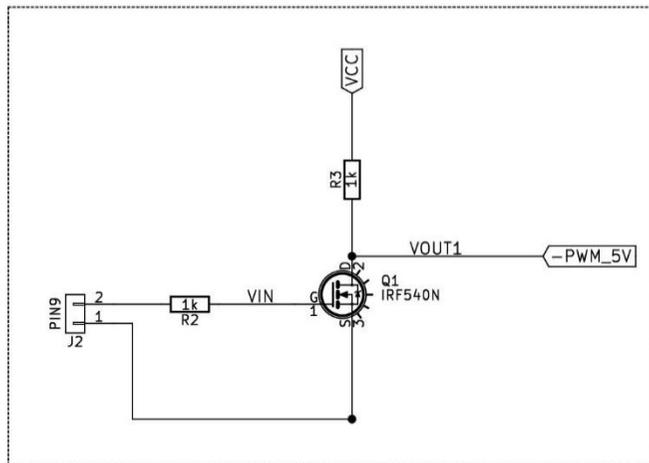
Anexo IV:

Circuito adaptador de señal. Esquemático.

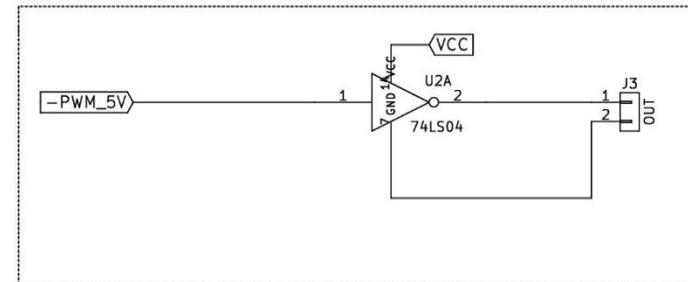
Alimentación 5V



Reductor de amplitud

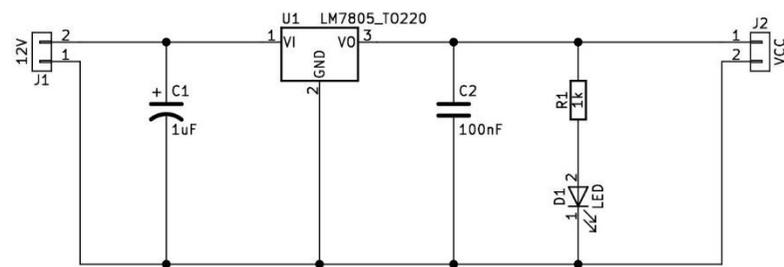


Aislador de circuitos



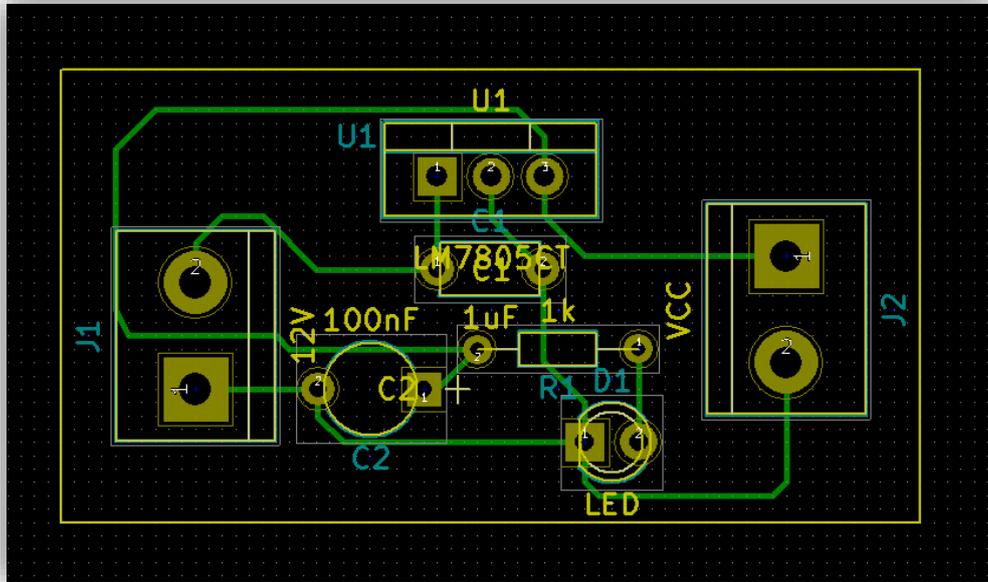
Diseño de una impresora para la realización de PCBs basada en grabador láser.			
	Fecha	Autor	 Universidad de La Laguna
Dibujado	02/07/2017	Elena Velázquez	
Comprobado	02/07/2017		
Id. s. normas	UNE-EN-DIN		Nº Plano
Escala S/E	Círculo adaptador de señal.		003

