



*“Responsabilidad con pensamiento positivo”*

**UNIVERSIDAD TECNOLÓGICA ISRAEL**

**TRABAJO DE TITULACIÓN EN OPCIÓN AL GRADO DE:  
INGENIERO EN ELECTRÓNICA DIGITAL Y  
TELECOMUNICACIONES**

**TEMA:**

**PROTOTIPO MULTIPROPÓSITO FRESADORA CNC E IMPRESORA  
VOLUMÉTRICA.**

**AUTOR:**

**FERNANDO ARTURO CAJO JIMÉNEZ**

**TUTOR:**

**ING. FIDEL DAVID PARRA BALZA, PhD**

**QUITO, ECUADOR**

**2019**

**UNIVERSIDAD TECNOLÓGICA ISRAEL**  
**APROBACIÓN DEL TUTOR**

En mi calidad de tutor del trabajo de titulación certifico:

Que el trabajo de titulación **“PROTOTIPO MULTIPROPÓSITO FRESADORA CNC E IMPRESORA VOLUMÉTRICA”**, presentado por el **Sr. Fernando Arturo Cajo Jiménez**, estudiante de la carrera de Electrónica Digital y Telecomunicaciones, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación.

Quito D.M. Septiembre del 2018

TUTOR

.....  
Ing. Fidel Parra, Mg

**UNIVERSIDAD TECNOLÓGICA ISRAEL**  
**APROBACIÓN DEL TUTOR**

En mi calidad de tutor del componente práctico certifico:

Que el trabajo de titulación **“PROTOTIPO MULTIPROPÓSITO FRESADORA CNC E IMPRESORA VOLUMÉTRICA”**, presentado por el **Sr. Fernando Arturo Cajo Jiménez**, estudiante de la carrera de Electrónica Digital y Telecomunicaciones, reúne los requisitos y méritos suficientes para ser sometido a la evaluación del Tribunal de Grado, que se designe, para su correspondiente estudio y calificación.

Quito D.M. Septiembre del 2018

TUTOR

.....  
Ing. Fidel Parra, Mg

## **AGRADECIMIENTO**

En el presente trabajo de titulación quisiera agradecer el apoyo incondicional que me brindaron mis padres: Luis Cajo y Esthela Jiménez, además el respaldo brindado por mis amigos, familiares y antiguos educadores que siempre estuvieron dispuestos a tomarse una parte de su tiempo para poder atender mis inquietudes y necesidades académicas. También quiero agradecer el tiempo y la paciencia dedicados por parte de quien fue mi tutor en este proceso de titulación.

## **DEDICATORIA**

Dedico este proyecto a mis padres Luis Eudoro Cajo Tixi y Esthela Margarita Jiménez Arrobo, quienes me apoyaron económica, emocional e incondicionalmente en todo este proceso de titulación y siempre estuvieron pendientes de todo aquello que me hizo falta para alcanzar esta meta. Para ellos es esta dedicatoria, pues a ellos es a quienes les debo mis logros.

## TABLA DE CONTENIDO

PORTADA .....	i
CERTIFICACIÓN DEL TUTOR.....	ii
CERTIFICACIÓN DEL TUTOR.....	iii
AGRADECIMIENTO .....	iv
DEDICATORIA .....	v
TABLA DE CONTENIDO .....	vi
LISTA DE FIGURAS .....	x
LISTA DE TABLAS .....	xiv
LISTA DE ECUACIONES .....	xv
RESUMEN .....	xvi
ABSTRACT .....	xvii
INTRODUCCIÓN.....	1
Antecedentes de la situación objeto de estudio .....	1
Planteamiento del problema .....	3
Justificación .....	3
Objetivo General.....	4
Objetivo Específico .....	4
Alcance .....	4
Descripción de los capítulos .....	5
CAPÍTULO 1 .....	7
FUNDAMENTACIÓN TEÓRICA .....	7
1.1. Geeetech Prusa I3 pro B .....	8
1.1.1. Tarjeta de control GT2560 .....	10
1.1.2. Firmware Marlin.....	11
1.1.3. Código G .....	12

1.1.4.	Diseño asistido por computadora (CAD) .....	13
1.1.5.	Fresadora CNC .....	14
1.2.	Circuito integrado Atmega 328.....	15
1.2.1.	Especificaciones técnicas del Atmega 328.....	15
1.2.2.	Descripción de pines (Atmega 328) .....	16
1.2.3.	Equivalencia de pines de Atmega 328 para Arduino. ....	17
1.2.4.	Conexiones del Atmega 328.....	18
1.2.5.	Migración de Arduino a un microcontrolador Atmega 328 .....	20
1.3.	Regulador de voltaje LM317 .....	21
1.3.1.	Regulador de voltaje LM317 configuración para salida de hasta 4 A.....	22
1.3.2.	Capacitores para fuente de alimentación .....	22
1.4.	Transistor TIP122 .....	24
1.5.	Servomotor MG995 .....	25
1.6.	Herramienta giratoria Dremel 3000 .....	27
1.6.1.	Dremel eje flexible 225 .....	28
1.7.	Software Arduino.....	28
1.8.	Processing .....	29
1.9.	MIT App Inventor 2.....	30
1.10.	Serial Bluetooth Terminal.....	30
1.11.	Solidworks .....	31
1.12.	Inkscape .....	32
1.13.	Ultimaker Cura .....	32
1.14.	Proteus Design Suite 8 .....	33
CAPÍTULO 2 .....		35
MARCO METODOLÓGICO .....		35
2.1.	Tipo y diseño de investigación .....	35
2.2.	Técnicas e instrumentos de recolección de datos. ....	36

2.3. Metodología seleccionada.....	37
CAPÍTULO 3 .....	40
PROPUESTA .....	40
3.1. Aspectos técnicos requeridos para la fresadora CNC .....	40
3.2. Ventajas de un prototipo multipropósito .....	42
3.3. Estudio técnico para el diseño de hardware .....	42
3.3.1. Atmega 328 .....	42
3.3.2. Regulación de voltaje para el microcontrolador .....	43
3.3.3. Servomotor MG995 .....	44
3.3.4. Regulación de voltaje para el servomotor .....	44
3.3.5. Control de relés con el transistor TIP122 .....	45
3.4. Análisis de costos.....	46
3.5. Módulo que alojará el sistema .....	49
3.6. Conclusión .....	50
CAPÍTULO 4 .....	51
IMPLEMENTACIÓN .....	51
4.1. Desarrollo.....	51
4.1.1. Diseño de elementos fijos y móviles .....	51
4.1.2. Diseño de tarjeta electrónica .....	56
4.1.3. Diseño de software .....	63
4.2. Implementación .....	72
4.3. Pruebas de funcionamiento .....	76
4.4. Análisis de resultados .....	85
4.4.1. Modificación de la infraestructura.....	85
4.4.2. Detalle técnico .....	87
4.4.3. Reinterpretación del código G.....	88
CONCLUSIONES.....	91



RECOMENDACIONES .....	93
BIBLIOGRAFÍA .....	94
ANEXOS .....	96
ANEXO 1. Cronograma de actividades .....	97
ANEXO 2. Montaje de elementos diseñados en Solidworks .....	98
ANEXO 3. Impresora 3D antes y después de la colocación de acoples .....	103
ANEXO 4. Diagramas electrónicos y eléctricos .....	104
ANEXO 5. Programación en Arduino.....	106
ANEXO 6. Programación en Processing.....	111
ANEXO 7. Manual técnico y de usuario .....	139

## LISTA DE FIGURAS

Figura. 1.1. Impresora 3D Geeetech I3 pro B. ....	9
Figura. 1.2. Tarjeta de control GT2560. ....	11
Figura. 1.3. Código G ejecutado por Marlin. ....	12
Figura. 1.4. Código G. ....	13
Figura. 1.5. Fresadora CNC. ....	14
Figura. 1.6 Distribución de pines para el Atmega 328. ....	16
Figura. 1.7 Equivalencia de pines de Atmega 328 para Arduino. ....	18
Figura. 1.8 Conexiones para el Atmega 328. ....	19
Figura. 1.9 Esquema para grabar secuencia de inicio de Arduino. ....	20
Figura. 1.10 Regulador de voltaje Lm317. ....	21
Figura. 1.11. Regulador LM317 configuración para salida de hasta 4 amperios. ....	22
Figura. 1.12. Capacitor de desacoplo y bulk. ....	23
Figura. 1.13. TIP122, distribución de pines ....	24
Figura. 1.14. Transistor como interruptor. ....	24
Figura. 1.15. Servomotor MG995. ....	26
Figura. 1.16. Distribución de pines y longitud de pulso. ....	26
Figura. 1.17. DREMEL 300. ....	27
Figura. 1.18. DREMEL 300 datos técnicos. ....	27
Figura. 1.19. Dremel eje flexible 225. ....	28
Figura. 1.20. IDE Arduino. ....	29
Figura. 1.21. Interfaz de Processing. ....	29
Figura. 1.22. Interfaz del App Inventor 2. ....	30
Figura. 1.23. Interfaz del Serial Bluetooth Terminal. ....	31
Figura. 1.24. Interfaz de trabajo en Solidworks ....	31
Figura. 1.25. Interfaz de trabajo en Inkscape. ....	32
Figura. 1.26. Interfaz de Ultimaker Cura. ....	33
Figura. 1.27. Proteus Design Suite 8. ....	34
Figura. 3.1. Regulación de voltaje para el microcontrolador ....	43
Figura. 3.2. Regulación de voltaje para el servomotor. ....	44
Figura. 3.3. Control de relés con el transistor TIP122. ....	45
Figura. 3.4. Diseño tentativo del módulo ....	49

Figura. 4.1. Dimensionamiento de figuras. ....	52
Figura. 4.2. Cabezal de fresadora (lado izquierdo).....	52
Figura. 4.3. Cabezal de fresadora (lado derecho). ....	53
Figura. 4.4. Acople para borneras.....	53
Figura. 4.5. Nueva plataforma de impresión. ....	53
Figura. 4.6. Plataforma de (corte/grabado).....	54
Figura. 4.7. Acople para el montaje de plataforma (corte/grabado).....	54
Figura. 4.8. Plataforma para servomotor. ....	54
Figura. 4.9. Perilla para Dremel. ....	55
Figura. 4.10. Soporte superior para Dremel. ....	55
Figura. 4.11. Soporte inferior para Dremel. ....	55
Figura. 4.12. Impresión de piezas de acople. ....	56
Figura. 4.13. Regulación de fuente para el Atmega 328. ....	57
Figura. 4.14. Regulación de fuente para el Servomotor MG995.....	57
Figura. 4.15. Circuito de control con transistores TIP122.....	58
Figura. 4.16. Esquema de control utilizando al Atmega 328. ....	59
Figura. 4.17. Diseño de la PCB. ....	59
Figura. 4.18. Diseño para ser impreso. ....	60
Figura. 4.19. Esquema para ser impreso en la parte superior de la placa.....	60
Figura. 4.20. Impresión del circuito en papel termo transferible. ....	61
Figura. 4.21. Aplicación de calor para transferir diseño de placa. ....	61
Figura. 4.22. Quemado de placa usando Cloruro Férrico.....	61
Figura. 4.23. Placas listas para ser perforadas. ....	62
Figura. 4.24. Montaje de componentes y finalización de fabricación.....	62
Figura. 4.25. Diagrama eléctrico. ....	63
Figura. 4.26. Subiendo archivo compilado al Atmega 328. ....	64
Figura. 4.27. Prueba de funcionamiento del programa en Arduino. ....	64
Figura. 4.28. Prueba de funcionamiento del programa en Serial Bluetooth.....	65
Figura. 4.29. Interfaz de programación en Processing. ....	66
Figura. 4.30. Interfaz para el control de la CNC (cortadora / grabadora).. ....	66
Figura. 4.31. Comunicación Bluetooth.....	67
Figura. 4.32. Botones de avance.....	67
Figura. 4.33. Control de velocidad. ....	68
Figura. 4.34. Intercambio de pantallas. ....	68

Figura. 4.35. Programación de Canvas.....	69
Figura. 4.36. Programación de Canvas.....	69
Figura. 4.37. Programación de Canvas.....	70
Figura. 4.38. Envío de comandos G .....	70
Figura. 4.39. Control (Dremel / Aspiradora).....	71
Figura. 4.40. Interfaz principal del control de desplazamiento direccional. ....	71
Figura. 4.41. Interfaz secundaria del control de desplazamiento direccional.....	72
Figura. 4.42. Montaje del Dremel y servomotor en el módulo. ....	72
Figura. 4.43. Montaje del circuito electrónico y eléctrico en el módulo.....	73
Figura. 4.44. Acoples para la CNC fresadora.....	73
Figura. 4.45. Estructura final para la CNC de (corte/grabado).....	74
Figura. 4.46. Archivos ejecutables para la interfaz CNC. ....	74
Figura. 4.47. Paquete de instalación del control para smartphone. ....	75
Figura. 4.48. Estructura de la impresora 3D vs CNC fresadora. ....	76
Figura. 4.49. Comprobación del movimiento de los ejes X, Y. ....	77
Figura. 4.50. Comprobación del movimiento del eje Z.....	78
Figura. 4.51. Paso previo al movimiento de los ejes. ....	78
Figura. 4.52. Envío de código G.....	79
Figura. 4.53. Control de velocidad. ....	79
Figura. 4.54. Encendido y apagado del Dremel. ....	80
Figura. 4.55. Posición actual de los ejes.....	80
Figura. 4.56. Botón para seleccionar archivos. ....	81
Figura. 4.57. Reinterpretación de código. ....	81
Figura. 4.58. Enviar a cortar / grabar.....	82
Figura. 4.59. Aviso para empezar a utilizar el software. ....	82
Figura. 4.60. Desplazamiento direccional para la fresadora.....	83
Figura. 4.61. Valor agregado interfaz #2.....	84
Figura. 4.62. Grabado realizado por la CNC fresadora. ....	84
Figura. 4.63. Corte brindado por la CNC fresadora. ....	85
Figura. 4.64. Intercambio de cabezal.....	86
Figura. 4.65. Colocación de plataforma para fresado.....	86
Figura. 4.66. Estructura del módulo. ....	87
Figura. 4.67. Altura de la plancha. ....	88
Figura. 4.68. Archivo original. ....	89

Figura. 4.69. Archivo modificado. .... 90

## LISTA DE TABLAS

Tabla 1.1. Especificaciones técnicas del Atmega 328.....	9
Tabla 1.2. Características técnicas del GT2560 .....	10
Tabla 1.3. Especificaciones técnicas del Atmega 328.....	15
Tabla 1.4 Especificaciones para cristal y condensadores.....	19
Tabla 3.1 Componentes electrónicos.....	46
Tabla 3.2 Herramientas eléctricas y materiales varios.....	47
Tabla 3.3 Software.....	47
Tabla 3.4 Tarjeta electrónica.....	48
Tabla 3.5 Mano de obra.....	48

## LISTA DE ECUACIONES

Ecuación 1.1. Cálculo de la resistencia R2 para el transistor LM317 .....	21
Ecuación 1.2. Cálculo de la corriente en la base en el transistor TIP 122.....	25
Ecuación 1.3. Cálculo de la resistencia para la base del transistor TIP 122.....	25

## RESUMEN

En la actualidad, existen varios modelos de impresoras 3D que se dedican únicamente a la modelación de volúmenes en tres dimensiones, de igual forma, existen sistemas de grabado y corte por CNC que se encuentran estructurados de manera similar pero que utilizan una infraestructura propia. Por esta razón se propone generar una impresora 3D multifuncional, aprovechando la infraestructura de la impresora para crear no solo figuras en tres dimensiones si no también brindar la capacidad de corte y grabado por medio de un cabezal de fresado intercambiable de fácil acceso.

Esto será posible por medio de la creación de un software que interprete código G el cual será posteriormente cargado a la estructura electrónica /actuadores de la impresora para que ejecute el proceso de corte o grabado automáticamente. Este nuevo proceso estará destinado para ser utilizado sobre materiales de baja densidad como: madera, MDF, ambos de un espesor máximo de 1cm, o balsa en base a un diseño asistido por computadora (CAD). Además el nuevo sistema de corte y grabado contará con un control externo de manipulación directa, el cual permitirá la operación del movimiento de la cortadora/grabadora de una forma manual.

***Palabras claves*** – CAD, CNC, código G, Arduino, Processing, App Inventor, puerto serial, Bluetooth, programación, Java.



## **ABSTRACT**

At present, there are several 3D printers models that are dedicated exclusively to modeling volumes in three dimensions, in the other hand, there are engraving and CNC cutting systems that are structured in a similar way working in their own infrastructure. For this reason, it is proposed to generate a multifunctional 3D printer, taking advantage of the printer's infrastructure to create not only figures in three dimensions but to provide cutting and engraving capacity through an easily accessible interchangeable milling head.

This will be possible through the creation of a software that interprets G code which will then be loaded into the electronic structure / actuators of the printer so that it executes the cutting or engraving process automatically. This new process will be destined to be used on materials of low density like: wood, MDF, both of a maximum thickness of 1cm, or raft based on a computer-aided design (CAD). In addition, the new cutting and engraving system will have an external control of direct manipulation, which will allow the movement's operation of the cutter / recorder in a manual way.

***Keywords* - CAD, CNC, G code, Arduino, Processing, App Inventor, serial port, Bluetooth, programming, Java.**

# INTRODUCCIÓN

## **Antecedentes de la situación objeto de estudio.**

Para el desarrollo del presente proyecto de titulación se realiza una investigación de los antecedentes de la situación del objeto a estudiar, ya que, es importante saber sobre las aportaciones que se han realizado en el campo del control numérico por computadora (CNC).

Primero, como se puede observar, a través de la historia siempre se ha tratado de emplear un solo mecanismo para realizar varias tareas diferentes, así lo indica Schvab (2011): Los tornos antiguamente se usaban con una sola herramienta, la diferencia estaba en agregar algunas herramientas para diferentes trabajos en una misma máquina. De esta forma, posterior a la implementación del presente proyecto de titulación, la impresora 3D se encontrará en la capacidad de cortar y grabar por medio de una fresadora en materiales de baja densidad a un espesor máximo de 1cm como: madera, MDF o balsa.

Con el desarrollo de la tecnología, en los años 50 se planificaba a futuro poder manejar en forma automática a una máquina equipada con una herramienta, por esta necesidad nace la creación y desarrollo del Control Numérico Computarizado (CNC) como lo explica Schvab (2011): El arribo del ordenador representó un cambio de la información entregada por el control numérico, permitiendo la evolución de las máquinas que eran utilizadas como herramientas. Ahora bien, el término "control numérico" proviene de las órdenes o instrucciones entregadas a la máquina, las cuales son códigos numéricos. Dichos códigos numéricos son un conjunto de instrucciones que siguen una secuencia lógica construyendo así un programa mecanizado. Para este caso el código numérico que maneja la impresora 3D con la que se va a trabajar es el Código G.

Por otra parte, como lo expresa Herrera (2018): “La implementación de una máquina CNC es ampliamente utilizada en la industria para la creación y moldeo de piezas complejas de manera rápida y precisa. No obstante, la implementación de este tipo de tecnología en la industria resulta sumamente costosa.” (p.2). Por esta razón se busca agregar funciones a equipos que pueden compartir una relación a nivel mecánico y de software, generando así un ahorro económico importante. Esto es lo que busca el presente proyecto de titulación, al crear un software que sea capaz de interpretar un código G generado por un diseño asistido por computadora (CAD).

Además, Herrera (2018), en su trabajo de titulación: “Implementar un sistema de manejo lineal para una mesa de control numérico computarizado”, también elaboró un App que permite controlar la alimentación de la máquina por medio de un smartphone mediante Bluetooth. Es en este punto, donde el prototipo multipropósito fresadora CNC e impresora volumétrica, busca implementar un App que, no solo controle la alimentación si no también el desplazamiento direccional para la fresadora a través de un control remoto.

Así mismo, Turriago (2018) realizó una: “Implementación de una tarjeta de control para una mesa mediante Control Numérico Computarizado con precisión a velocidades de producción, en respuesta a la problemática de las pequeñas y medianas empresas locales por la inaccesibilidad a las máquinas CNC, debido a su alto precio.” (p.5). Como se puede observar la problemática que se presenta para la adquisición de un prototipo CNC es alta en nuestro entorno, pero éste al igual que otros proyectos, solo se enfocan a la generación de un prototipo más accesible económicamente, pero que cumple con una sola función específica y es aquí donde se busca innovar al abaratar aún más el costo para las pequeñas y medianas empresas que pueden adquirir un solo equipo con el cual puedan ejecutar varias funciones.

Como se puede observar por lo antes expuesto, el presente proyecto de titulación busca innovar al unificar las tecnologías que son compatibles, de esta forma, no solo generar un nuevo prototipo sino crear nuevas aplicaciones aprovechando la infraestructura ya existente, para lo cual, es necesario diseñar el sistema de fresado (corte / grabado) en un plano (X - Y) que sea compatible con la impresora 3D. Asimismo se implementará un software que sea compatible con el firmware que se encuentra integrado a la electrónica de

la impresora 3D para la comunicación CAD fresadora. Conjuntamente se busca implementar un control de desplazamiento direccional para la fresadora con el cual se podrá realizar cortes y grabados manuales en un plano (X - Y). Finalmente se demostrará el pleno funcionamiento de las nuevas capacidades que adquirirá la maquinaria.

### **Planteamiento del problema**

Como lo indica el diario El Comercio en una de sus publicaciones de actualidad (I) extraída de la web: La impresora 3D es un equipo que analiza un dibujo y lo fracciona en diferentes capas, para luego superponerlas y construir un objeto con el material elegido. La impresión 3D ha revolucionado diferentes áreas como: la alimentación, la salud, la industria automotriz, etc. En el Ecuador las impresoras 3D están siendo empleadas para fabricar prototipos para el área de la publicidad. Asimismo se elaboran manillas, figuras y jarros personalizados. En la ciudad de Quito. Esteban Armendáriz es precursor de Eard 3D en el país, en lo cual apenas se está iniciando. Él, además de imprimir, también ensambla impresoras y las comercializa. Estas cuestan de USD 1300 a 2200. Cada equipo es preparado en seis horas y ha vendido unos 40 en el Ecuador.

Además se puede percibir el precio de un “Router 3020 CNC fresadora Pcb taladro software bajo pedido” desde el portal de Mercado Libre Ecuador a una cantidad de: 1180\$. Como se puede observar el problema radica en el costo que representa la adquisición de uno de estos equipos. Pero que mejor, si es posible emplear la infraestructura de uno para generar la funcionalidad del otro, esta es la problemática que desea resolver el presente proyecto de titulación al unificar las funcionalidades de una impresora 3D con las de una CNC fresadora (cortadora /grabadora).

### **Justificación**

En la actualidad, existen varios modelos de impresoras 3D que se dedican únicamente a la modelación de volúmenes en tres dimensiones, de igual forma existen sistemas de grabado y corte por CNC que se encuentran estructurados de forma similar pero que utilizan una infraestructura propia. A nivel nacional para adquirir alguno de estos sistemas se debe hacer por separado, es decir invertir en un sistema 3D y en un sistema de

fresado. Por esta razón se propone generar una impresora 3D multifuncional, aprovechando la infraestructura de la impresora para crear no solo figuras en tres dimensiones si no también brindar la capacidad de corte y grabado por medio de un cabezal de fresado intercambiable de fácil acceso.

Esto será posible por medio de la creación de un software que interprete código G el cual será posteriormente cargado a la estructura electrónica /actuadores de la impresora para que ejecute el proceso de corte o grabado automáticamente. Este nuevo proceso estará destinado para ser utilizado sobre materiales de baja densidad como: madera, MDF, balsa, con un espesor máximo de 1cm, en base a un diseño asistido por computadora (CAD). Además el nuevo sistema de corte y grabado contará con un control externo de manipulación directa, el cual permitirá la operación del movimiento de la cortadora/grabadora de una forma manual.

### **Objetivo General:**

Desarrollar una fresadora de corte y grabado CNC a partir de una impresora volumétrica para adquirir un equipo multipropósito.

### **Objetivos Específicos:**

- Diseñar el sistema de fresado (corte / grabado) compatible con la estructura de la impresora volumétrica existente en un plano X, Y.
- Desarrollar el software para la comunicación CAD fresadora.
- Implementar el diseño del fresado y software en la impresora volumétrica existente.
- Implementar un control de desplazamiento direccional para la fresadora a través de un control remoto.
- Validar el funcionamiento del prototipo multipropósito.

### **Alcance**

Se diseñará un sistema de fresado CNC (corte/grabado) en un plano (X – Y) que será compatible con la estructura de la impresora 3D, por esta razón, se implementará un

cabezal que pueda ser fácilmente intercambiado por el cabezal de impresión, así mismo, se deberá adecuar la plataforma de (corte/grabado) para que no interfiera con la plataforma de impresión. Por otra parte se implementará una tarjeta electrónica para efectuar el control de encendido y apagado de una herramienta eléctrica giratoria, dicha herramienta también contará con la capacidad de variar su velocidad.

A nivel de software se diseñará e implementará una interfaz que se encargará de la reinterpretación del código G generado por un “Diseño Asistido por Computadora” (CAD) y de esta manera se efectuará la comunicación entre éste último y la fresadora, haciendo posible la ejecución del corte o grabado desde el mismo software. Adicional a esto, dicha interfaz contará con la capacidad de controlar el movimiento de los ejes (X-Y-Z) en la CNC, también controlará la velocidad de la herramienta giratoria, además contará con un posicionamiento de los ejes X-Y y Z a su home. Ésta misma interfaz tendrá la capacidad para encender y apagar una herramienta giratoria, e igualmente se podrá enviar códigos G propios del firmware Marlín a través de la interfaz hacia la CNC (cortadora/grabadora). Así mismo se podrá visualizar la posición actual de los ejes (X-Y-Z) dentro de la interfaz.

Por otra parte, se diseñará un control remoto de desplazamiento direccional para la fresadora, empleando una aplicación ejecutable en un smartphone bajo el sistema operativo Android.

Como un valor agregado y fuera del alcance inicial del presente proyecto de titulación, el control remoto también contará con la capacidad de encender y apagar la herramienta giratoria, asimismo el control remoto podrá variar la velocidad de dicha herramienta. En este control, también se podrá dibujar sobre una imagen extraída del repositorio del teléfono, para que luego dicho dibujo pueda ser interpretado por la CNC fresadora.

## **Descripción de los capítulos**

Para comenzar la descripción del presente proyecto de titulación, es necesario aclarar que éste se encuentra estructurado a partir de las normas APA sexta edición:

- 
- El capítulo 1 describe la fundamentación teórica necesaria para el desarrollo del presente proyecto de titulación, abordando así temas como: componentes de una impresora 3D a nivel de hardware, software, firmware Marlin, código G, fresadora CNC, circuito integrado Atmega 328, regulador de voltaje LM317, transistor TIP122, servomotor MG995, herramienta giratoria y finalmente todo el software necesario para el funcionamiento del proyecto.
  - El capítulo 2 contempla el marco metodológico que se empleó para establecer la solución más adecuada para resolver el inconveniente demostrado anteriormente en el planteamiento del problema.
  - El capítulo 3 comprende la propuesta que se ofrece para solventar el problema de investigación, aquí se podrán encontrar las factibilidades para realizar el proyecto, así como un estudio técnico y económico.
  - El capítulo 4 abarca la implementación de la propuesta, con la cual se hará posible la utilización de la infraestructura de una impresora 3D para generar las funciones de una fresadora CNC (grabadora/cortadora). Se podrán encontrar diagramas eléctricos, electrónicos, detalle a nivel de hardware y software, entre otros.

# CAPÍTULO 1

## FUNDAMENTACIÓN TEÓRICA

Para hacer posible el presente proyecto de titulación, es necesario realizar una indagación sobre algunos estudios anteriores que se han realizado por parte de otros investigadores, estos mismos guardan una relación cercana con el tema que se desarrolla en este documento, el cual es: prototipo multipropósito fresadora CNC e impresora volumétrica. Si bien no todos los conceptos y aplicaciones técnicas que se han desarrollado en investigaciones previas no serán aplicados de igual forma para este proyecto, por lo menos servirán como una pauta para el desarrollo del mismo.

Y es así que, Robinson (2018) de la Universidad Tecnológica Israel, en su proyecto de titulación “Diseño y construcción de una máquina de corte y grabado láser en materiales de baja densidad para un centro de diseño gráfico”. Explora los diferentes parámetros a nivel mecánico, electrónico y de software que son necesarios para la implementación de su proyecto. De dicho proyecto, se puede extraer la necesidad que conlleva el contar con un software libre, por medio del cual se puedan ejecutar las acciones de corte y grabado. Para la implementación del prototipo multipropósito fresadora CNC e impresora volumétrica, este proceso se realizará por medio de una herramienta giratoria.

Además es evidente, requerir de un control sobre la herramienta a ser utilizada, parámetro que para el caso de este proyecto es la velocidad de la herramienta giratoria, por esta razón, la interfaz a ser desarrollada deberá contar con dicho control ya que este es un parámetro que no será entregado por el CAD. El desarrollo de la interfaz es necesario, debido a que, se trabajará con el firmware ya existente en la impresora 3D al cual se deberá acoplar la nueva CNC (cortadora/grabadora). Por otra parte, como la intención es dejar intacta la infraestructura de la maquinaria antes mencionada, también se debe desarrollar un módulo que albergue y controle la nueva herramienta eléctrica (fresadora).



Por otra parte, Herrera (2018), elaboró un App que le permitía controlar la alimentación de la máquina cortadora por medio de un smartphone mediante Bluetooth. Actualmente el desarrollo de aplicaciones ha ido ganando un gran espacio en el ámbito tecnológico y es por eso que, el desplazamiento direccional para la fresadora a través de un control remoto se realizará por medio de un App, la cual será ejecutada en un smartphone y transmitirá vía Bluetooth bajo el sistema operativo Android. Esta aplicación, también contará con algunas funciones adicionales que ya fueron especificadas anteriormente en el alcance de este proyecto.

Gracias a toda esta información recolectada, se puede definir lo necesario para hacer posible la implementación del: prototipo multipropósito fresadora CNC e impresora volumétrica, a continuación, se detallarán todos los conocimientos teóricos requeridos para la implementación del proyecto, los cuales, en su mayoría han sido adquiridos en el transcurso de la carrera de ingeniería en electrónica digital y telecomunicaciones en la Universidad Tecnológica Israel. A continuación se iniciará con la exploración de los conocimientos básicos de una impresora 3D.

### **1.1 Geeetech Prusa I3 pro B**

La impresora a ser utilizada es una Geeetech Prusa I3 pro B, en los siguientes apartados se detallarán algunos parámetros necesarios para empezar a trabajar con dicha impresora. Geeetech.com ofrece la siguiente información: La impresora 3D Geeetech I3 pro B está diseñada y fabricada por Shenzhen Getech Co., Ltd, basada en Geeetech I3 pro, excepto por su simple montaje, cuenta con una fácil depuración y un rendimiento más estable, en comparación con I3 pro. I3 pro B presenta un potencial recientemente agregado para impresión 3D: admite 5 tipos de filamentos: ABS, PLA, nylon, entre otros. Se pueden crear bisagras flexibles, juntas, piezas con formas; Esto le permitirá sacar el máximo provecho a la impresora 3D. En la figura 1.1 se puede observar su aspecto físico, mientras que en la tabla 1.1 se pueden apreciar sus especificaciones técnicas.



**Figura 1.1. Impresora 3D Geeetech I3 pro B.**

Fuente: Geeetech, (2016).

**Tabla 1.1. Especificaciones técnicas del Atmega 328**

### Especificaciones técnicas

#### **Especificaciones de Impresión:**

Volumen de trabajo: 200  
x200x180mm  
Resolución de capas: 0.1-0.3mm  
Precisión: 0.1-0.3mm  
Diámetro del filamento: 1.75mm,  
3mm  
Tipo de filamento: ABS/ PLA

#### **Temperatura:**

Máxima temperatura de la cama de  
impresión: alrededor de 110°C  
Máxima temperatura de extrusión:  
alrededor de 240°C

#### **Software:**

Sistemas Operativos:  
Windows/Mac/Linux  
Formato del archivo: .STL

#### **Eléctrica:**

Fuente de poder:  
input: 115V/1.5A 230V/0.75A  
Output : DC12V/0-15A  
Conectividad: USB, SD card

**Mecánica:**

Cuerpo de Acrílico de 8 mm de espesor (negro/ transparente)  
 Plataforma de Impresión: Acrílico con plataforma de vidrio + MK2A cama caliente  
 Motores: 1.8° ángulo de paso con 1/16 micro-pasos.

Fuente: Geeetech, (2016).

### 1.1.1 Tarjeta de control GT2560

Es necesario conocer con qué tarjeta de control trabaja la impresora 3D, debido a que el proyecto debe adaptarse a la infraestructura de la misma, a continuación se presenta una descripción de la tarjeta de control: Geeetech.com: GT2560 es una placa compacta que se integra con la poderosa función del Arduino mega2560 + Ultimaker. Cuenta con 256k de memoria, la frecuencia de operación es de 16 Mhz, el convertidor USB de alto rendimiento FT232RQ, todo esto hace del GT2560 una opción ideal para un tablero de control de impresora 3D. En la tabla 1.2 se presentan las características técnicas de la tarjeta de control GT2560, por otra parte, se puede apreciar su representación gráfica en la figura 1.2.

**Tabla 1.2. Características técnicas del GT2560**

#### Características

- **Atmega2560 es el chip de control principal, junto con el convertidor serial USB FT232RQ de alto rendimiento.**
- **Fuente de energía: voltaje de operación es 12V-24V.**
- **Salidas de ventilador de 12 V y una salida PWM para LED u otro equipo de iluminación.**
- **Admite 5 controladores paso a paso A4988 (3 para ejes X / Y / Z y 2 para extrusoras.**
- **Sensores de temperatura de entrada (se recomienda 100k termistor)**

- La interfaz extendida de LCD y tarjeta SD es compatible con LCD2004 y LCD 12864, con la que puede imprimir sin su PC.
- El firmware es el Marlin, que es conocido por su estabilidad, facilidad de uso y alto rendimiento.

Fuente: Geeetech, (2016).

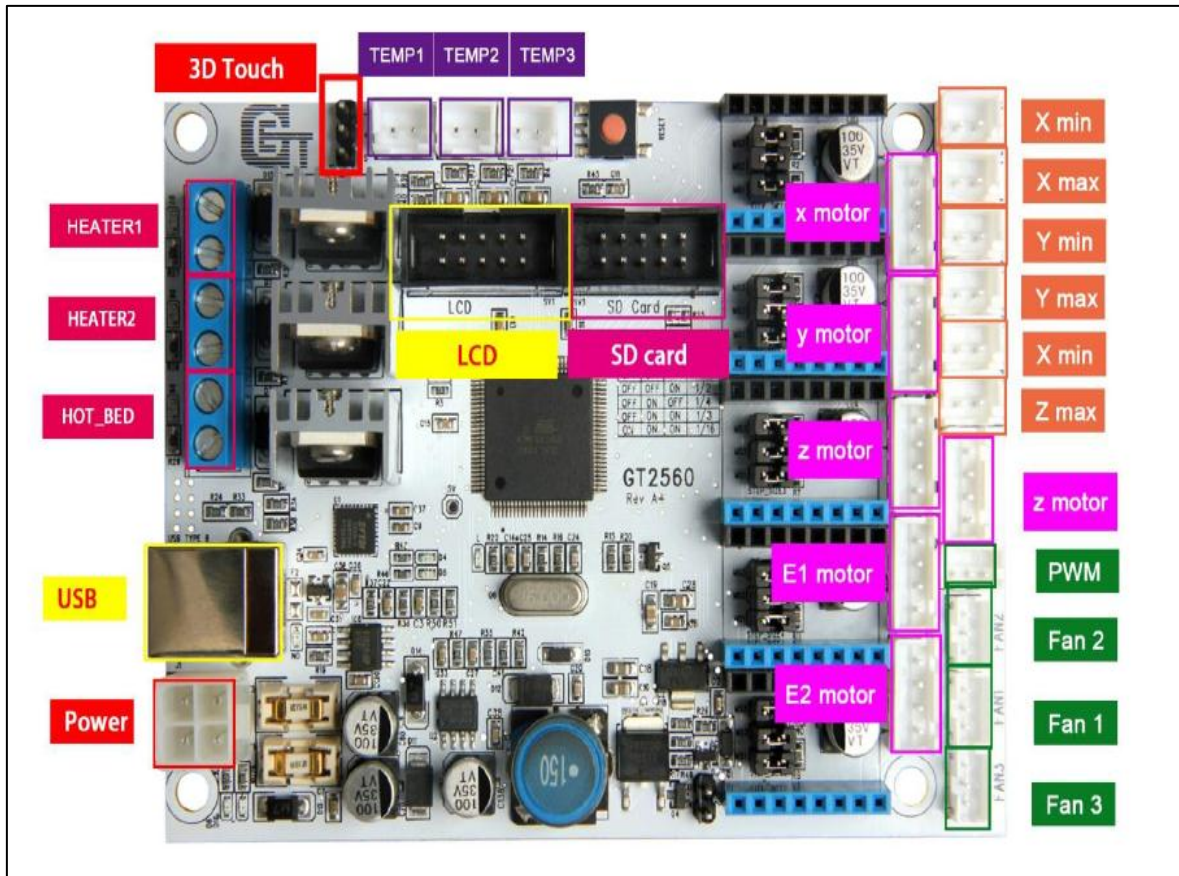


Figura 1.2. Tarjeta de control GT2560.

Fuente: Geeetech, (2016).

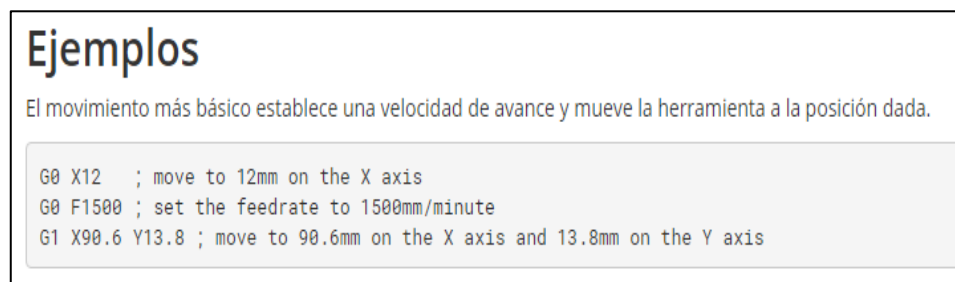
### 1.1.2 Firmware Marlin

Para el control de la impresora es necesario instalar el firmware Marlin, ya que, éste es el encargado de leer línea a línea y ejecutar el código G, así se hace posible la impresión en 3 dimensiones. Es necesario el conocimiento del firmware ya que la cortadora CNC se adaptará al firmware ya instalado.

Marlinfw.org: Marlin es un firmware de código abierto para la familia RepRap de replicadores de prototipos rápidos, popularmente conocido como "impresoras 3D". Se derivó de Sprinter y GRBL, y se convirtió en un proyecto de código abierto independiente el 12 de agosto de 2011 con su versión Github. Marlin es gratuito para todas las aplicaciones.

El firmware Marlin se ejecuta en la placa principal de la impresora 3D, gestionando todas las actividades en tiempo real de la máquina. Coordina los calentadores, los motores paso a paso, los sensores, las luces, la pantalla LCD, los botones y todo lo relacionado con el proceso de impresión 3D.

El lenguaje de control para Marlin es un derivado del G-código. A medida que Marlin recibe comandos de movimiento, los agrega a una cola de movimiento que se ejecutará en el orden recibido. La "interrupción paso a paso" procesa la cola, convirtiendo los movimientos lineales en pulsos electrónicos de tiempo preciso en los motores paso a paso. Incluso a velocidades modestas, Marlin necesita generar miles de pulsos paso a paso cada segundo. (Por ejemplo,  $80 \text{ pasos por mm} * 50 \text{ mm} / \text{s} = 4000 \text{ pasos por segundo}$ ). Dado que la velocidad de la CPU limita la velocidad de movimiento de la máquina, siempre se está buscando nuevas formas de optimizar la interrupción por pasos. En la figura 1.3 se muestra un ejemplo del código G empleado por Marlin.



**Figura. 1.3. Código G ejecutado por Marlin.**

Fuente: Marlin, (2018).

### 1.1.3 Código G

En los inicios de la programación de máquinas de control numérico computarizado se utilizaba un lenguaje de bajo nivel denominado G, el cual es un lenguaje de programación vectorial por el cual se describen acciones simples, dicho lenguaje está

acompañado de entidades geométricas sencillas, básicamente segmentos de recta y arcos de circunferencia.

El nombre G proviene del hecho de que el programa está formado por instrucciones generales. En la actualidad existen diferentes adaptaciones de programación con códigos G, pero gracias al estándar ISO 6983 / EIA RS274 el código puede ser empleado en distintas máquinas CNC de manera directa o con ajustes menores. En la figura 1.4 se puede apreciar una línea de código G.

Por ejemplo: Las siguientes líneas ordenan a una fresadora de CNC que ejecute en la línea de código 100 un corte relativo al origen con un avance de 20 in./min a lo largo del eje X 1.25 in. y del eje Y 1.75 in.

**N100 G01 X1.25 Y1.75 F20**

**Figura. 1.4. Código G.**

Fuente: Ing. Ricardo Jiménez, (2003).

#### **1.1.4 Diseño asistido por computadora (CAD)**

Torres (2015): “una herramienta CAD es un sistema software que aborda la automatización global del proceso de diseño de un determinado tipo de ente” (p.3). Un sistema CAD está basado en la representación computacional de un modelo gráfico, esto permite efectuar modificaciones en el dibujo a detalle y permite la documentación del diseño, de esta forma posibilita la utilización de métodos numéricos para ejecutar simulaciones sobre el modelo o dibujo por ende esto sirve como una alternativa a la construcción de prototipos.

Torres (2015): “otro aspecto importante de la automatización del diseño es la posibilidad de utilizar la información del modelo como base para un proceso de fabricación asistida por ordenador (CAM)” (p.3). Los sistemas CAM son empleados para automatizar

un proceso de fabricación, así como del control de máquinas / herramientas, Por otra parte un sistema CAM debe diseñarse para hacer uso de la base de datos del modelo gráfico.

### 1.1.5 Fresadora CNC

Una fresadora CNC, es una máquina que actúa bajo un código emitido por una computadora, llamado código G, esta maquinaria es capaz de realizar cortes y grabados gracias a una herramienta giratoria que se encuentra provista de una fresa, la cual tiene un aspecto parecido al de una broca pero que generalmente no cuenta con una punta cónica. Existe un sinnúmero de fresadoras, adecuadas para trabajar en distintas densidades de materiales, pero la que se va a adecuar en el presente proyecto de titulación será una fresadora CNC para materiales de máximo 1 cm de espesor (baja densidad). A continuación en la figura 1.5 se representa una fresadora CNC.



**Figura. 1.5. Fresadora CNC.**

Fuente: Julio Aguilar, (2013).

Hasta el momento se han presentado los conceptos básicos de los diferentes equipos que se relacionan directamente para el: prototipo multipropósito fresadora CNC e impresora volumétrica, en los siguientes apartados se abordarán temas que son necesarios para el desarrollo de dicho proyecto.

## 1.2 Circuito integrado Atmega 328

El Atmega 328, es un circuito integrado de alto rendimiento que cumple con las siguientes características técnicas, las cuales son descritas por Atmel Corporation (2016) en su datasheet como: “El núcleo AVR® de Atmel combina un rico conjunto de instrucciones con 32 registros de trabajo de uso general. Todos estos 32 registros están conectados directamente a la unidad lógica aritmética (ALU). La arquitectura resultante es un código más eficiente, al tiempo que logra rendimientos hasta diez veces más rápidos que los microcontroladores CISC convencionales” (p.9). Dicho dispositivo será utilizado para el control de encendido/apagado y control de velocidad de la herramienta giratoria (fresadora).

### 1.2.1 Especificaciones técnicas del Atmega 328

Las especificaciones técnicas para el Atmega 328, se presentan de la siguiente forma: En la tabla 1.3 se pueden observar las especificaciones técnicas que componen a un Atmega 328, así mismo en la figura 1.6 se muestra la distribución de los pines del circuito integrado.

**Tabla 1.3. Especificaciones técnicas del Atmega 328**

Microcontrolador	ATmega328
Voltaje de operación	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (límite)	6-20V
Pines de entrada/salida digitales	14
Pines de entrada analógicos	6
Corriente DC para pines I/O	40 mA
Corriente DC para 3.3V	50 mA
Memoria Flash	32 KB (ATmega328)
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)



Clock Speed

16 MHz

Fuente: Atmel Corporation, (2016).