Anexo V:
PCB de prueba.
Esquemático, PCB, fotolito y
visión 3D.

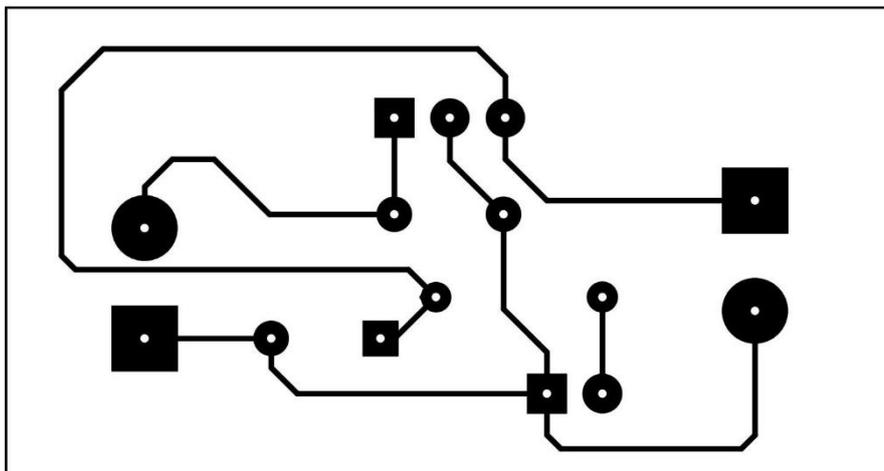


Diseño de una impresora para la realización de PCBs basada en grabador láser.			
	Fecha	Autor	 Universidad de La Laguna
Dibujado	02/07/2017	Elena Velázquez	
Comprobado	02/07/2017		
Id. s. normas	UNE-EN-DIN		Nº Plano
Escala S/E	Circuito de prueba.		003

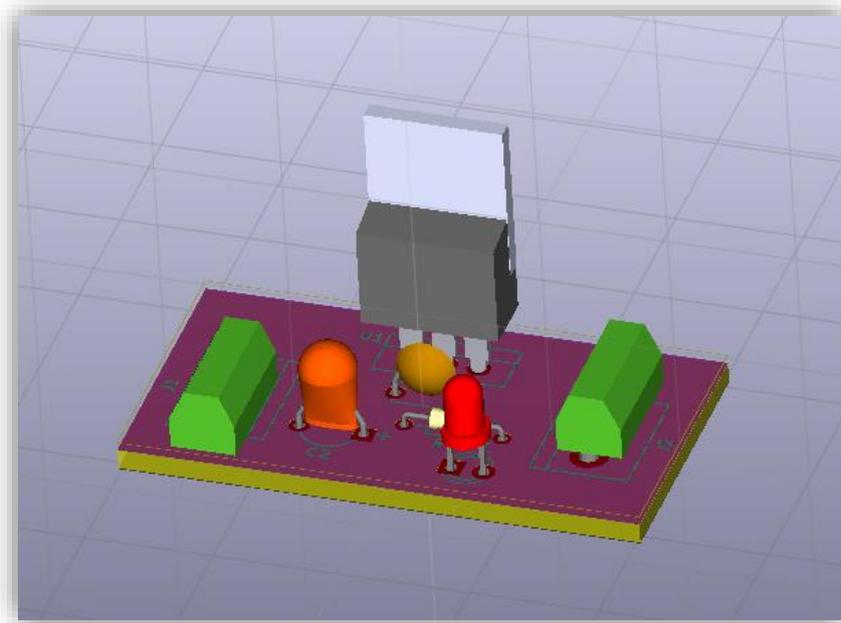
PCB.