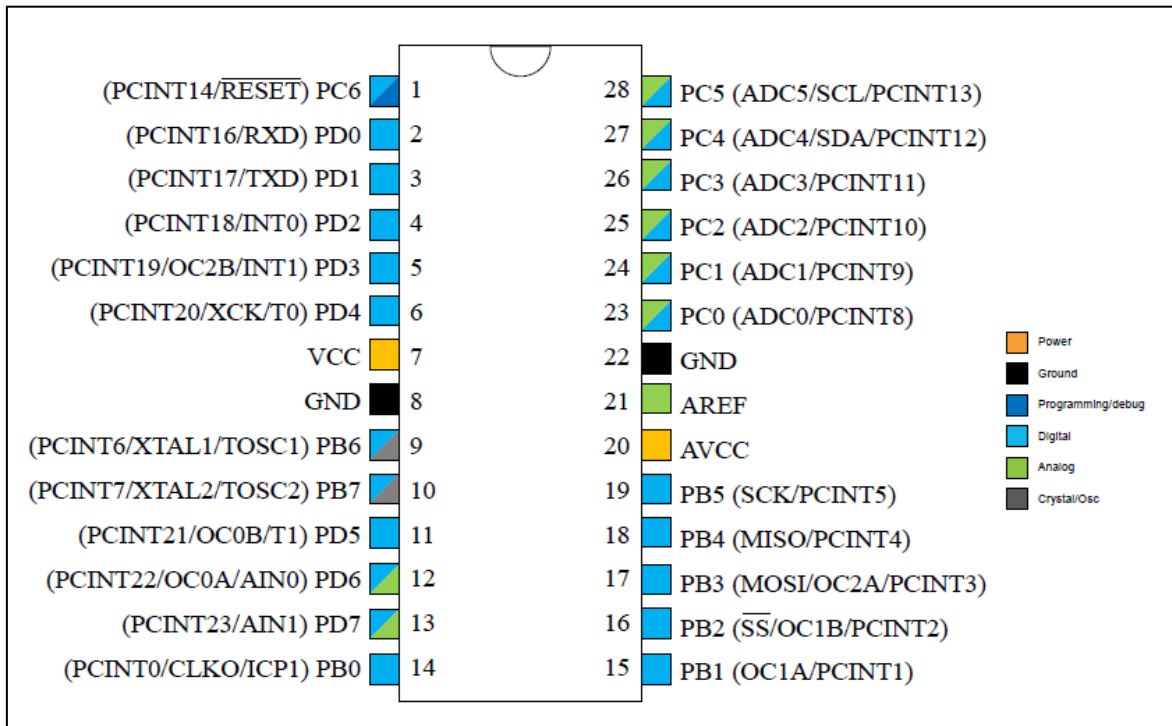


Figura. 1.6 Distribución de pines para el Atmega 328.

Fuente: Atmel Corporation, (2016).

### 1.2.2 Descripción de pines (Atmega 328)

La descripción de pines fue tomada directamente del datasheet del Atmega 328, dicha descripción se realiza para entender posteriormente la relación que guardan estos pines con la distribución de los pines de un Arduino Uno. Esto es necesario ya que se realizará una utilización del software Arduino para la programación del microcontrolador y de esta manera poder efectuar el control de las herramientas.

- **VCC.**  
Tensión de alimentación digital.
- **GND.**  
Tierra

- **Port B (PB[7:0]) XTAL1/XTAL2/TOSC1/TOSC2**

El Puerto B es un puerto bidireccional de entrada/salida a 8 bits y con resistencias pull-up interna.

- **Port C (PC[5:0])**

Al igual que el puerto B posee la capacidad de ser usado como E/S además posee la capacidad de activarse las resistencias pull-up o dejarse en alta impedancia. Este puerto posee 7 bits, además de poseer la opción de ser usado como entrada de ADC

- **PC6/RESET**

Si el fusible (RSTDISBL) es seleccionado PC6 se usa como un pin de entrada, en caso de no ser seleccionado se usa como un pin de reset.

- **Port D (PD[7:0])**

Es similar al PORTB con la diferencia de no poseer entrada para el temporizador/contador 2.

- **AVCC**

Es la alimentación interna en caso de usarse el convertidor A/D

- **AREF**

Es el pin de referencia analógico para el convertidor A/D.

- **ADC[7:6]**

Sirve como entradas analógicas al convertidor A/D. Estos pines son alimentados desde la fuente analógica y sirve como canales ADC (convertidor análogo digital) de 10 bits.

### 1.2.3 Equivalencia de pines de Atmega 328 para Arduino.

La equivalencia de los pines del Atmega 328 hacia Arduino, se utiliza para trabajar con el microcontrolador como si se tratase de un Arduino Uno, dicha equivalencia de pines se muestra a continuación en la figura 1.7.

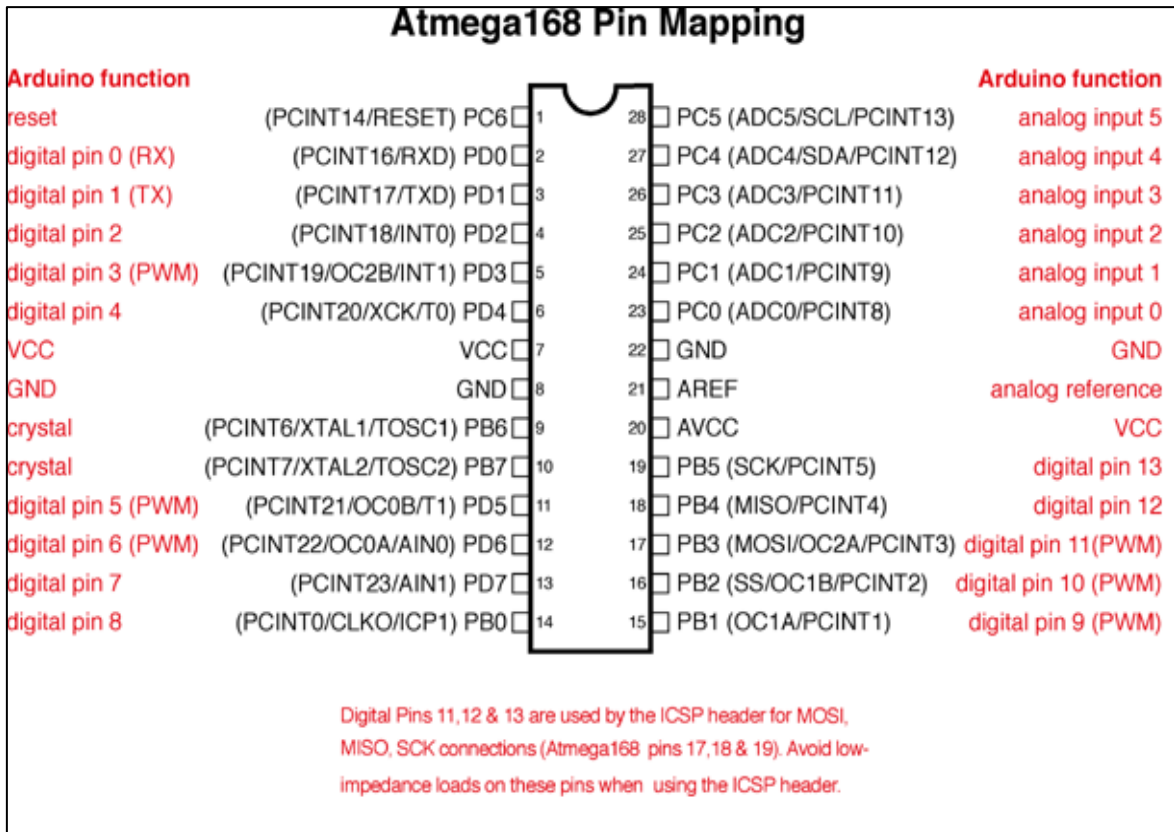


Figura. 1.7 Equivalencia de pines de Atmega 328 para Arduino.

Fuente: Atmel Corporation, (2016).

### 1.2.4 Conexiones del Atmega 328

A continuación se presentan las conexiones necesarias para que el Atmega 328 entre en funcionamiento, estas conexiones son prácticamente un estándar ya establecido, por lo cual el mismo datasheet del Atmega brinda esta información. A continuación en la figura 1.8 se pueden observar las conexiones de alimentación, ground, reset, cristal, led indicador de funcionamiento.

De igual forma el cristal sugerido para utilizar junto a sus condensadores los indica Atmel como: Los pines XTAL1 y XTAL2 son entrada y salida y pueden ser configuradas para su uso como un oscilador. Se puede utilizar condensadores cerámicos C1 y C2 los cuales siempre deben ser de igual capacidad. El valor óptimo de los condensadores depende del cristal en uso. Atmel Corporation (2016). Los valores correspondientes de los condensadores según su cristal se pueden observar en la tabla 1.4.

Tabla 1.4 Especificaciones para cristal y condensadores.

Rango de frecuencia [MHz]	CKSEL[3:1](2)	Rango para C1 y C2 [pF]
0.4 - 0.9	100(3)	–
0.9 - 3.0	101	12 - 22
3.0 - 8.0	110	12 - 22
8.0 - 16.0	111	12 - 22

Fuente: Atmel Corporation, (2016).

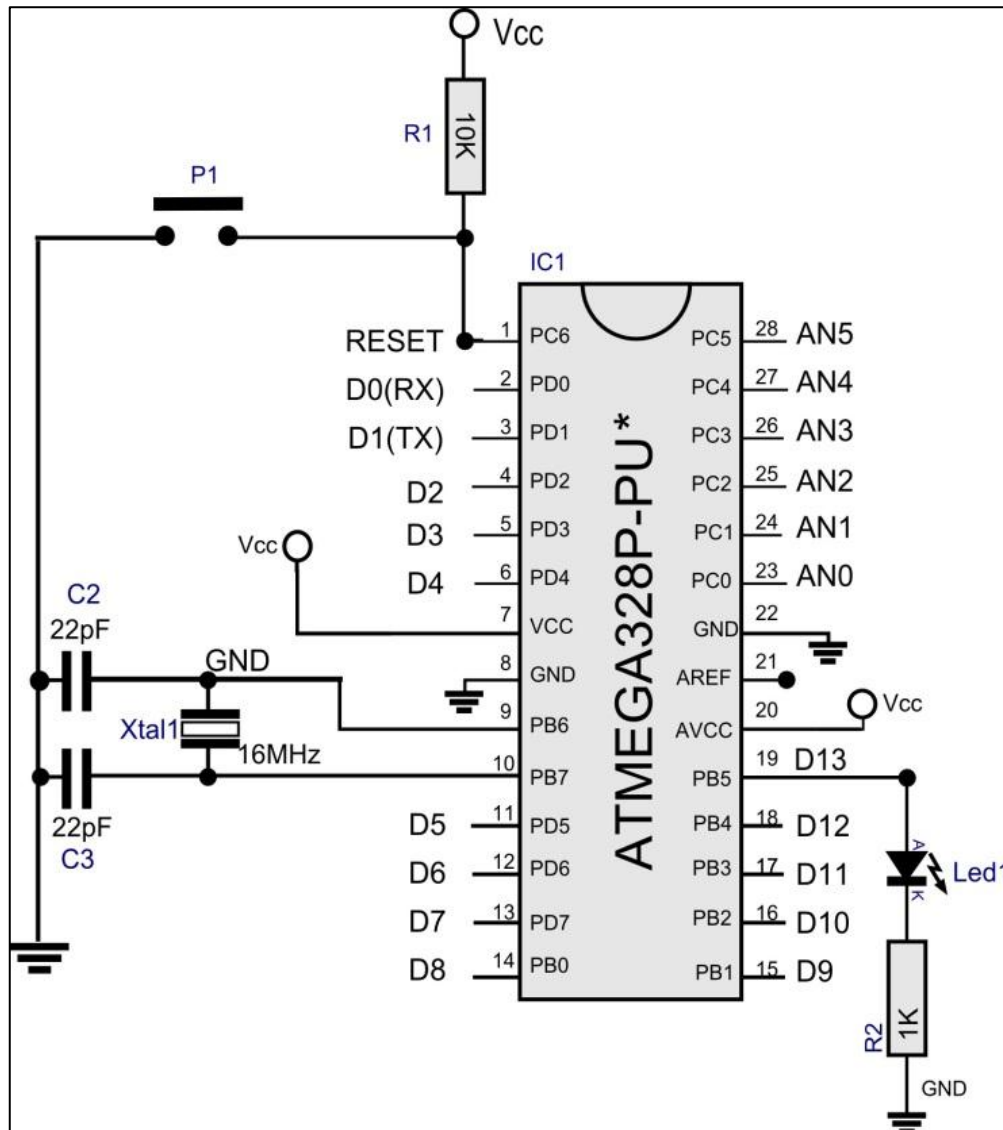


Figura. 1.8 Conexiones para el Atmega 328.

Fuente: Atmel Corporation, (2016).

### 1.2.5 Migración de Arduino a un microcontrolador Atmega 328

Para la utilización del Atmega 328 como si se tratase de un Arduino pero sin la necesidad de contar con su placa, es necesario cargar la secuencia de inicio de Arduino en el microcontrolador, este procedimiento se encuentra especificado en la página oficial de Arduino, en la cual se brinda un esquema de conexión y se especifican los pasos a seguir en el software Arduino. En la figura 1.9 se muestra el esquema electrónico necesario para efectuar este procedimiento. Por otra parte los pasos a seguir son los siguientes:

1. Abrir el software Arduino
2. Conectar la placa del Arduino Uno a la PC
3. Descargar el programa bootloader de la página oficial de Arduino como ISP
4. Cargar el programa al Arduino Uno
5. Desconectar el Arduino Uno de la PC
6. Montar el diagrama electrónico sugerido por Arduino en protoboard
7. Conectar nuevamente la placa de Arduino Uno a la PC
8. Abrir el software de Arduino
9. Dirigirse a la pestaña herramientas / programado / Arduino as ISP
10. Nuevamente a herramientas / grabar secuencia de inicio

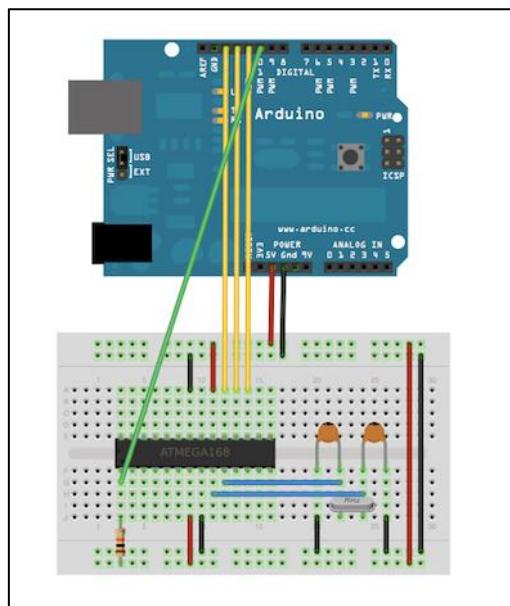
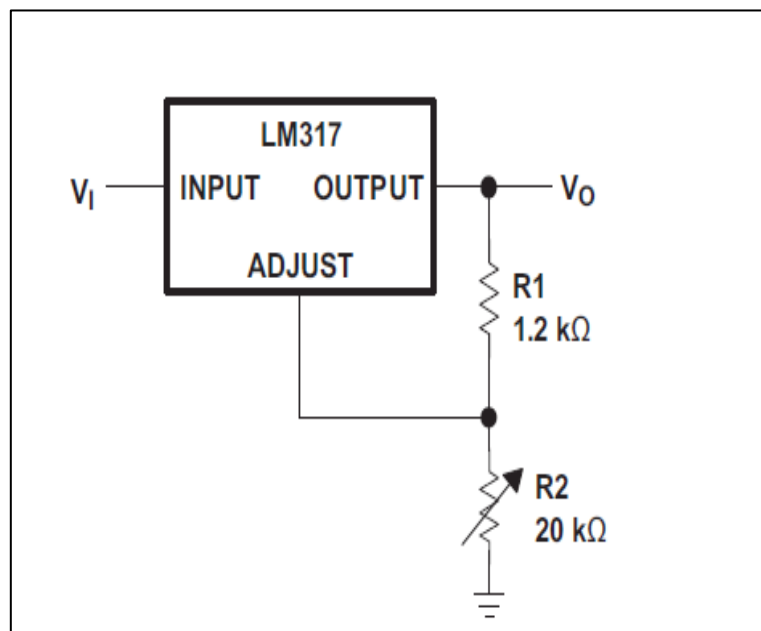


Figura. 1.9 Esquema para grabar secuencia de inicio de Arduino.

Fuente: Arduino, (2018).

### 1.3 Regulador de voltaje LM317

Texas Instruments (2016): “El dispositivo LM317 es un regulador de voltaje positivo ajustable de tres terminales capaz de suministrar más de 1.5 A en un rango de voltaje de salida de 1.25 V a 37 V. Requiere solo dos resistencias externas para configurar la tensión de salida” (p.1). En la figura 1.10 se puede observar su representación gráfica.



**Figura. 1.10** Regulador de voltaje Lm317.

Fuente: Texas Instruments, (2016).

Para calcular el valor de la resistencia R2 en el diagrama del regulador de voltaje del Lm317 es necesario aplicar la siguiente (Ecuación 1.1), en la cual, se necesita de un valor aproximado de R1 y el valor de voltaje de salida requerido. La presente ecuación se puede encontrar en el datasheet del LM317 en la página número 13.

$$V_{out} = 1.25v \left( 1 + \frac{R2}{R1} \right) \quad \text{(Ecuación 1.1)}$$

El fabricante del dispositivo también da la siguiente recomendación para la colocación de un condensador de 1uF a la salida del regulador: Texas Instruments (2016): El CO mejora la respuesta transitoria, pero no es necesario para la estabilidad” (p.10).

### 1.3.1 Regulador de voltaje LM317 configuración para salida de hasta 4 A

Texas Instruments (2016): Esta configuración mantiene la corriente de salida a 4A, mientras tiene la capacidad de ajustar el voltaje de salida usando una resistencia de (1.5 k $\Omega$  como se ve en el esquema). Dicha configuración se muestra en la figura 1.11.

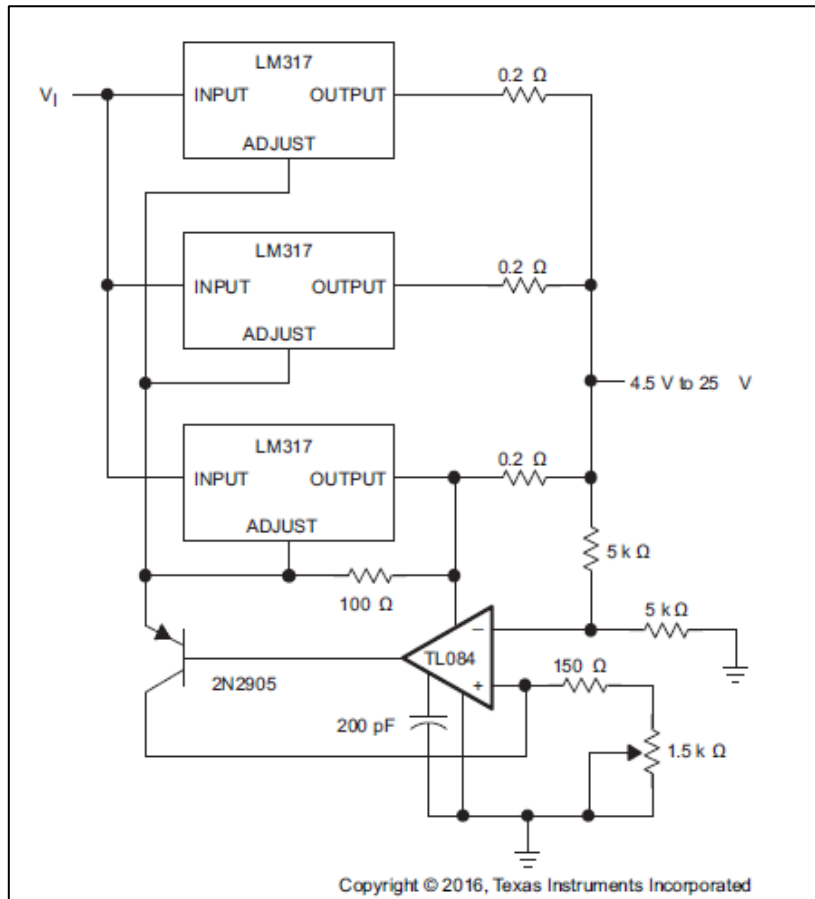


Figura. 1.11. Regulador LM317 configuración para salida de hasta 4 amperios.

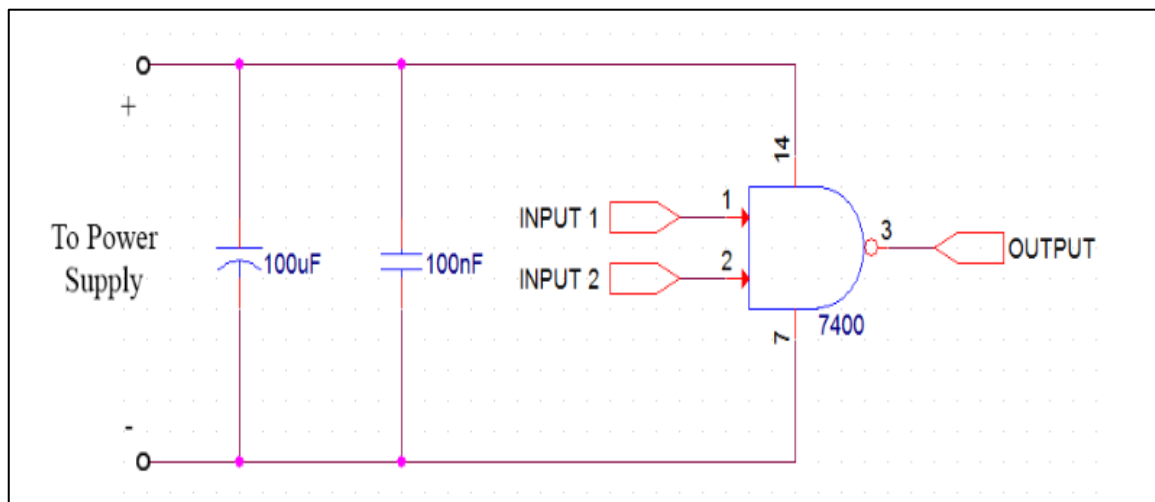
Fuente: Texas Instruments, (2018).

### 1.3.2 Capacitores para fuente de alimentación

Gracias a una investigación en la web, se encontraron ciertas recomendaciones para una PCB que contenga circuitos integrados, dicha investigación arrojó que, lo más usual es que todos los circuitos integrados cuenten con dos tipos de condensadores para su alimentación, los cuales son: condensadores de desacoplo y condensadores de bulk.

jmnelectronics.com: ofrece la siguiente información y se refiere a los primeros como condensadores en el orden de nano faradios y estos se encargan de proporcionar un filtrado de ruido en la señal digital de la fuente. Por otra parte, los capacitores tipo bulk, son condensadores del orden de microfaradios y se utilizan como una reserva de energía que alimente al circuito integrado ante la posible bajada o subida en la tensión de alimentación. (<http://jmnelectronics.com/archives/726>).

Así mismo, verificando dicha información se hallaron los valores generalmente utilizados para estos condensadores, como lo indica, [capacitorguide.com](http://www.capacitorguide.com): Los condensadores de desacoplamiento se utilizan a menudo para desacoplar el circuito de la fuente de alimentación. Algunos componentes requieren una fuente de alimentación estrictamente regulada para funcionar correctamente. Un buen ejemplo son los microcontroladores y los microprocesadores. Si hay un pico de voltaje, el programa cargado en el procesador podría omitir las instrucciones y comenzar a comportarse de manera impredecible. Una buena regla general para los circuitos digitales es utilizar un solo condensador cerámico de 100 nF para cada circuito integrado lógico, así como un solo condensador electrolítico más grande hasta unos pocos cientos de  $\mu\text{F}$  (<http://www.capacitorguide.com/coupling-and-decoupling/>). En la figura 1.12 se puede observar la colocación de los condensadores descritos anteriormente.



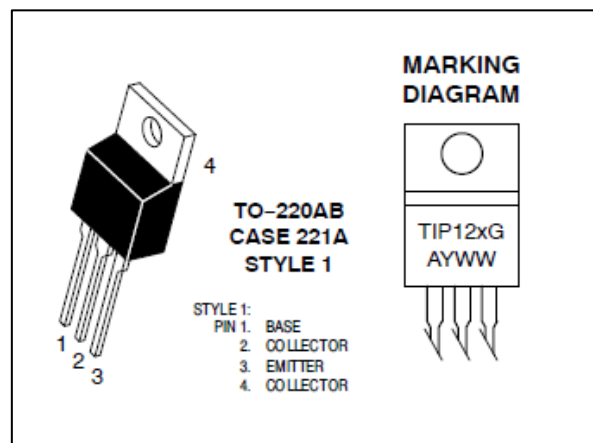
**Figura. 1.12. Capacitor de desacoplo y bulk**

Fuente: Capacitorguide.com, (2018).



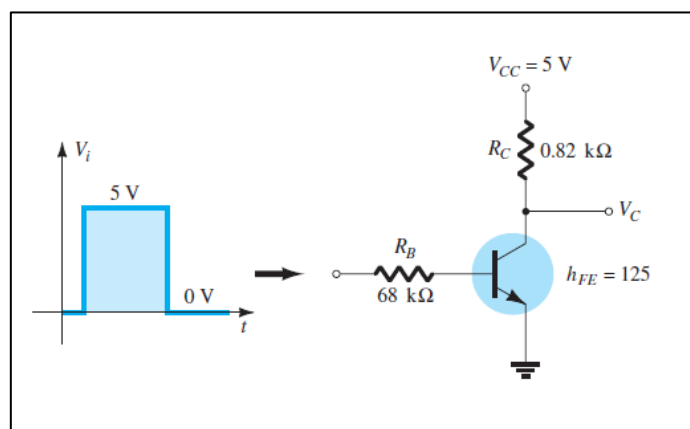
## 1.4 Transistor TIP122

On Semiconductor (2014): El TIP122 es un transistor NPN que se utiliza para la conmutación lineal de potencia media. En la figura 1.13 se puede observar su representación gráfica y distribución de pines. Para el presente proyecto de titulación dicho transistor será utilizado como un interruptor, esta es una de sus funciones como lo indica Boylestad (2009): “La aplicación de transistores no se limita únicamente a la amplificación de señales. Mediante un diseño apropiado se pueden utilizar como interruptores en computadoras y sistemas de control” (p.206). La figura 1.14 muestra el funcionamiento del transistor como un interruptor.



**Figura. 1.13. TIP122, distribución de pines.**

Fuente: On semiconductor, (2014).



**Figura. 1.14. Transistor como interruptor.**

Fuente: Boylestad, (2009).

Para el cálculo de la corriente y la resistencia de base en el transistor, es necesario emplear las ecuaciones 1.2 y 1.3. Estas ecuaciones han sido extraídas del libro Boylestad (2009), capítulo 4, página 206.

$$IB \max \cong \frac{Ic \text{ sat}}{\beta cd} \quad (\text{Ecuación 1.2})$$

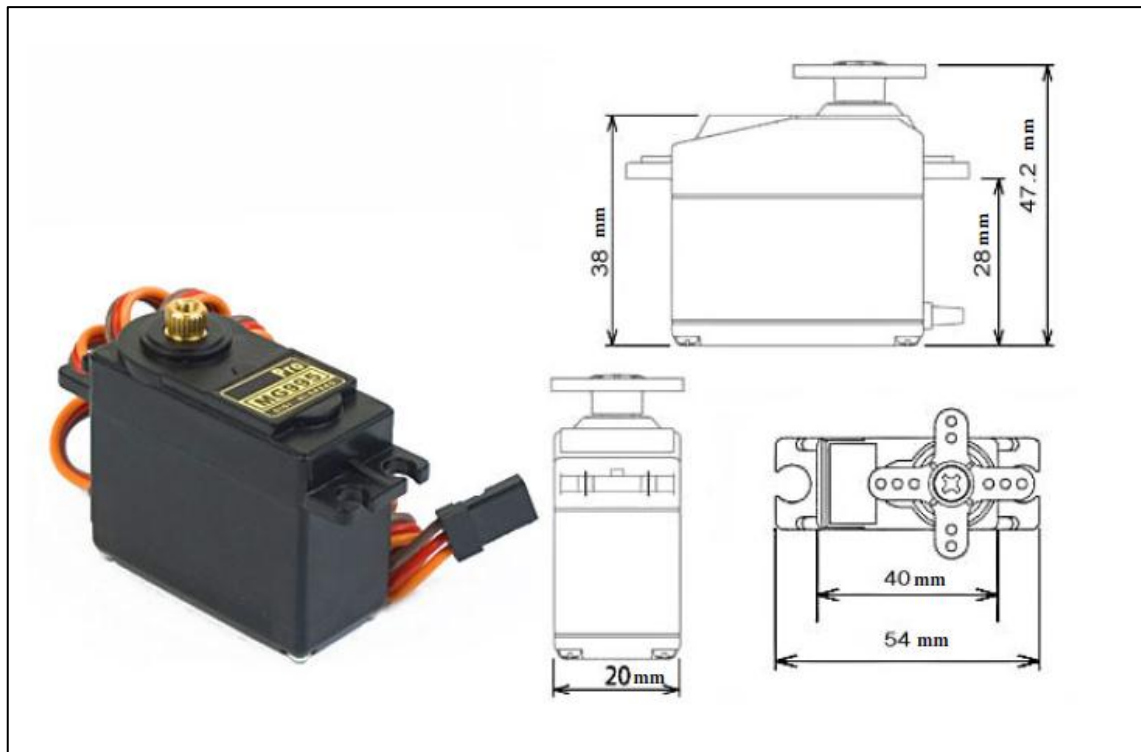
$$IB = \frac{vi-0.7v}{RB} \quad (\text{Ecuación 1.3})$$

## 1.5 Servomotor MG995

El servomotor Tower MG995 cuenta con un alto torque. Es robusto, de alto rendimiento, compacto, con engranes de metal, tamaño estandarizado. En su paquete incluye accesorios para ser montado en cualquier proyecto de electrónica, robótica. En la figura 1.15 se puede apreciar sus dimensiones y en la figura 1.16 su distribución de pines y longitud de pulso. El servomotor MG995, será el encargado de la manipulación de la velocidad en el Dremel, este procedimiento se hará de una forma mecánica utilizando el torque del motor para efectuar un movimiento sobre el chasis de la herramienta giratoria.

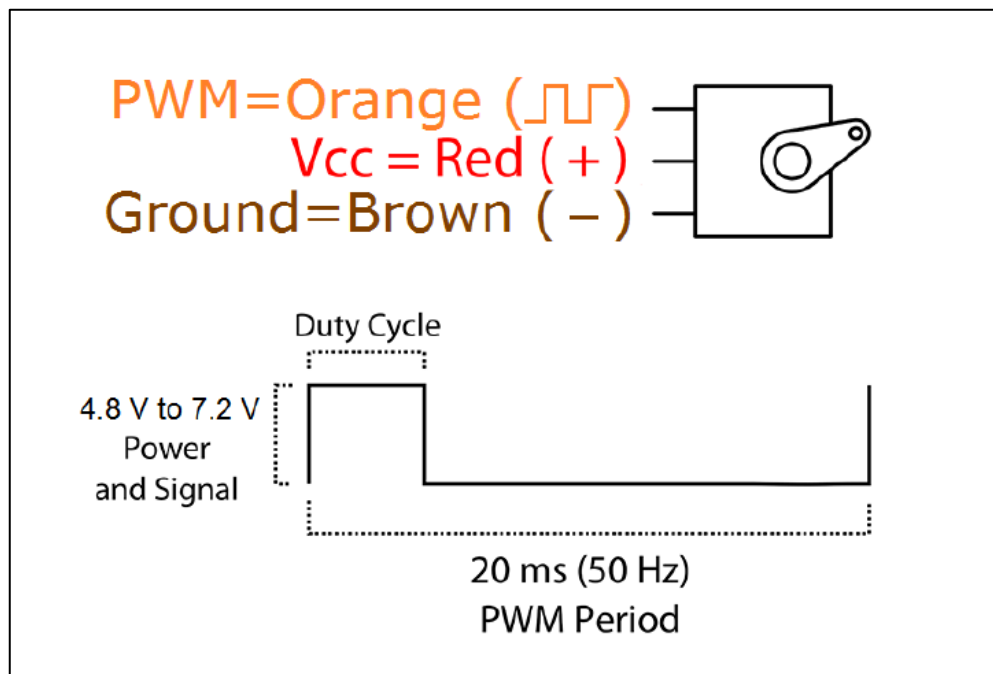
### Especificaciones:

- **Peso:** 55 g
- **Dimensión:** 40.7 x 19.7 x 42.9 mm aprox.
- **Torque:** 8,5 kgf por cm - (4,8 V), 10 kgf por cm - (6 V)
- **Velocidad de operación:** 0.2 s / 60° (4.8 V), 0.16 s / 60° (6 V)
- **Voltaje de operación:** 4.8 V a 7.2 V
- **Rango de temperatura:** 0 °C - 55 °C.
- **Ángulo de rotación:** 120 grados. (+ - 60 desde el centro)



**Figura. 1.15. Servomotor MG995.**

Fuente: Marlin P. Jones and Associates inc, (2014).



**Figura. 1.16. Distribución de pines y longitud de pulso.**

Fuente: Marlin P. Jones and Associates inc, (2014).

## 1.6 Herramienta giratoria Dremel 3000

Dremel es una herramienta giratoria de velocidad variable, con capacidad para cortar, lijar, limpiar, pulir, esculpir, grabar, esmerilar, afilar, fresar y perforar. Puede trabajar en superficies como: madera, vidrio, plástico entre otras. A continuación se representa en la figura 1.17 su imagen y en la figura 1.18 sus datos técnicos.



**Figura. 1.17. DREMEL 300.**

Fuente: Dremel.com, (2018).

REGISTRO TÉCNICO	
<b>Velocidad</b>	variable
<b>Utilizado para</b>	cortar, lijar, limpiar, pulir, esculpir, grabar, esmerilar, afilar, fresar y perforar
<b>Aplicado en</b>	vidrio, fibra de vidrio, madera, metal, plástico, cerámica, azulejo, drywall, avenarfa, cartón y EVA
<b>Rotación (rpm)</b>	10.000 - 32.000
<b>Voltaje</b>	127/220 V
<b>Watts</b>	130W (127 V) / 90W (220 V)
<b>Garantía</b>	2 años
<b>Dimensiones (cm)</b>	26 x 14 x 7,6
<b>Peso (kg)</b>	0,8

**Figura. 1.18. DREMEL 300 datos técnicos**

Fuente: Dremel.com, (2018).

### 1.6.1 Dremel eje flexible 225

Dremel eje flexible 225 es un complemento para ser instalado fácilmente en el Dremel 300, este aditamento entrega mayor comodidad, practicidad y precisión al momento de trabajar. En el presente proyecto de titulación será utilizado para generar un cabezal más liviano y fácil de intercambiar. En la figura 1.19 se puede apreciar dicho complemento.



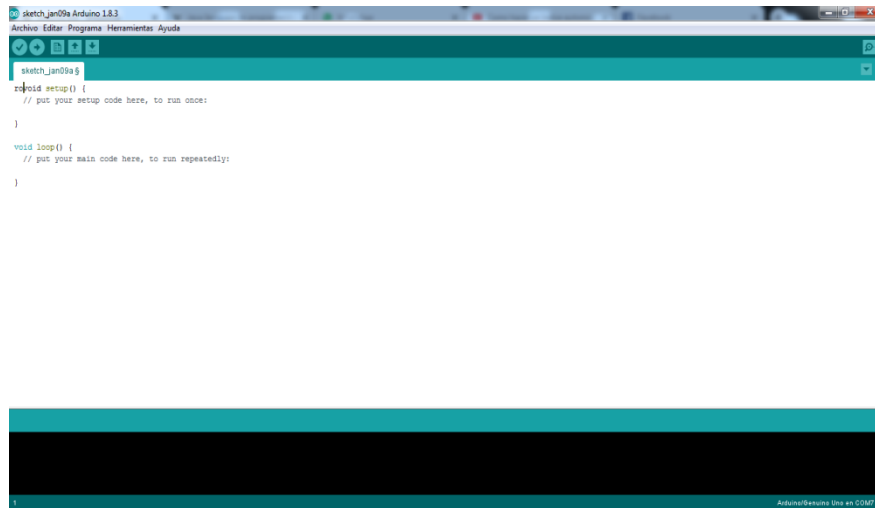
**Figura. 1.19. Dremel eje flexible 225.**

Fuente: Dremel.com, (2018).

El software necesario para desarrollar el presente proyecto de titulación consta de: Arduino, Processing, MIT App Inventor 2, Serial Bluetooth Terminal, Solidworks, Inkscape, Cura, Proteus. El concepto de cada uno de estos programas será abordado a continuación.

## 1.7 Software Arduino

El (IDE) o más conocido como Entorno de Desarrollo Integrado, es un software que se utiliza para crear, compilar y cargar programas en la placa Arduino, esto se puede realizar en diferentes plataformas como Windows, Linux y MacOS. IDE de Arduino utiliza lenguaje de programación Java. En la figura 1.20 se muestra su interfaz.

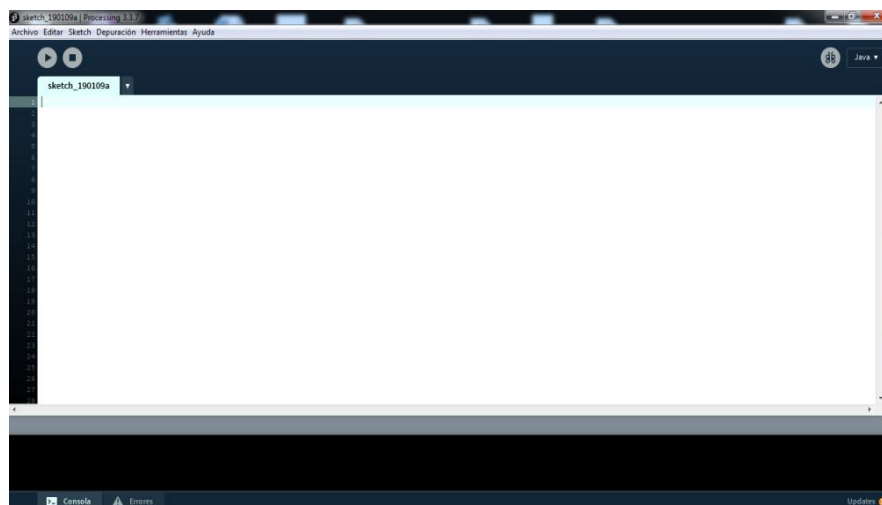


**Figura. 1.20. IDE Arduino.**

Fuente: Arduino, (2018).

## 1.8 Processing

Processing es un software flexible y con un lenguaje para aprender a codificar en el contexto de las artes visuales. Desde 2001, Processing ha incentivado la educación en el tema de software dentro de los conocimientos visuales. Hay decenas de miles de estudiantes, artistas, diseñadores, investigadores y aficionados que utilizan Processing para el aprendizaje y la creación de prototipos. Su interfaz es similar a Arduino como se puede observar a continuación en la figura 1.21.



**Figura. 1.21. Interfaz de Processing.**

Fuente: Processing, (2018).

## 1.9 MIT App Inventor 2

MIT App Inventor, es un entorno de programación visual intuitivo que permite a todos, crear aplicaciones totalmente funcionales para teléfonos inteligentes y tabletas. Es una herramienta basada en bloques lo cual facilita la creación de aplicaciones complejas y de alto impacto en mucho menos tiempo que los entornos de programación tradicionales. El proyecto MIT App Inventor persigue generalizar el desarrollo de software al permitir que todas las personas, especialmente los jóvenes, pasen del consumo de tecnología a la creación de tecnología. En la figura 1.22 se puede apreciar su interfaz.

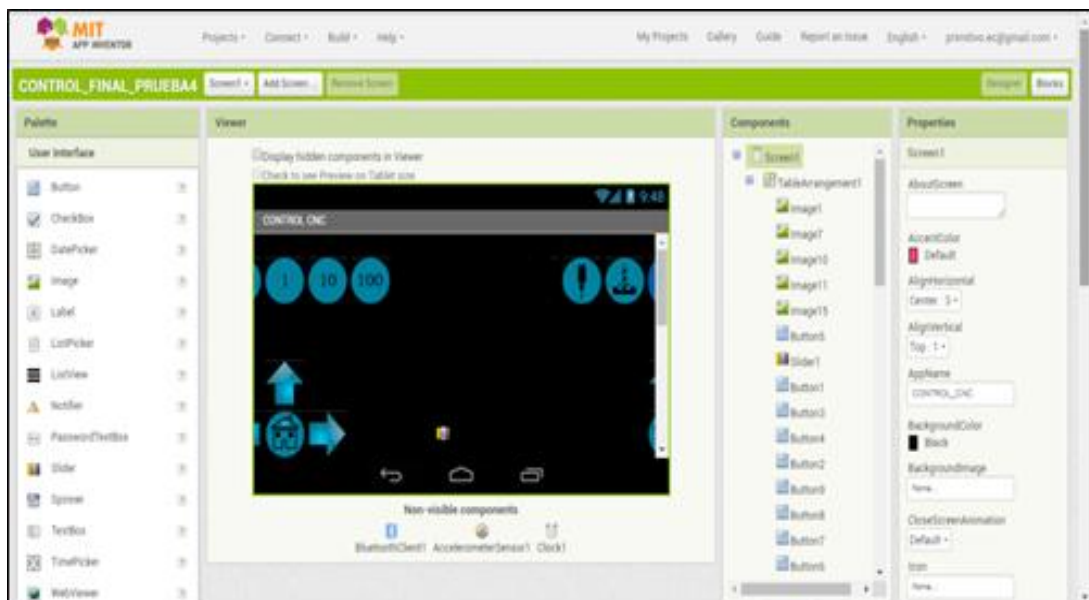
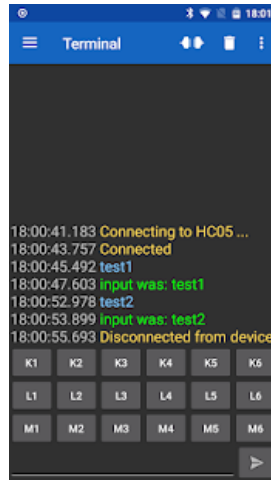


Figura. 1.22. Interfaz del App Inventor 2.

Fuente: MIT App Inventor, (2018).

## 1.10 Serial Bluetooth Terminal

Serial Bluetooth Terminal, es una aplicación que sirve como convertidor de Bluetooth a serial para dispositivos Android, esto facilita la comunicación entre el terminal y la consola de programación de un microcontrolador. Este software (aplicación) servirá para realizar pruebas de comunicación mientras se crea la aplicación del control de desplazamiento direccional para la fresadora. En la figura 1.23 se puede apreciar la interfaz del Serial Bluetooth Terminal.

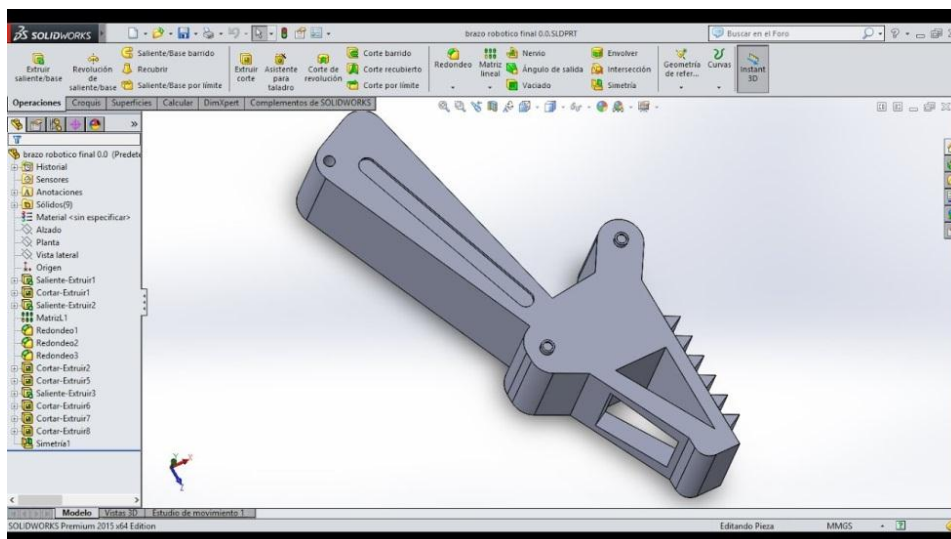


**Figura. 1.23. Interfaz del Serial Bluetooth Terminal.**

Fuente: Serial Bluetooth Terminal, (2018).

## 1.11 Solidworks

Solidworks es un software de diseño 3D CAD (diseño asistido por computadora) para modelar piezas y ensamblajes en 3D y planos en 2D. Es un software que ofrece una gran variedad de soluciones para el desarrollo de productos. Solidworks será empleado para el diseño y creación de los acoples necesarios para utilizar la infraestructura de la impresora 3D como una CNC de corte o grabado. En la figura 1.24 se puede observar la interfaz de trabajo de Solidworks.