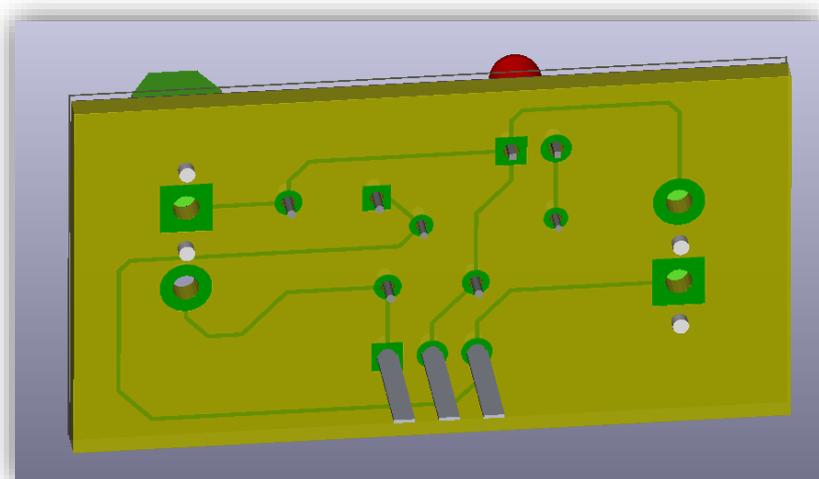
Fotolito.



Visión 3D de la cara TOP.



Visión 3D de la cara BOTTOM.



Anexo VI:

Datasheet diodo laser modelo HL40023MG.

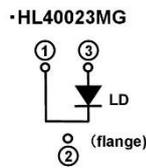


HL40023MG - 405nm band / 500mW -
GaN Violet High Power Laser Diode

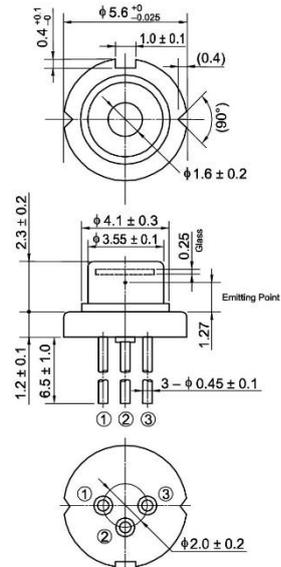
Preliminary
 Rev.2
 24. Nov. 2010

- Applications
 - Direct Imaging for PCB
 - Industry

■ Internal circuit



■ Outline



- Features
 - Optical output powr: $P_o=400mW(CW)$
 - Violet lasing $\lambda_p=398\sim 410nm$
 - Low operating current: $I_{op}=390mA$ Typ.
 - Low operating voltage: $V_{op}=5.5V$ Max.
 - Small package: $\phi 5.6mm$
 - Multi transverse mode oscillation

■ Absolute Maximum Ratings($T_c=25^\circ C$)

Item	Symbol	Ratings	Unit
Optical output power	P_o	500	mW
LD Reverse Voltage	$V_{R(LD)}$	2	V
Operating Temperature	T_{opr}	0 ~ +30	$^\circ C$
Storage Temperature	T_{stg}	-40 ~ +85	$^\circ C$

■ Optical and Electrical Characteristics($T_c=25^\circ C$)

Item	Symbol	Min.	Typ.	Max.	Unit	Test condition
Threshold current	I_{th}	-	-	160	mA	-
Operating current	I_{op}	-	390	420	mA	$P_o=400mW$
Operating voltage	V_{op}	-	-	5.5	V	$P_o=400mW$
Lasing Wavelength	λ_p	398	-	410	nm	$P_o=400mW$
Beam divergence Parallel to the junction	$\theta_{//}$	5	-	25	$^\circ$	$P_o=400mW$, Full angle $1/e^2$
Beam divergence Perpendicular to the junction	θ_{\perp}	30	-	60	$^\circ$	$P_o=400mW$, Full angle $1/e^2$

Note : This type is underdevelopment. Therefore, this data sheet may be changed without any notice.