**Figura. 1.24. Interfaz de trabajo en Solidworks.**

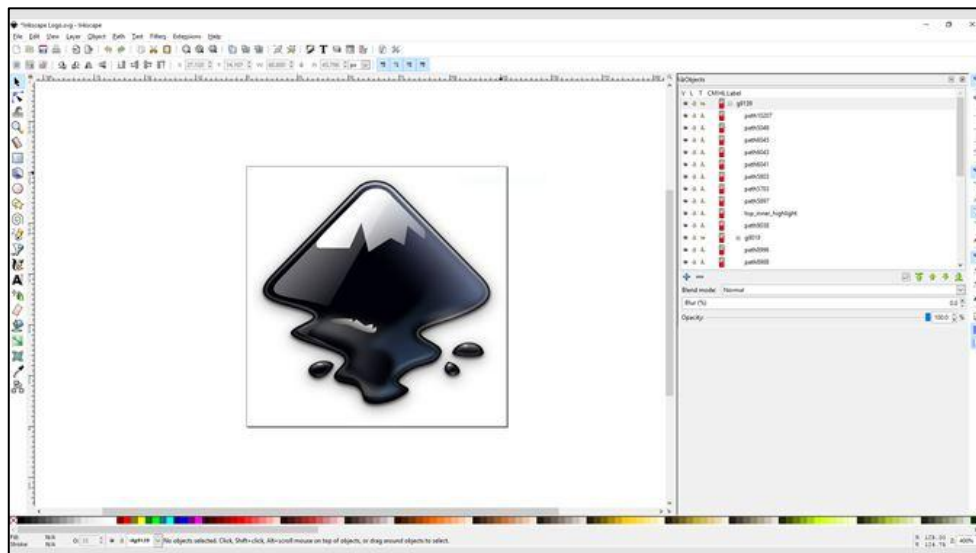
Fuente: Solidworks, (2018).



## 1.12 Inkscape

inkscape.org: Inkscape es un editor de gráficos vectoriales de código abierto, similar a programas como Adobe Illustrator, Corel Draw, Freehand o Xara X.

El desarrollo de un diseño comienza con un garabato en un papel, un boceto de tus ideas, una foto de un objeto memorable o una maqueta virtual que no llegaría a funcionar para acabar un proyecto. Inkscape te ayuda, desde esta fase, a conseguir un diseño de calidad profesional, preparado para ser publicado en la web o en un medio impreso. En la figura 1.25 se muestra la interfaz de trabajo de Inkscape.



**Figura. 1.25. Interfaz de trabajo en Inkscape.**

Fuente: Inkscape, (2018).

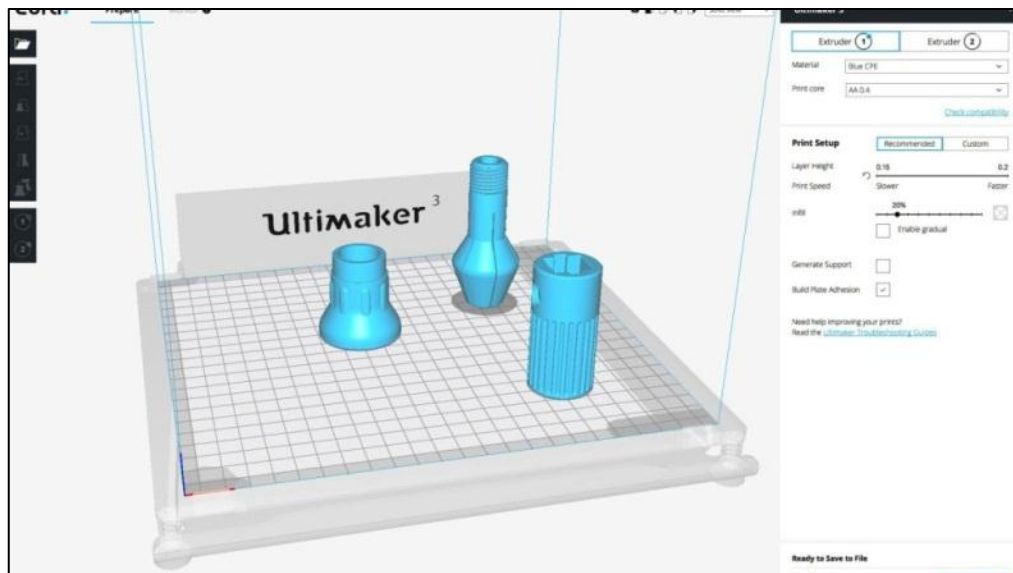
## 1.13 Ultimaker Cura

Cura Connect, facilita al equipo la administración de impresoras 3D en la oficina o estudio. Algunas de las ventajas que ofrece Ultimaker Cura son las siguientes:

- Maximiza el tiempo de trabajo: Los trabajos de impresión se pueden iniciar automáticamente, además de llevar un monitoreo de la impresión y cuenta con un programa de mantenimiento para la impresora 3D.

- Optimiza el flujo de trabajo: Se pueden crear grupos de impresoras 3D, haciendo colas de trabajo de impresión.
- Incrementa operaciones: Cura Connect puede funcionar para una o varias impresoras 3D, por lo que es flexible para necesidades cambiantes.
- Configuración flexible: no se necesitan licencias de servidor o de usuario: se puede ejecutar un proceso de impresión con Ultimaker aprovechando una conectividad de red junto con el software gratuito Ultimaker Cura.

En la figura 1.26 se puede observar la interfaz de Ultimaker Cura.

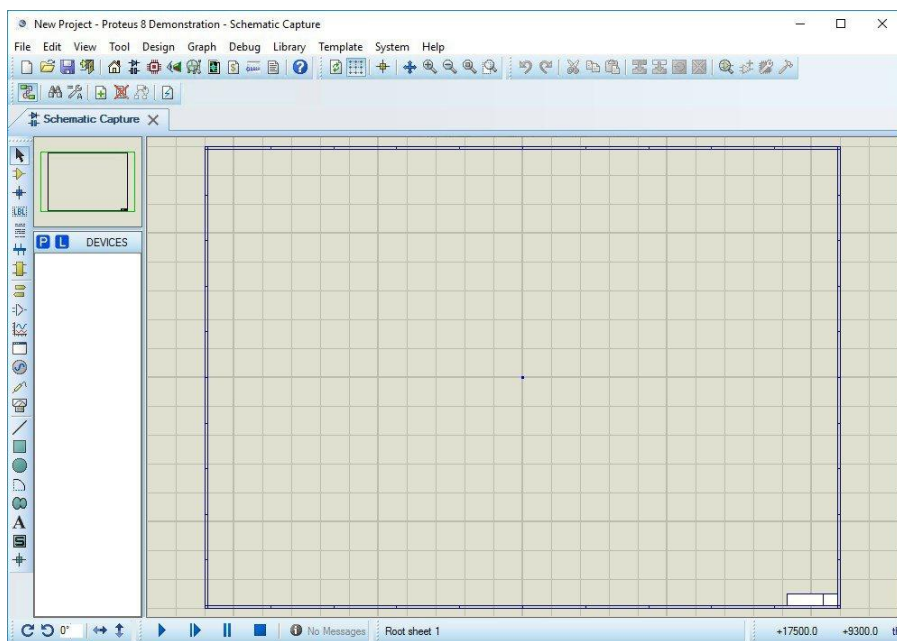


**Figura. 1.26. Interfaz de Ultimaker Cura.**

Fuente: Ultimaker Cura, (2018).

## 1.14 Proteus Design Suite 8

Proteus combina la facilidad de uso con potentes funciones para ayudarlo a diseñar, probar y diseñar PCB profesionales como nunca antes. Con cerca de 800 variantes de microcontroladores listas para la simulación directamente desde el esquema, Proteus Design Suite 8 ofrece el paquete de software completo para los ingenieros de hoy y mañana. En la figura 1.27 se puede observar su interfaz.



**Figura. 1.27. Proteus Design Suite 8.**

Fuente: Proteus Design Suite 8, (2018).

## **CAPÍTULO 2**

### **MARCO METODOLÓGICO**

En este capítulo se trazarán los aspectos metodológicos y prácticos que se utilizarán para desarrollar un prototipo multipropósito fresadora CNC e impresora volumétrica. Por otra parte en este capítulo se describirán los tipos de investigación, las técnicas e instrumentos de recaudación de datos, así como la metodología seleccionada (fases del proyecto).

#### **2.1 Tipo y diseño de investigación**

De acuerdo con los criterios expuestos por, Palella y Martins (2012, p.94), una investigación es aquel proceso que conlleva al diseño o creación de un prototipo para cubrir una necesidad y todo esto basado en conocimientos previamente adquiridos. De allí que la investigación proyectiva se refiere a un proceso que conlleva a un resultado tangible, que puede ser un modelo teórico o práctico. El investigador puede llegar a crear un proyecto mediante vías diferentes, las cuales involucran procesos, enfoques, métodos y técnicas propias.

Por lo expuesto anteriormente, se puede decir que la investigación que se empleará para el desarrollo del presente proyecto de titulación es de tipo proyectiva, debido a que, gracias a estudios previos, se propone el desarrollo de un prototipo multipropósito fresadora CNC e impresora volumétrica, buscando proporcionar un aporte de gran importancia al desarrollo tecnológico, el cual involucra procesos, enfoques, métodos y técnicas propias.

Por otro lado, según Ackerman y Com (2013, p.38). La investigación descriptiva es un tipo de investigación que se realiza cuando ya se avanzó, aunque sea un poco en el tratamiento de un problema, por lo cual, se pueden establecer relaciones o vínculos entre los elementos que se ponen en juego. Por ende los trabajos descriptivos realizan diagnósticos respecto a un tema en particular, observando y recolectando información en torno a la problemática central.

Por lo antes mencionado, la investigación a ser aplicada en este proyecto también se considerará como descriptiva, ya que procura observar, dar detalles y recolectar información relacionada con la problemática que se busca resolver al realizar un prototipo multipropósito fresadora CNC e impresora volumétrica.

## **2.2 Técnicas e instrumentos de recolección de datos.**

La recolección de datos es una parte muy significativa en este proceso de investigación ya que, se utilizarán técnicas e instrumentos para buscar información que esté relacionada con el problema planteado. Como es de conocimiento general toda investigación está basada en datos e información recolectada a lo largo de la elaboración de sí misma. Gracias a la documentación el investigador se encuentra facultado para manipular cada uno de los procesos relacionados con su proyecto de investigación, lo cual es indispensable para la ejecución objetiva de su proyecto y de esta manera también es un respaldo para la elaboración de su propuesta, conclusiones, recomendaciones, entre otros aspectos que involucran a su investigación.

Para esta investigación se utilizará la técnica de la observación científica, la cual es definida por Yuni y Urbano (2014), como una técnica para la recolección de datos consistente en el reconocimiento y estudio de los acontecimientos tal como suceden en la realidad social a través del empleo de los sentidos con o sin ayuda de soportes tecnológicos, conforme a los requerimientos de la investigación científica y a partir de las categorías perceptivas construidas a partir y por las teorías científicas que utiliza el investigador. Dicha técnica será aplicada para recolectar datos e información gracias a la observación de manuales técnicos, herramientas electrónicas, software, etc. Así también se

utilizará la lectura de textos informativos provenientes de fuentes veraces, los cuales deberán contener datos útiles e interesantes para el desarrollo del presente proyecto de titulación.

### **2.3 Metodología seleccionada**

Para la realización de este proyecto de titulación se ejecutarán una serie de actividades, las cuales están basadas en la metodología planteada por el autor Córdoba (2018) ya que existe una gran concordancia en su mayoría de los objetivos de este proyecto con las fases descritas por dicho autor, estas son 8 y se presentan a continuación:

#### **Fase I: Observar**

Observar también es el origen de la investigación científica. Un sujeto que no observa, que no posee una estructura intelectual apropiada o que carece de conocimientos en el asunto, más que mirar impone interpretaciones de la realidad o hace suyos los puntos de vista de otros, por lo que serán vagas sus apreciaciones. En este sentido se procede a identificar las posibles soluciones a nivel físico, electrónico y de software para solventar el problema que engloba el propósito de la realización del proyecto de titulación.

#### **Fase II: Determinar**

Una investigación científica es un conjunto de decisiones, procedimientos encaminados a la construcción de un determinado tipo de información sobre algunos fenómenos que al tener la posibilidad de convertirse en conocimiento nos permita actuar.

Y es así que para solucionar los requerimientos se utilizará software como: Solidworks y Ultimate Cura para el ámbito de diseño y creación de piezas, por otra parte software como Arduino y Processing se usarán para las programaciones requeridas. Así mismo los requerimientos en el ámbito electrónico serán saldados con la implementación de circuitos de alimentación y control.

## **Fase II: Reflexionar**

La fase es continuamente llamada de incubación, lo que tiene una explicación debido a que pertenece tanto al ámbito de la creatividad como al de la innovación. Es un momento clave para generar nuevas ideas relativas a la reformulación, comprensión y/o solución del problema. En esta fase se pudo deliberar sobre la utilidad de las herramientas a ser empleadas y la necesidad de otras, así como la adquisición de ciertos conocimientos técnicos y teóricos necesarios. Además se puso a consideración el cronograma de actividades propuesto en el “Plan del proyecto integrador de carrera” que se realizó con anterioridad, analizando así la cantidad de tiempo disponible para realizar el proyecto de titulación.

## **Fase IV: Documentar**

La acción de documentar es una manera de investigar que da el carácter de especialización a las actividades, reuniendo los puntos de vista de autores calificados en el tema y de las personas involucradas o con experiencia en el asunto. Este fue un ámbito crucial, ya que, es sumamente necesario realizar una investigación científica, teórica y técnica con la cual respaldar la implementación del proyecto de titulación.

## **Fase V: Diseñar**

La experimentación es el recurso más importante y avanzado que ha ideado la ciencia para corroborar una posible verdad. Para la realización del proyecto, es necesario diseñar el sistema de fresado y desarrollar el software requerido y estos son procesos que se hacen experimentando de diferentes formas, probando diferentes configuraciones o diseños.

## **Fase VI: Valorar**

El proyecto se presenta para ser analizado y enriquecido, no para realizar una feroz defensa. Este es un punto intermedio debido a que se puede detener el proceso de diseño y

valorarlo antes de realizar su completa implementación, de esta forma se pueden ahorrar posibles inconvenientes a futuro.

### **Fase: VII Aplicar**

En la implementación, es importante solucionar eficaz y oportunamente cualquier inconveniente para una acertada ejecución del proyecto. En esta fase, luego del desarrollo y una oportuna valoración previa se puede implementar el proyecto en su totalidad y así poder pasar a una secuencia de pruebas para constatar el correcto funcionamiento del proyecto.

### **Fase: VIII Comunicar**

La redacción es un modo de representar el mundo interior. Es un modo de almacenar información, pero también de comunicar ideas. Es importante poder comunicar los resultados obtenidos luego de todo el proceso de investigación desarrollo e implementación del proyecto. Y es por esto que se realiza la redacción del presente proyecto de titulación, en el cual se pueden dejar todos los conceptos investigados así como las conclusiones y recomendaciones a las que llegó el proyecto.



## **CAPÍTULO 3**

### **PROPUESTA**

El presente proyecto de titulación, busca realizar un aporte en el campo de la impresión 3D y fresado CNC, haciendo posible la utilización de la infraestructura de un equipo para generar las funciones de otro. La infraestructura de la impresora se mantendrá intacta mientras se fabrican elementos fijos y móviles que se acoplen a ella, haciendo posible el proceso de fresado (corte/grabado). En este punto se diseñarán y se fabricarán los siguientes elementos: Un cabezal para la fresadora, una plataforma para corte y grabado, acoples de sujeción para la plataforma, acople para la colocación de una tarjeta con borneras desarmables, soportes para la sujeción de la herramienta giratoria y un elemento móvil para la colocación de un servomotor.

#### **3.1 Aspectos técnicos requeridos para la fresadora CNC**

A nivel electrónico, se diseñará e implementará una tarjeta electrónica la cual se encargará del control de una herramienta giratoria. Dicha tarjeta electrónica albergará un Atmega 328, reguladores de voltaje LM317, transistores TIP122 y botones para el control de velocidad de la herramienta giratoria. Por otra parte el desarrollo e implementación de software para el prototipo multipropósito fresadora CNC e impresora volumétrica se realizará utilizando las siguientes herramientas: (IDE) Arduino, Processing, Solidworks, MIT App Inventor 2, Serial Bluetooth Terminal, Solidworks, Inkscape, Ultimaker Cura, Proteus Design Suite 8.

Proteus Design Suite 8, será el software empleado para el diseño de la placa electrónica encargada del proceso de control de las herramientas para el corte o grabado CNC.

(IDE) Arduino, será utilizado para realizar la programación del Atmega 328 y de esta forma poder controlar: herramienta giratoria (encendido, apagado y velocidad).

Processing, será empleado para el diseño, programación e implementación de una interfaz dotada de controlar: movimientos de los ejes X,Y,Z, retorno a posiciones “home” en los ejes (X,Y) y (Z), control de una herramienta giratoria (encendido, apagado y velocidad), cargar archivo de (corte/grabado), reinterpretar código G, ejecutar acción de (corte/grabado), visualización de la posición de los ejes X, Y, Z, escribir y enviar código G bajo el firmware Marlin.

Serial Bluetooth Terminal, es un software que será el encargado de realizar pruebas de comunicación vía Bluetooth entre una PC y un smartphone, proceso previo al diseño de una aplicación.

MIT App Inventor 2, Será utilizado para la creación de una aplicación compatible con el sistema operativo Android, encargada de ejecutar controles como: movimientos de los ejes X,Y,Z, retorno a posiciones “home” en los ejes (X,Y) y (Z). Como un valor agregado además será posible: controlar la herramienta giratoria (encendido, apagado y velocidad), cargar una imagen desde el repositorio del smartphone, dibujar sobre dicha imagen, escribir y enviar código G bajo el firmware Marlin.

Solidworks, será usado para el diseño de elementos fijos y móviles útiles para posibilitar el proceso de (corte/grabado) con una fresadora, utilizando de esta forma la infraestructura de una impresora 3D.

Ultimaker Cura, será el encargado de transformar los archivos CAD generados por Solidworks, y de esta manera hacer realidad los diseños generados en dicho software.

Inkscape, será el software empleado para obtener el código G, el cual será la base para la reinterpretación de código a ser utilizado por la fresadora CNC (cortadora/grabadora).

### **3.2 Ventajas de un prototipo multipropósito**

Las ventajas que ofrece la realización de un prototipo multipropósito fresadora CNC e impresora volumétrica, se traducen como beneficios para la industria y el conocimiento técnico y científico, de aquí se desprenden algunos beneficiarios directos e indirectos.

Beneficiarios directos: la pequeña empresa es un beneficiario directo, ya que, en la actualidad existen muchos emprendedores que necesitan de herramientas más económicas para impulsar sus negocios, ésta es una excelente opción al poder adquirir un impresora 3D para la generación de figuras en tres dimensiones y además a bajo costo poder obtener las funciones de una fresadora CNC para materiales de baja densidad.

Beneficiarios indirectos: quienes se benefician indirectamente con la propuesta para realizar el presente proyecto de titulación, son los pequeños y grandes vendedores de dispositivos electrónicos y herramientas eléctricas respectivamente. Otros beneficiarios son los estudiantes de Ingeniería, que pueden acudir al repositorio digital de la Universidad Tecnológica Israel y tener acceso a este documento para obtener información acerca del tema y realizar nuevas aportaciones al ámbito tecnológico, social y económico.

### **3.3 Estudio técnico para el diseño de hardware**

El estudio técnico sirve para obtener una aproximación de todo aquello que será necesario a nivel de tecnología para realizar la implementación del presente proyecto de titulación. Por ejemplo, es necesario saber que herramientas, dispositivos, software y componentes electrónicos serán necesarios, así mismo, es preciso conocer detalles técnicos de cada uno de ellos, que en lo posterior servirán para el diseño o selección de materiales. A continuación se muestran los cálculos y consideraciones necesarias para la selección de materiales.

#### **3.3.1 Atmega 328**

Se seleccionó un Atmega 328 por su compatibilidad de trabajo con la (IDE) de Arduino, esto luego de realizar un procedimiento de migración de su secuencia de inicio

hacia el microcontrolador, esto permitirá emplear al Atmega 328 fuera de la placa de Arduino. El microcontrolador opera en un rango de voltaje entre: 1.8 - 5.5V, así mismo genera un consumo de corriente equivalente a: 0,2 mA, estos datos son importantes ya que se debe calcular un regulador de voltaje para el microcontrolador.

### 3.3.2 Regulación de voltaje para el microcontrolador

El regulador de voltaje a ser utilizado es un LM317 ya que cuenta con un rango de salida entre 1.25 V a 37 V, entregando un máximo de corriente de: 1,5 A. Cálculos para generar una salida de voltaje de 5,5 V para alimentar al microcontrolador. Ya que se utilizará una fuente de voltaje externa DC de 12 V a 3A tenemos lo siguiente:

Para la obtención de un voltaje de salida de 5,5V y tomando una R1 como referencia de 240Ω, se calcula el valor de R2. (Figura. 3.1.).

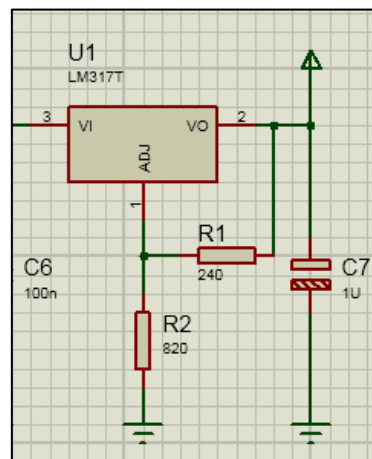


Figura. 3.1. Regulación de voltaje para el microcontrolador.

Fuente: Proteus Design Suite 8, (2018).

$$V_{out} = 1.25V \left( 1 + \frac{R2}{R1} \right) \quad (\text{Ecuación 1.1})$$

$$R2 = R1 \left( \frac{V_{out}}{1.25V} - 1 \right)$$

$$R2 = 240\Omega \left( \frac{5.5V}{1.25V} - 1 \right)$$

$$R2 = 816\Omega$$

### 3.3.3 Servomotor MG995

Se eligió al MG995 por la capacidad de su torque que equivale a 8.5 kgf·cm (4.8 V), 10 kgf·cm (6 V), a un consumo de 1500mA. Esto es necesario ya que se requerirá al servomotor para empujar o jalar un potenciómetro lineal en la herramienta giratoria para controlar su velocidad.

### 3.3.4 Regulación de voltaje para el servomotor

El regulador de voltaje a ser utilizado será un LM317 ya que cuenta con un rango de salida entre 1.25 V a 37 V, entregando un máximo de corriente de: 1,5 A. Los cálculos generados anteriormente servirán para ser aplicados en este regulador ya que los rangos de voltaje requeridos para el servomotor son similares. Ahora bien ya que el consumo de corriente del servomotor es de 1500 mA y para evitar futuros percances por falta de abastecimiento de corriente por parte del regulador, se utilizará una configuración recomendada por el fabricante para obtener a la salida del regulador una corriente de hasta 4A. (Figura. 3.2).

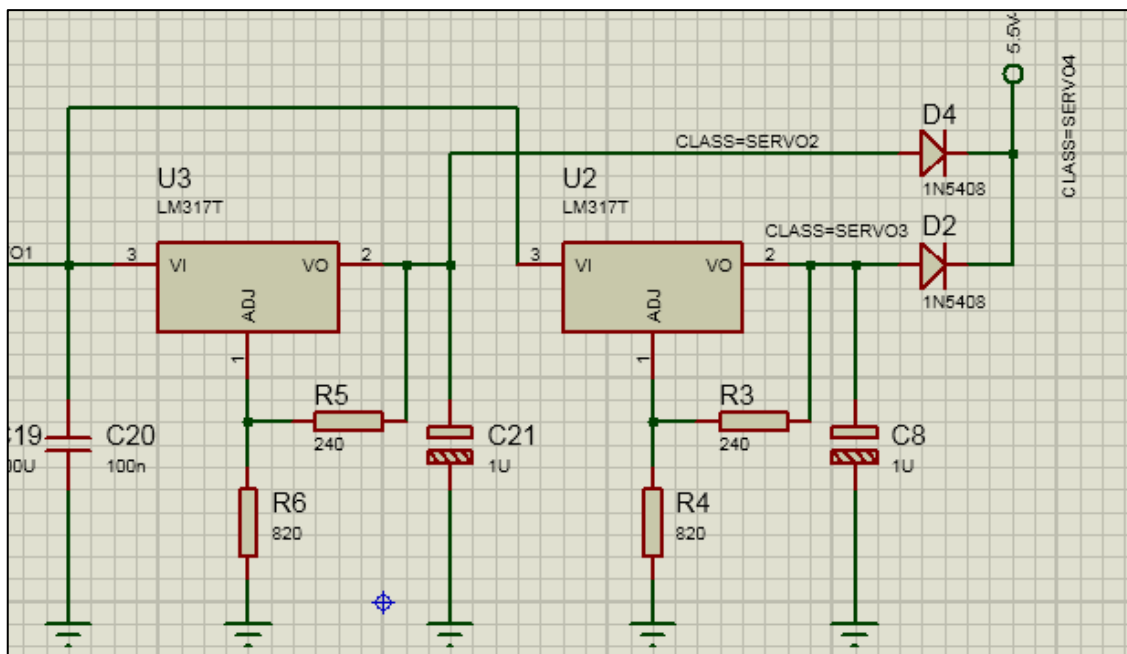


Figura. 3.2. Regulación de voltaje para el servomotor.

Fuente: Proteus Design Suite 8, (2018).

### 3.3.5 Control de relés con el transistor TIP122

El TIP122 será empleado como un switch para control de la (herramienta eléctrica giratoria), los cálculos son los siguientes. Ya que serán utilizadas dos de las salidas del microcontrolador, las cuales entregarán un valor máximo de 5 V en nivel alto y el voltaje de  $VCC = 12\text{ V}$  (fuente externa) a una corriente de consumo máxima de 1,5 A (Relés) se puede calcular:  $I_b$  del transistor con un factor de  $h_{fe} = 1000$  estipulado por el datasheet del TIP122 lo cual permitirá calcular su  $R_b$ . (Figura. 3.3). Se usan dos salidas del microcontrolador y dos relés en el caso de ser necesaria una segunda herramienta eléctrica.

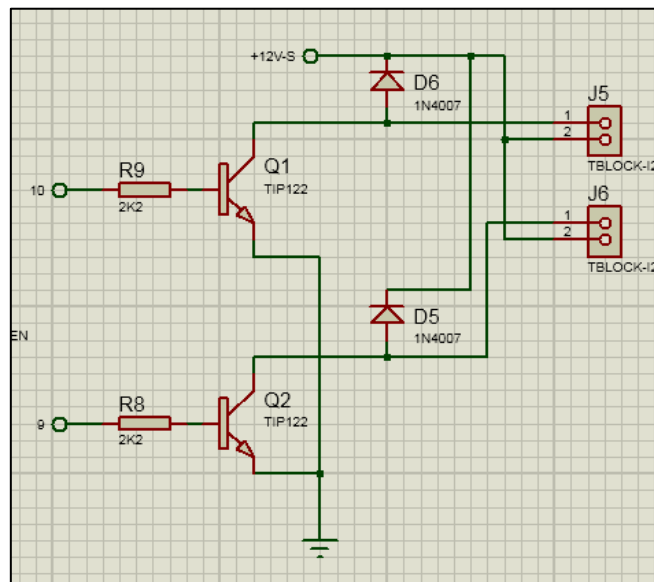


Figura. 3.3. Control de relés con el Transistor TIP122.

Fuente: Proteus Design Suite 8, (2018).

$$I_B \max \cong \frac{I_c \text{ sat}}{\beta \text{ cd}} \quad (\text{Ecuación 1.2})$$

$$I_B \text{ máx} = \frac{1,5\text{ A}}{1000}$$

$$I_B \text{ máx} = 1,5\text{ mA}$$

$$I_B = \frac{v_i - 0,7\text{ v}}{R_B} \quad (\text{Ecuación 1.3})$$

$$Rb = \frac{5v - 0,7v}{1,5mA}$$

$$Rb = 2866,6 \Omega$$

$$Rb = 2,8 k\Omega$$

$$Rb = 2,2 k\Omega$$

### 3.4 Análisis de costos

Una vez realizado el estudio técnico, se procede a un análisis del costo que representa cada uno de los recursos necesarios que hagan posible la implementación de un prototipo multipropósito fresadora CNC e impresora volumétrica. A continuación se presenta la cantidad económica necesaria para la ejecución del proyecto. En la tabla 3.1 se puede observar el costo de los componentes electrónicos, así mismo en la tabla 3.2 se representa el costo de las herramientas eléctricas y materiales varios, además en la tabla 3.3 se analizan los costos a nivel de software, por otra parte en la tabla 3.4 se representa el costo de la fabricación de la tarjeta electrónica y por último en la tabla 3.5 se analiza el costo de la mano de obra.

**Tabla 3.1 Componentes electrónicos.**

	Descripción	Cantidad	Valor	Total
1	Convertidor USB 2.0	1	4.02	\$4.02
2	LM317	3	0.67	\$2.01
3	TIP122-NPN-	2	0,45	\$0,89
4	Resistencia 1K 1/4	12	0,02	\$0,21
5	Porta fusible	2	0,40	\$0,80
6	CRY 16Mhz	1	0,49	\$0,49
7	Atmega 328	1	6,25	\$6,25
8	Zoc 28p	1	0,25	\$0,25

<b>9</b>	Capacitor 22pf / 50v	4	0,07	\$0,28
<b>10</b>	Capacitor 22nf / 50v	2	0,07	\$0,14
<b>11</b>	Led 5 na	1	0,09	\$0,09
<b>12</b>	Capacitor 1000uf / 35v	9	0,31	\$2,81
<b>13</b>	Servo 995	1	14,29	\$14,29
<b>14</b>	Fus 1A peq	2	0,11	\$0,21
<b>15</b>	Con-Polhdr 5P	1	0,40	\$0,40
<b>16</b>	Diodo bilicon	6	0,16	\$0,96
<b>17</b>	Swith mini pulsar	6	0,25	\$1,50
<b>18</b>	Bornera verde x3	13	1	\$13,00
			Subtotal	\$48,60
			Iva 12%	\$5,83
			Total a facturar	\$54,43

Fuente: Elaborado por el autor.

**Tabla 3.2 Herramientas eléctricas y materiales varios.**

	Descripción	Cantidad	Valor	Total
<b>1</b>	Dremel 3000	1	90	\$90
<b>2</b>	Flex Dremel	1	35	\$35
<b>3</b>	Filamento Pla 1.75mm 1 Kg	1	40	\$40
<b>4</b>	Plataforma de corte y grabado	1	25	\$25
			Subtotal	\$190
			Iva 12%	\$22.8
			Total a facturar	\$212.8

Fuente: Elaborado por el autor.

**Tabla 3.3 Software.**

	Descripción	Cantidad	Valor	Total
<b>1</b>	(IDE) Arduino	1	0	\$0
<b>2</b>	Processing	1	0	\$0



<b>3</b>	Solidworks	1	0	\$0
<b>4</b>	MIT App Inventor 2	1	0	\$0
<b>5</b>	Serial Bluetooth Terminal	1	0	\$0
<b>6</b>	Inkscape	1	0	\$0
<b>7</b>	Ultimaker Cura	1	0	\$0
<b>8</b>	Proteus Design Suite 8	1	0	\$0
			Subtotal	\$0
			Iva 12%	\$0
			Total a facturar	\$0

Fuente: Elaborado por el autor.

**Tabla 3.4 Tarjeta electrónica.**

	Descripción	Cantidad	Valor	Total
<b>1</b>	Percloruro en fun	2	0,71	\$1,43
<b>2</b>	Impresión laser	1	1	\$1
<b>3</b>	Baquelita	1	1,25	\$1,25
<b>4</b>	Corte	1	1	\$1
<b>5</b>	Perforación	1	1	\$1
<b>6</b>	Papel de impresión	1	1	\$1
			Subtotal	\$6,68
			Iva 12%	\$0,80
			Total a facturar	\$7,48

Fuente: Elaborado por el autor.

**Tabla 3.5 Mano de obra.**

	Descripción	Horas	Valor	Total
<b>1</b>	Diseño y simulación	250	4,50	\$1125
<b>2</b>	Implementación	100	4,50	\$450
			Subtotal	\$1575

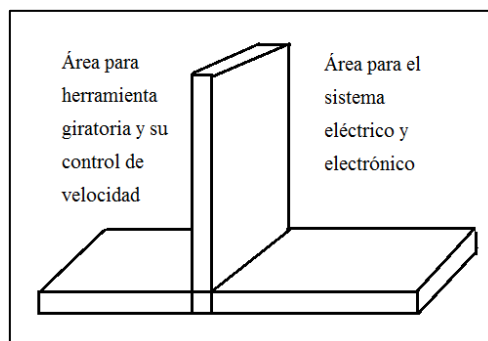
	Iva 12%	\$189
	Total a facturar	\$1764

Fuente: Elaborado por el autor.

Como se puede apreciar en la distribución de los cuadros, se especifica la cantidad monetaria que representa la adquisición de los componentes electrónicos \$54,43, por otra parte el costo de las herramientas eléctricas, así como el material para la impresión de los elementos fijos y móviles para la CNC, entre otros es de \$212.8. Así mismo el costo que representa el empleo de software es de \$0. Además el costo de la fabricación de la tarjeta electrónica bordea los \$7,48 y por último se representa el costo que genera el tiempo utilizado por el estudiante para realizar este proyecto.

### 3.5 Módulo que alojará el sistema

Deberá existir un módulo que albergue a la herramienta giratoria, así como al servomotor encargado de la manipulación de la velocidad del Dremel, dicho modulo además deberá alojar todo el sistema eléctrico y electrónico encargado del control de la fresadora CNC, por otra parte también será necesario el diseño de acoples que faciliten el intercambio de funcionalidades entre la impresora volumétrica y la fresadora CNC. Un diseño tentativo del módulo se presenta en la figura 3.4, el cual deberá contar con dos zonas o áreas, una para contener a la herramienta giratoria y su control de velocidad y otra para almacenar el sistema eléctrico y electrónico.



**Figura. 3.4. Diseño tentativo del módulo.**

Fuente: Elaborado por el autor.

### 3.6 Conclusión

Gracias al análisis técnico y económico se puede hacer una comparación entre el precio de fabricación y adquisición de una CNC fresadora (grabadora / cortadora). Producto de esta comparación se puede determinar que el proyecto es factible ya que por un precio de \$212.8 se puede desarrollar una fresadora de corte y grabado CNC a partir de una impresora volumétrica para adquirir un equipo multipropósito. El costo estipulado por mano de obra para esta implementación no existe debido a que se trata de un proyecto de titulación, pero es importante saber la cantidad de tiempo que tomó el realizar este sistema. Así mismo el costo a nivel de software no existe porque se utilizó software libre o una versión para estudiantes.

La factibilidad del proyecto también se establece debido a que el estudiante que propuso y efectuó el presente proyecto de titulación cuenta con los medios económicos así como los conocimientos técnicos y teóricos necesarios para ser aplicados y hacer realidad este proyecto. De igual forma para la implementación del proyecto se debió contar previamente con una impresora 3D a la cual adecuar el nuevo sistema.

## **CAPÍTULO 4**

### **IMPLEMENTACIÓN**

Para la implementación de un prototipo multipropósito fresadora CNC e impresora volumétrica, a continuación se describe el proceso de construcción del producto a nivel de (hardware y software) así como sus componentes fijos y móviles. También, se describe la puesta en marcha del sistema desarrollado. Así, posteriormente se realizarán pruebas de funcionamiento del equipo y finalmente se analizarán los resultados.

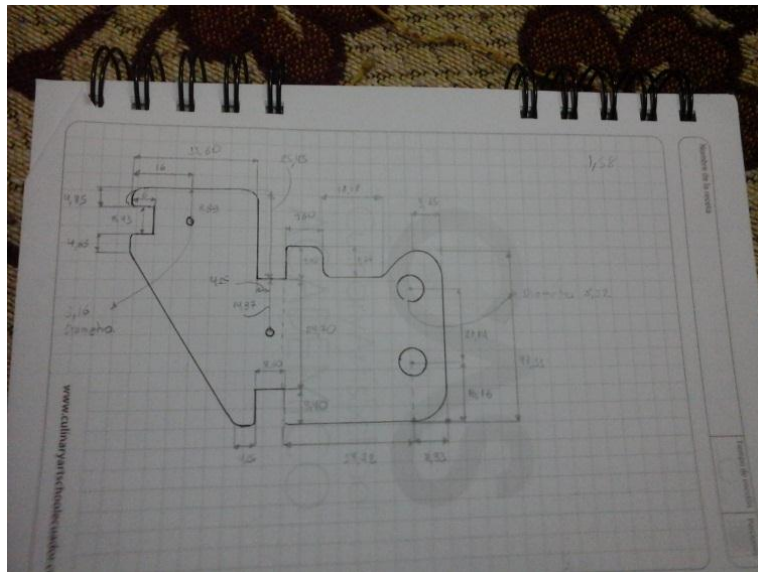
#### **4.1 Desarrollo**

##### **4.1.1 Diseño de elementos fijos y móviles**

Para el esquema de los elementos fijos y móviles se empleó el software Solidworks, en él se realizaron los diseños para acoplar la fresadora CNC en la infraestructura de la impresora volumétrica. Previamente para diseñar en Solidworks fue necesario tomar medidas a los diferentes objetos que necesitaban ser modificados (figura 4.1). Por ejemplo se realizaron los siguientes diseños: El cabezal se dimensionó en dos partes, un lado derecho y uno izquierdo los cuales aprisionan en conjunto al flex del Dremel manteniéndolo así estable y listo para ser colocado en la base donde se encontraba anteriormente el cabezal de extrusión (figuras 4.2 – 4.3). Por otra parte fue necesaria la construcción de una base para la colocación de las borneras que ayudarán en el proceso de desmontaje del cabezal de impresión (figura 4.4). Además, fue necesario enviar a fabricar una nueva cama de impresión 3D y una plataforma de (corte/grabado), ya que, estas dos deben acoplarse mutuamente para intercambiar sus funciones (figura 4.5 – 4.6). Así mismo, fue necesario diseñar acoples para montar la plataforma de (corte/grabado) sobre la nueva cama de impresión (figura 4.7). Adicionalmente, se diseñó un soporte fijo / móvil para el servomotor (figura 4.8). Además se realizó el diseño de una

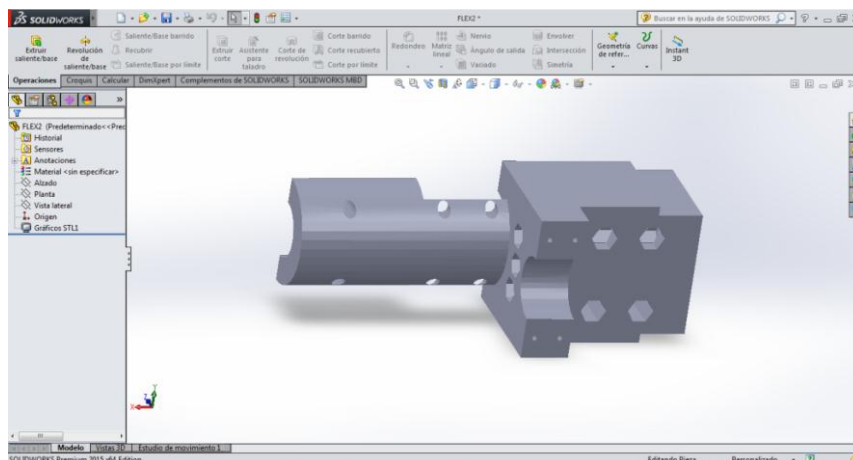
perilla la cual será reemplazada dentro del Dremel para su posterior manipulación con el servomotor (figura 4.9). Y finalmente se esquematizaron soportes para el Dremel tanto para su parte superior como inferior (figura 4.10 – 4.11).

Para el montaje de todos estos elementos sobre la impresora 3D y en el módulo de corte/grabado revisar el ANEXO 2. Además para observar el estado de la infraestructura de la impresora 3D antes y después de la colocación de los acoples dirigirse al ANEXO 3.



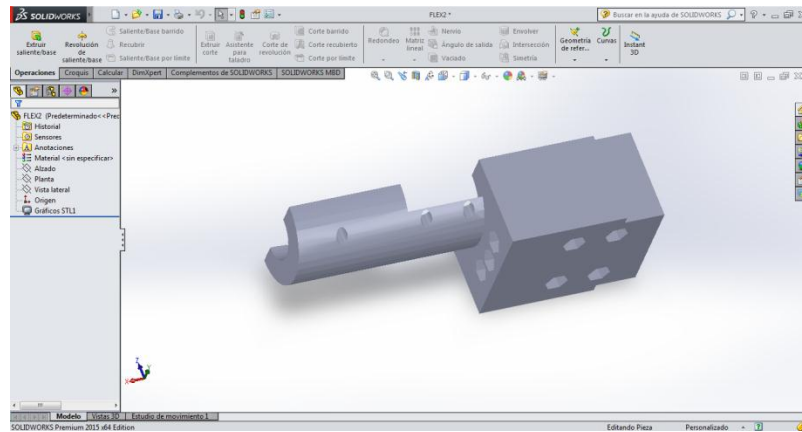
**Figura. 4.1. Dimensionamiento de figuras.**

Fuente: Elaborado por el autor.



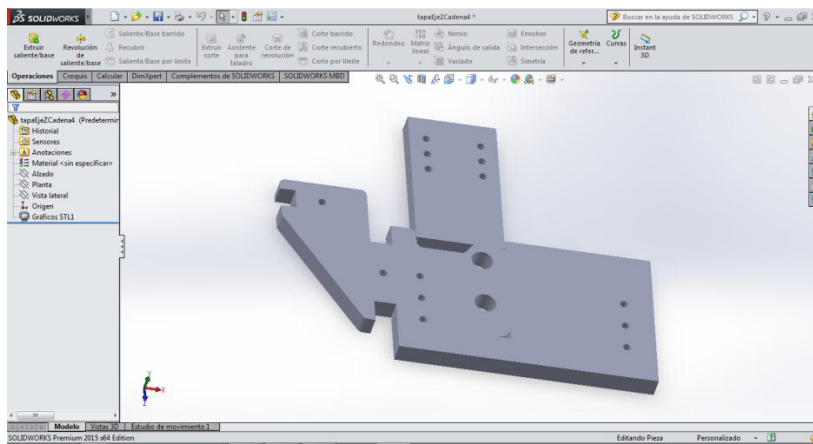
**Figura. 4.2. Cabezal de fresadora (lado izquierdo).**

Fuente: Solidworks, (2018).



**Figura. 4.3. Cabezal de fresadora (lado derecho).**

Fuente: Solidworks, (2018).



**Figura. 4.4. Acople para borneras.**

Fuente: Solidworks, (2018).



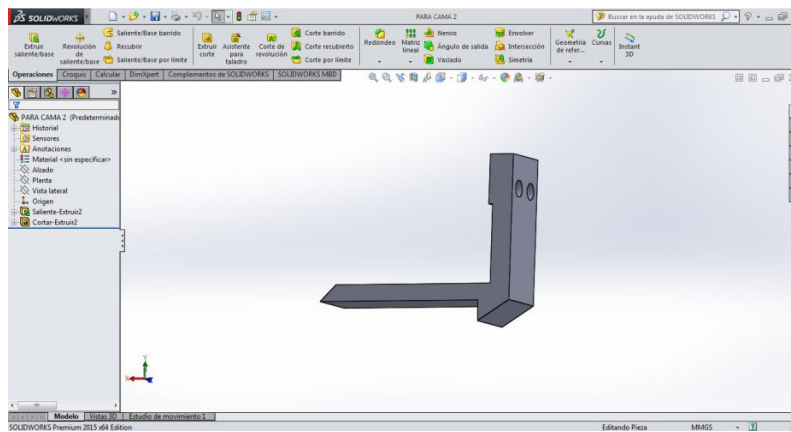
**Figura. 4.5. Nueva plataforma de impresión.**

Fuente: Elaborado por el autor.



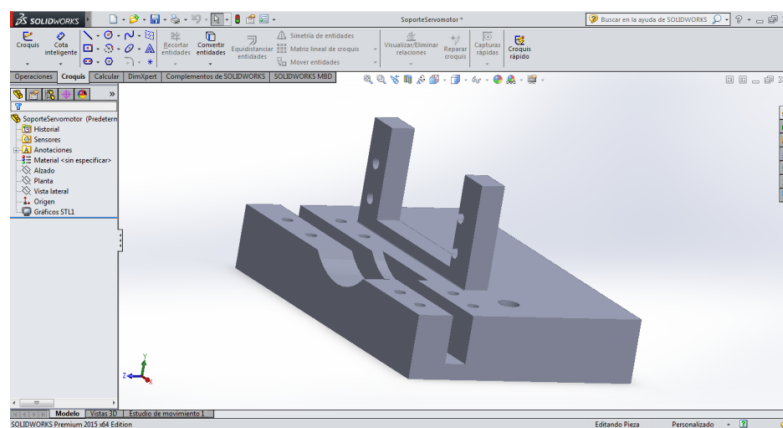
**Figura. 4.6. Plataforma de (corte/grabado).**

Fuente: Elaborado por el autor.



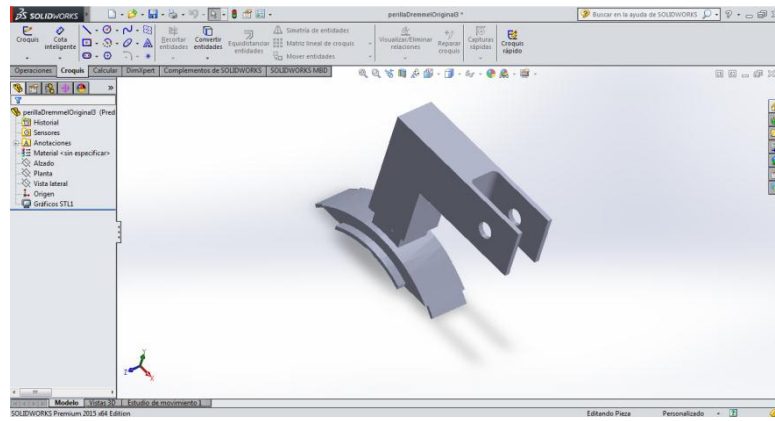
**Figura. 4.7. Acople para el montaje de plataforma (corte/grabado).**

Fuente: Solidworks, (2018).



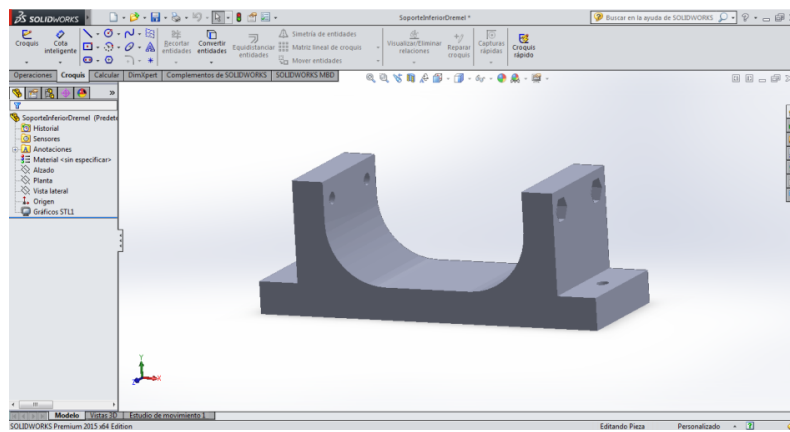
**Figura. 4.8. Plataforma para servomotor.**

Fuente: Solidworks, (2018).



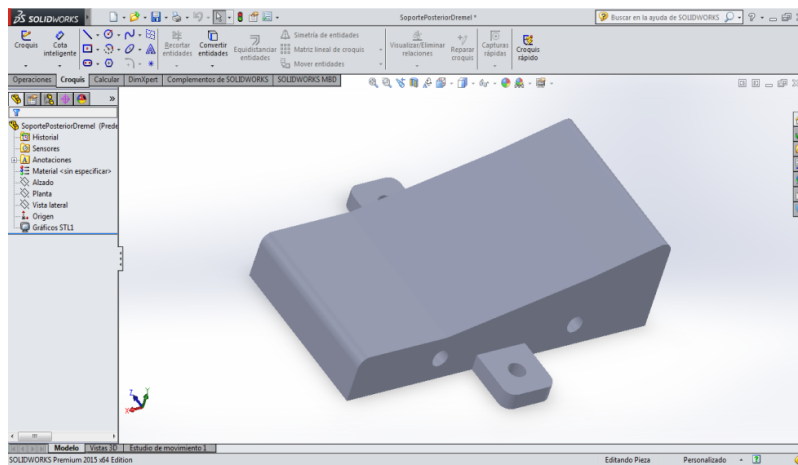
**Figura. 4.9. Perilla para Dremel.**

Fuente: Solidworks, (2018).



**Figura. 4.10. Soporte superior para Dremel.**

Fuente: Solidworks, (2018).

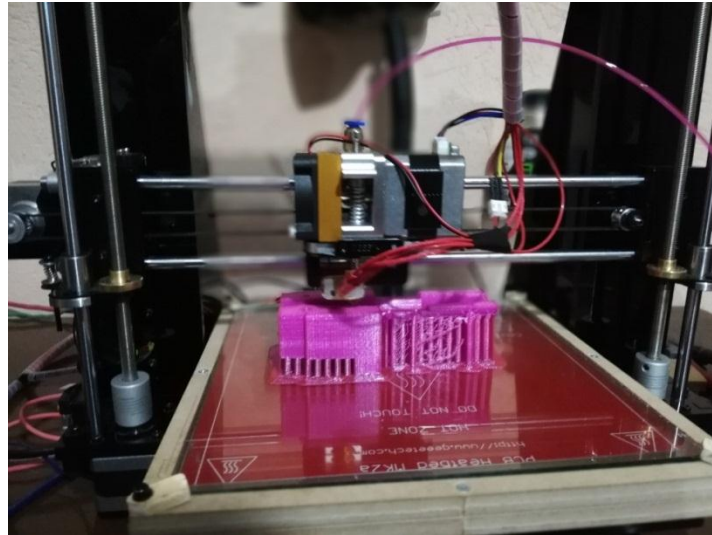


**Figura. 4.11. Soporte inferior para Dremel.**

Fuente: Solidworks, (2018).



Luego del diseño de las piezas en Solidworks, se las fabricó utilizando la impresora 3D, la cual será utilizada en este proyecto de titulación, este proceso se efectuó bajo el software Ultimaker Cura. En la figura 4.12 se puede apreciar el proceso de impresión.



**Figura. 4.12. Impresión de piezas de acople.**

Fuente: Elaborado por el autor.

#### **4.1.2 Diseño de tarjeta electrónica**

Por medio de la utilización del software, Proteus Design Suite 8 en conjunto con los cálculos realizados en la propuesta y la información adicional recolectada en el capítulo I, se diseñó la tarjeta que alberga al microcontrolador Atmega 328, encargado del control de una herramienta giratoria (encendido, apagado y control de velocidad), Para una apreciación unificada de todos los diagramas electrónicos y eléctricos revisar el ANEXO 4.

#### **Regulación de fuente para el Atmega 328**

A continuación en la figura 4.13, se puede observar el esquema realizado para alimentar al Atmega 328. Los cálculos fueron previamente realizados en la propuesta de este documento y los capacitores recomendados (desacoplo y bulk) en el capítulo I fueron colocados de igual forma. Así mismo, se colocó un fusible y un diodo rectificador a la entrada del circuito para una protección del mismo y de esta manera poder evitar cualquier desperfecto de corriente o de voltaje.

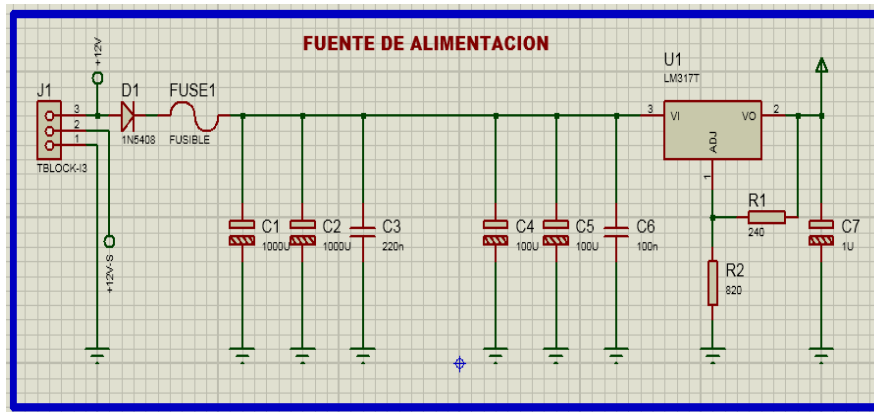


Figura. 4.13. Regulación de fuente para el Atmega 328.

Fuente: Proteus Design Suite 8, (2018).

### Regulación de fuente para el Servomotor MG995

Para la regulación de voltaje que alimentará al servomotor, se utilizaron los mismos cálculos realizados ya que dicho componente se alimentará de una fuente de 5.5 v, la diferencia está en que, como es un dispositivo de fuerza consume mayor cantidad de corriente, por lo cual, se eligió un esquema en el que se mantenga su voltaje de salida intacto, pero con una elevación de corriente a 4A aproximadamente, el mencionado esquema, fue justificado en la propuesta de este documento (capítulo 3). De igual forma que en el esquema anterior se colocó un fusible y un diodo rectificador a la entrada del circuito para evitar posibles daños de existir una sobre corriente o un retorno de voltaje no deseado. En la figura 4,14 se muestra el circuito regulador.

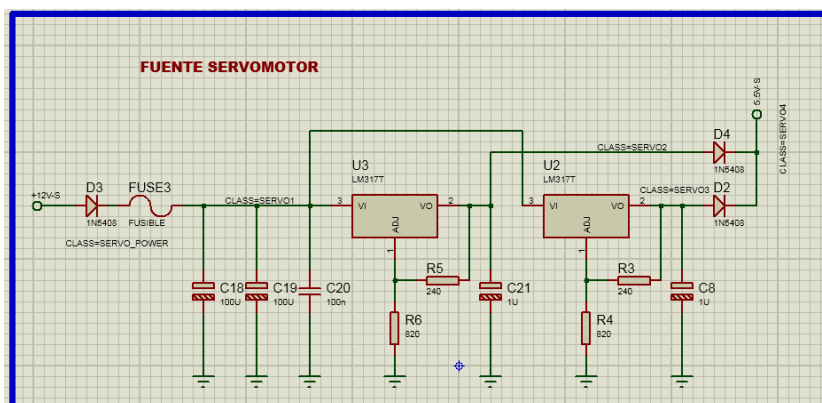


Figura. 4.14. Regulación de fuente para el Servomotor MG995.

Fuente: Proteus Design Suite 8, (2018).

### Circuito de control con transistores TIP122

Para el control del encendido y apagado de la herramienta eléctrica (Dremel) por medio de un relé, se empleó un transistor TIP122, pero se agregó una salida adicional para otra herramienta eléctrica en el caso de ser necesaria. Por lo cual se utilizarán dos relés y dos transistores. Los cálculos fueron establecidos en el capítulo 3 y en la figura 4.15 se demuestra su aplicación. Los diodos D6 y D5 fueron colocados para evitar el flujo no deseado de voltaje, desde y hacia la misma fuente.

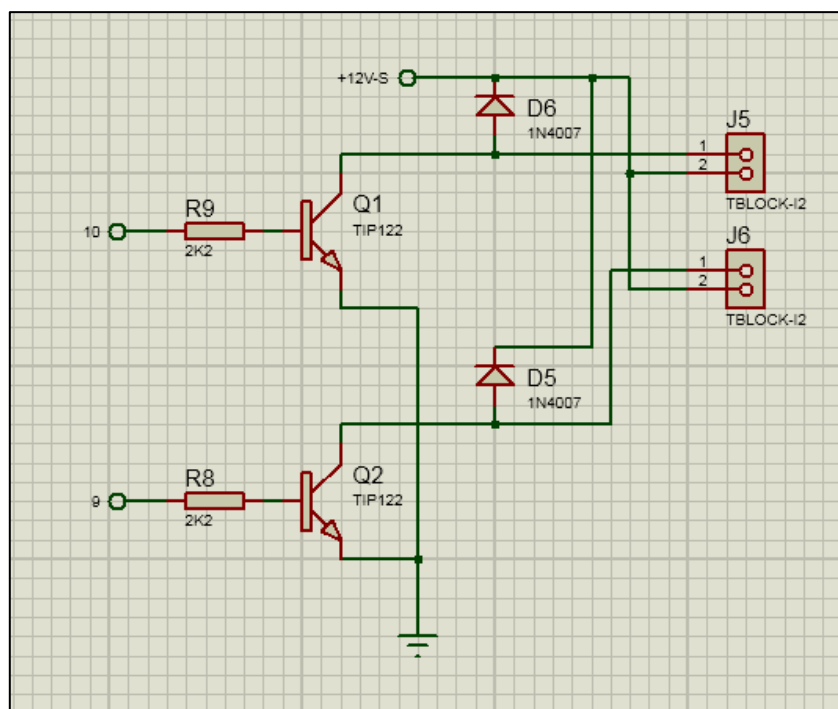


Figura. 4.15. Circuito de control con transistores TIP122.

Fuente: Proteus Design Suite 8, (2018).

### Esquema de control utilizando al Atmega 328

De igual forma en el capítulo I, se especifican las conexiones para un Atmega 328 estipulada por Atmel Corporation. Por lo que a continuación en la figura 4.16 se presenta el diagrama electrónico realizado en Proteus Design Suite 8.

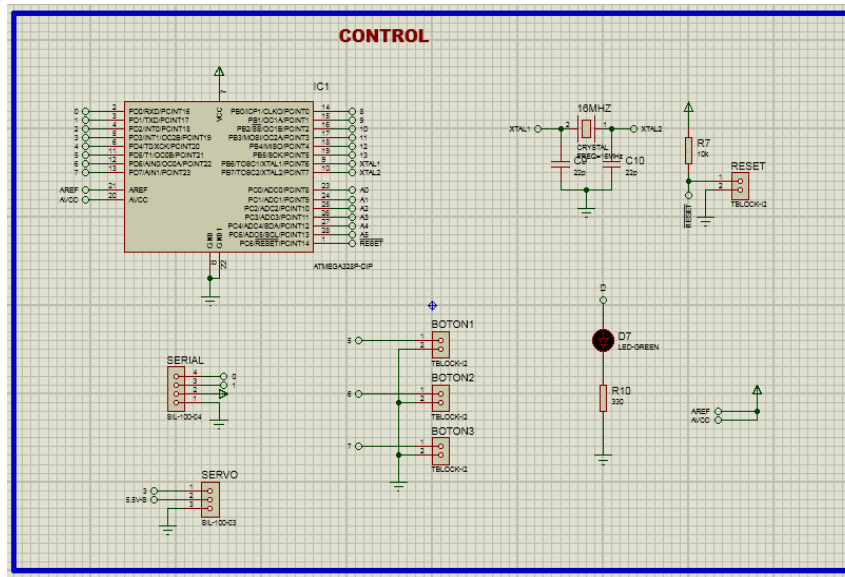


Figura. 4.16. Esquema de control utilizando al Atmega 328.

Fuente: Proteus Design Suite 8, (2018).

### Esquema electrónico finalizado

Luego de finalizar el diseño del circuito y efectuar su simulación, se realizó su esquema para ser colocado en una PCB, en la figura 4.17 se puede apreciar el diseño para la PCB utilizando Proteus Design Suite 8. Además en la figura 4.18 se aprecia su diseño para ser impreso sobre papel termo transferible, así mismo en la figura 4.19, se muestra su esquema a ser impreso en la parte superior de la placa, para tener una mejor percepción de la misma al momento de colocar los componentes.

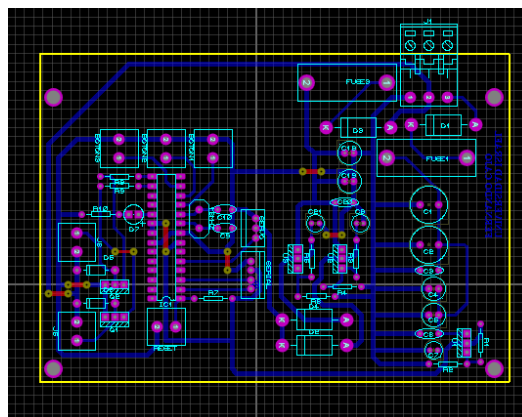
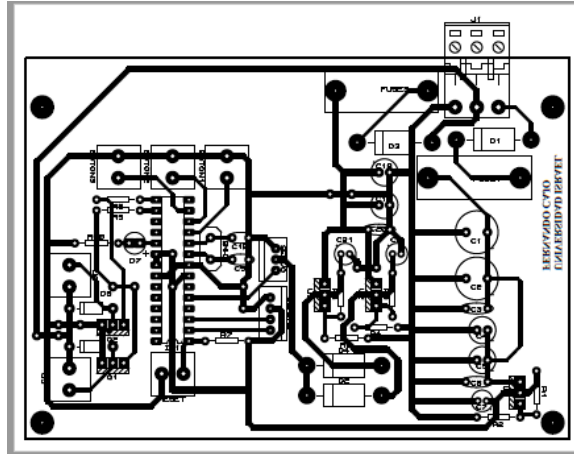


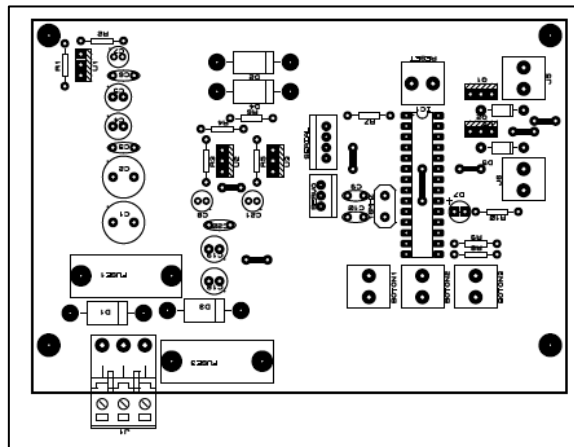
Figura. 4.17. Diseño de la PCB.

Fuente: Proteus Design Suite 8, (2018).



**Figura. 4.18. Diseño para ser impreso.**

Fuente: Proteus Design Suite 8, (2018).

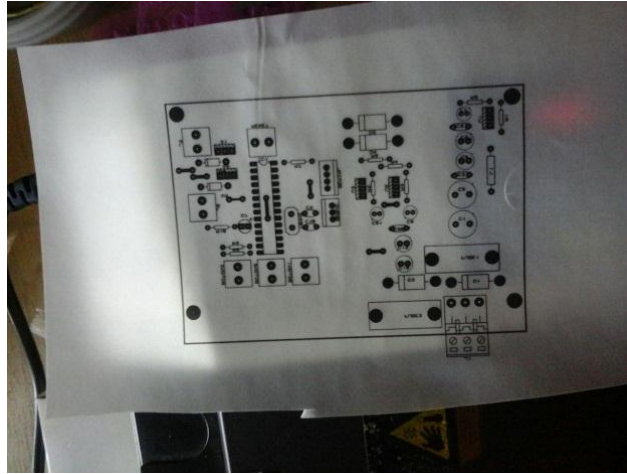


**Figura. 4.19. Esquema para ser impreso en la parte superior de la placa.**

Fuente: Proteus Design Suite 8, (2018).

### **Fabricación de la placa**

Después del diseño de la placa, se llevó a cabo su fabricación, proceso en el cual, se imprimió a laser en papel termo trasferible el circuito, para luego trasladar el esquema hacia la placa de cobre, aplicando calor desde una máquina emplastadora, posteriormente, se quemó la placa utilizando Cloruro Férrico, consecutivamente, se limpió la placa y se efectuó la colocación de los componentes electrónicos y finalmente los elementos fueron soldados a la placa. De la figura 4.20 a la 4.24 se puede apreciar este proceso.



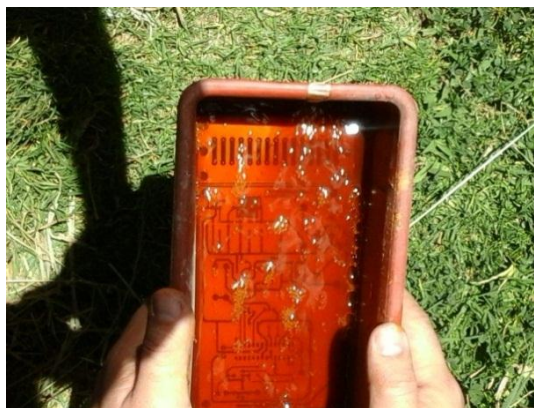
**Figura. 4.20. Impresión del circuito en papel termo transferible.**

Fuente: Elaborado por el autor.



**Figura. 4.21. Aplicación de calor para transferir diseño de placa.**

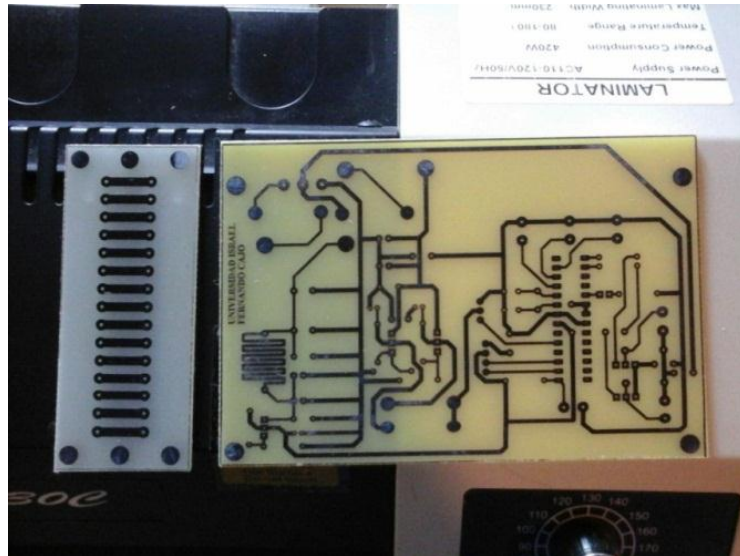
Fuente: Elaborado por el autor.



**Figura. 4.22. Quemado de placa usando Cloruro Férrico.**

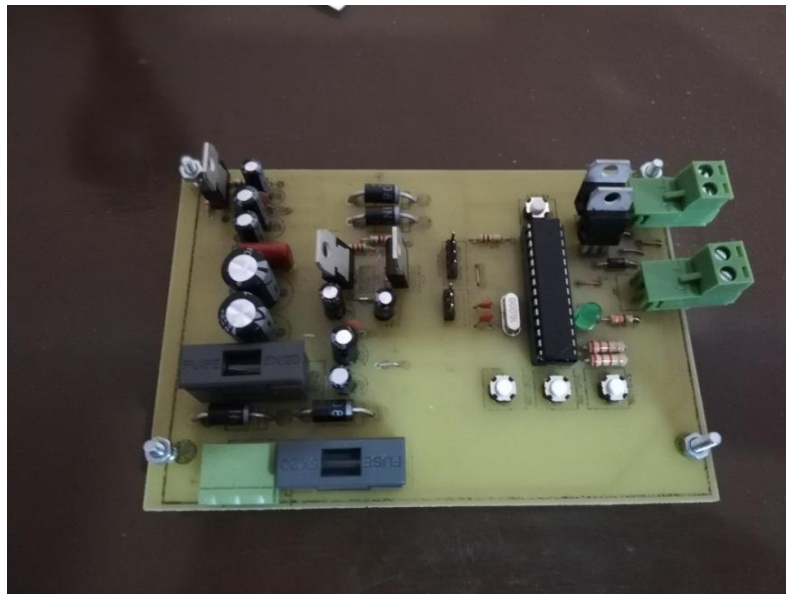
Fuente: Elaborado por el autor.





**Figura. 4.23. Placas listas para ser perforadas.**

Fuente: Elaborado por el autor.

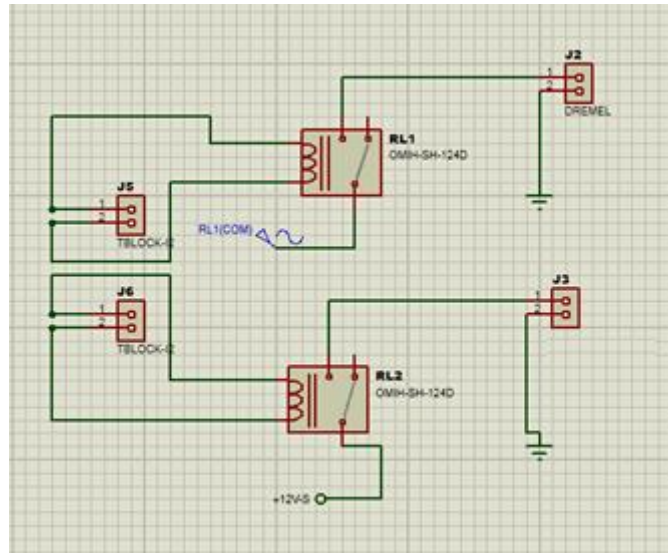


**Figura. 4.24. Montaje de componentes y finalización de fabricación.**

Fuente: Elaborado por el autor.

### **Diagrama eléctrico para la colocación de relés y de la herramienta (Dremel).**

Luego de la implementación de la tarjeta electrónica, es importante conocer el diagrama de conexión eléctrico para los relés y la herramienta (Dremel). A continuación en la figura 4.25, se puede apreciar el diagrama eléctrico.



**Figura. 4.25. Diagrama eléctrico.**

Fuente: Proteus Design Suite 8, (2018).

### 4.1.3 Diseño de software

Para el diseño de software se utilizaron dos herramientas: la (IDE) de Arduino y la interfaz Processing. La primera herramienta fue empleada para crear la programación encargada de controlar el funcionamiento de la herramienta eléctrica (Dremel). Dentro de las funciones del Dremel se puede controlar: (encendido, apagado y velocidad).

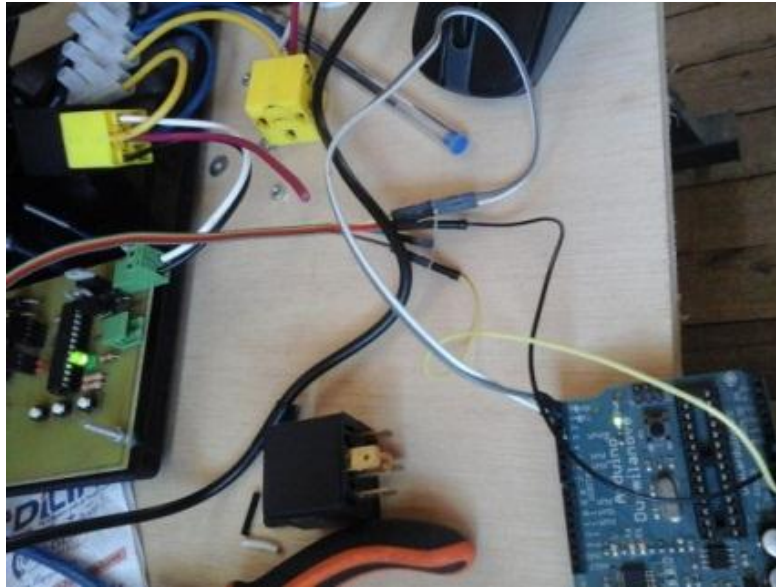
De modo similar fue utilizado el software Processing, con el cual se generó una interfaz capaz de controlar las siguientes acciones: movimientos de los ejes X,Y,Z, velocidad, encendido y apagado del Dremel, enviar código G hacia la impresora 3D, cargar archivo de corte/grabado, efectuar el corte/grabado desde la interfaz y finalmente mostrar las posiciones actuales de los ejes X,Y,Z.

- **Programación en Arduino**

Siguiendo la descripción del software empleado, se comenzará con la (IDE) de Arduino para luego continuar con la interfaz de Processing. Para la visualización de las líneas de código en la (IDE) de Arduino con una descripción del trabajo que efectúa cada bloque de programación dirigirse al ANEXO 5.

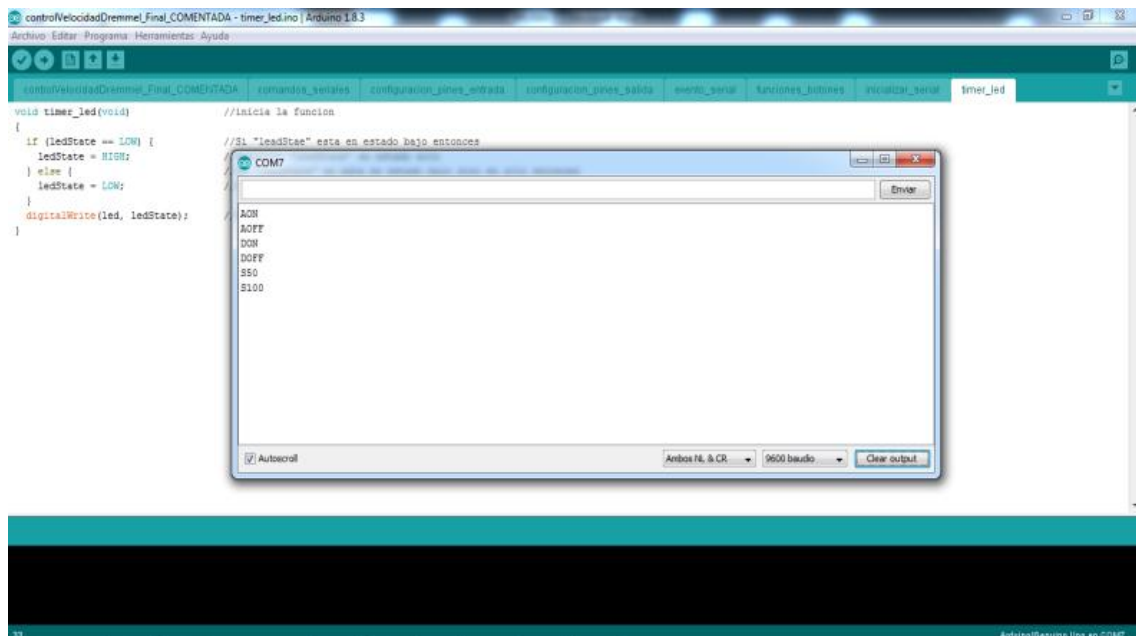


Finalmente, luego de crear la programación, se la cargó en el Atmega 328 para luego hacer pruebas de funcionamiento, esto se realizó utilizando el software Serial Bluetooth. En las figuras (4.26 – 4.28) se puede apreciar dicho proceso.



**Figura. 4.26.** Subiendo archivo compilado al Atmega 328.

Fuente: Elaborado por el autor.



**Figura. 4.27.** Prueba de funcionamiento del programa en Arduino.

Fuente: Elaborado por el autor.



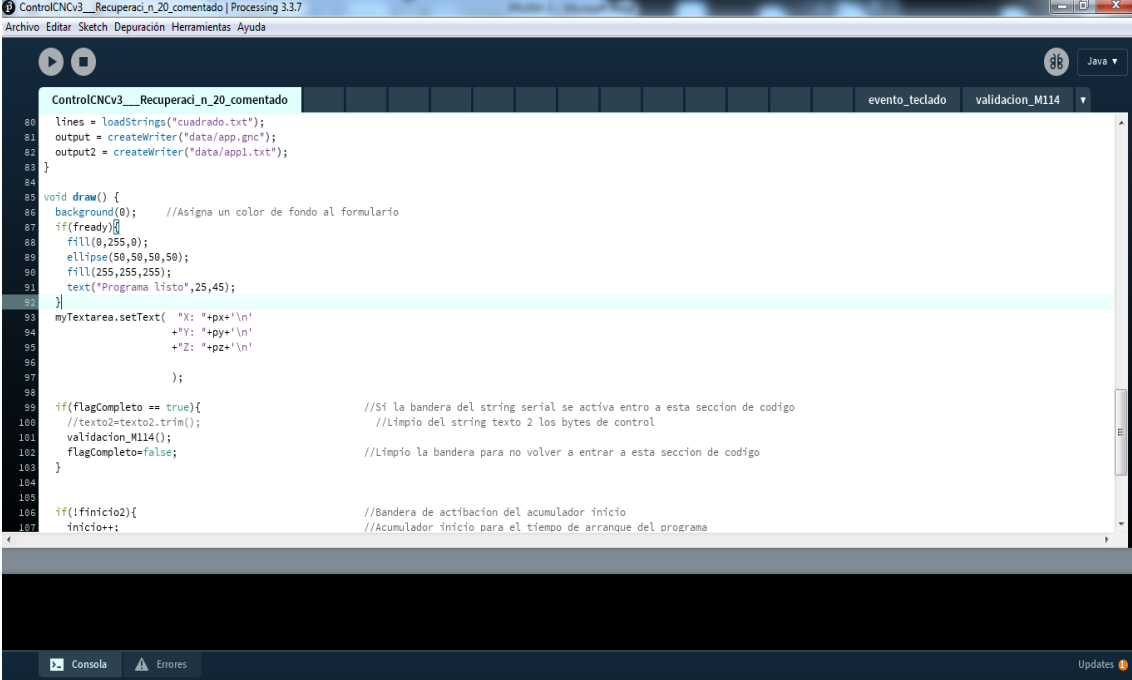
**Figura. 4.28. Prueba de funcionamiento del programa en Serial Bluetooth.**

Fuente: Serial Bluetooth Terminal, (2018).

- **Programación en Processing**

Luego de la programación en Arduino, se procede a programar en Processing para la creación de la interfaz que será la encargada de cumplir con las funciones de la fresadora CNC de (corte/grabado). De igual forma para la visualización de líneas de código dirigirse al ANEXO 6:

Como se puede apreciar, la programación en Processing es muy extensa, pero para tener una mejor apreciación de lo que fue la programación en la figura 4.29 se presenta una captura de su interfaz, así mismo en la figura 4.30 se muestra el producto de esta programación.



```

ControlCNCv3__Recuperaci_n_20_comentado | Processing 3.3.7
Archivo Editar Sketch Depuración Herramientas Ayuda

ControlCNCv3__Recuperaci_n_20_comentado
evento_teclado validacion_M114
80 lines = loadStrings("cuadrado.txt");
81 output = createWriter("data/app.gnc");
82 output2 = createWriter("data/app1.txt");
83 }
84
85 void draw() {
86   background(0); //Asigna un color de fondo al formulario
87   if(fready){
88     fill(0,255,0);
89     ellipse(50,50,50,50);
90     fill(255,255,255);
91     text("Programa listo",25,45);
92   }
93   myTextarea.setText("X: "+px+"\n"
94                     +"Y: "+py+"\n"
95                     +"Z: "+pz+"\n");
96
97   );
98
99   if(flagCompleto == true){ //Si la bandera del string serial se activa entro a esta seccion de codigo
100     //texto2=texto2.trim(); //Limpio del string texto 2 los bytes de control
101     validacion_M114();
102     flagCompleto=false; //Limpio la bandera para no volver a entrar a esta seccion de código
103   }
104
105
106   if(!inicio2){ //Bandera de actibacion del acumulador inicio
107     inicio++; //Acumulador inicio para el tiempo de arranque del programa

```

Figura. 4.29. Interfaz de programación en Processing.

Fuente: Processing, (2018).

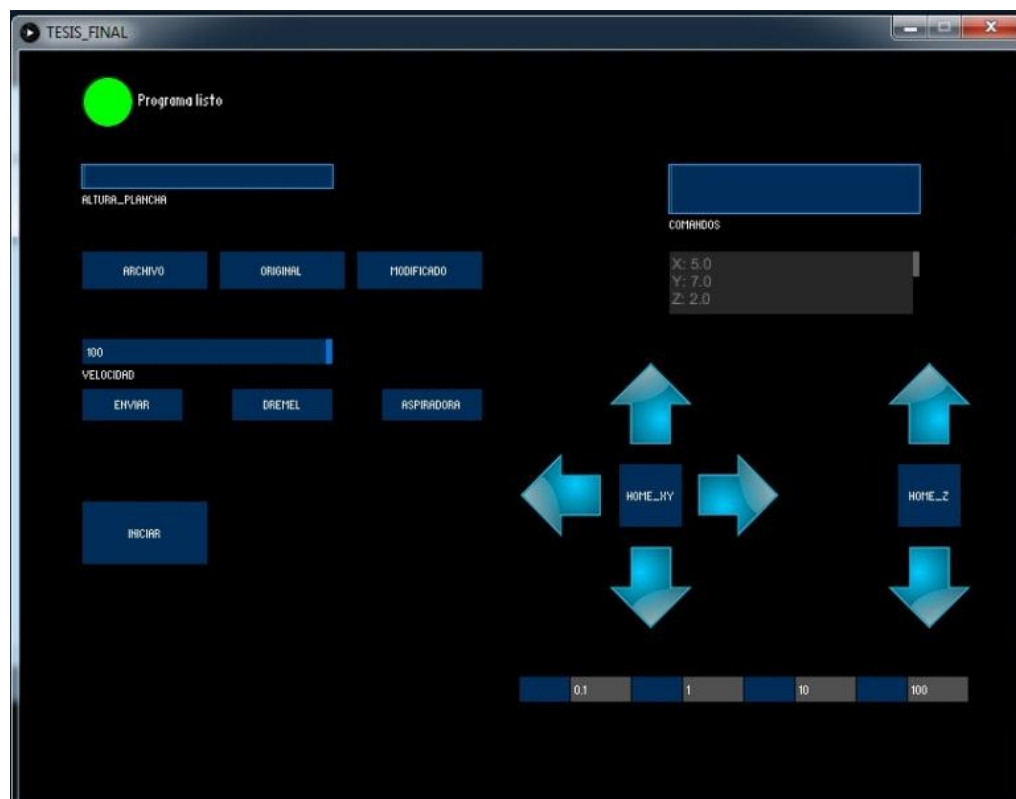


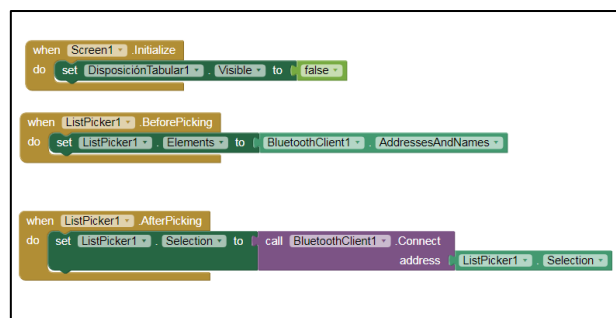
Figura. 4.30. Interfaz para el control de la CNC (cortadora / grabadora).

Fuente: Processing, (2018).

- **Programación en App Inventor 2**

Para realizar la interfaz que permita el control de desplazamiento direccional para la fresadora, se realizó un App en el software antes mencionado. A continuación se presenta la programación que se realizó.

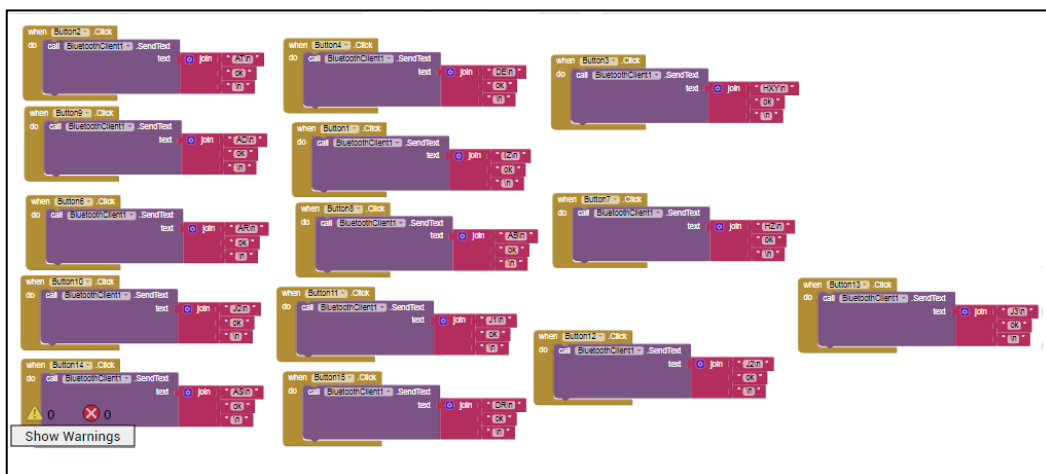
- **Establecimiento de comunicación vía Bluetooth**, por medio de un botón se puede desplegar una lista de servidores a los cuales vincularse. (Figura 4.31).



**Figura. 4.31. Comunicación Bluetooth.**

Fuente: App Inventor 2, (2018).

- **Establecimiento del funcionamiento de los botones de avance**: izquierda, derecha, arriba, abajo, adelante, atrás, así como los botones de home y los que seleccionan la cantidad de avance en cada movimiento. (Figura 4.32).



**Figura. 4.32. Botones de avance.**

Fuente: App Inventor 2, (2018).

- **Control de velocidad del Dremel**, aquí se genera una variable la cual irá cambiando con los valores entregados por el slider para luego enviarlos por la serial del teléfono. (Figura 4.33).

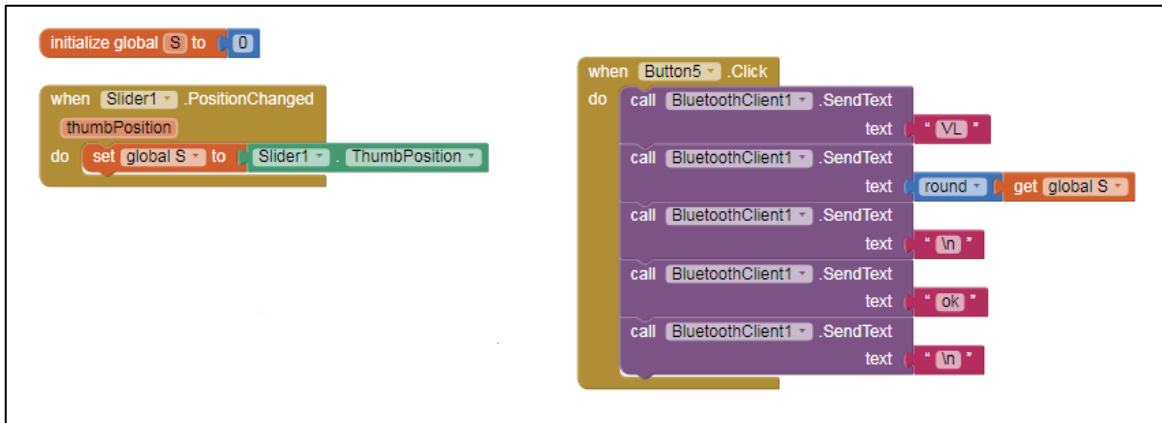


Figura. 4.33. Control de velocidad.

Fuente: App Inventor 2, (2018).

- **Intercambio de pantallas**, el intercambio de pantallas se genera para no perder la comunicación Bluetooth al cambiar de una interfaz a otra. (Figura 4.34)

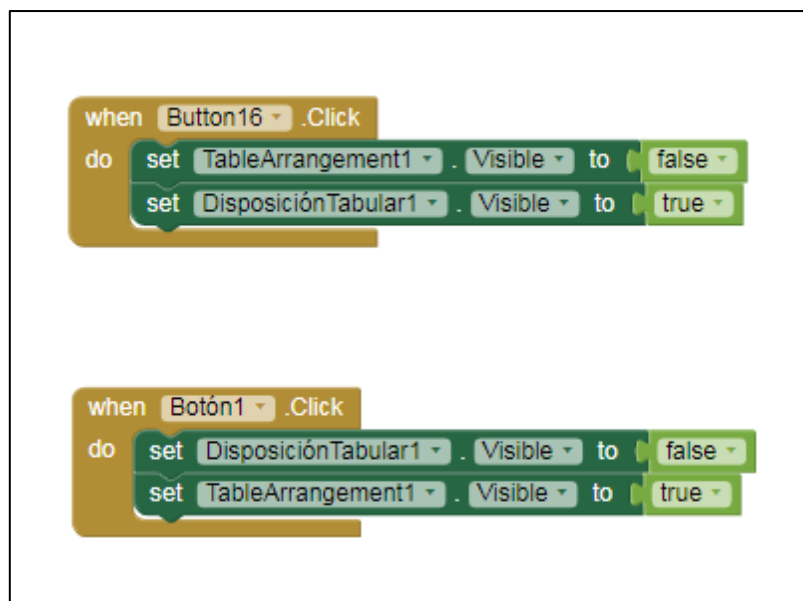


Figura. 4.34. Intercambio de pantallas.

Fuente: App Inventor 2, (2018).

- **Programación de Canvas.** Canvas es una herramienta que nos permite dibujar sobre la pantalla del smartphone, para luego enviar esos puntos generados por la herramienta, por la serial del teléfono. (Figura 4.35 – 4.37).

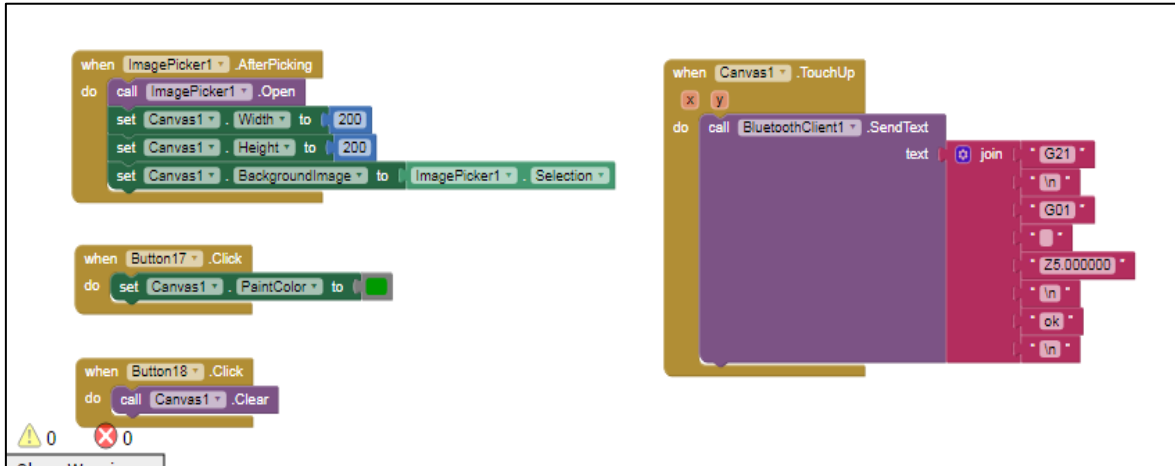


Figura. 4.35. Programación de Canvas.

Fuente: App Inventor 2, (2018).

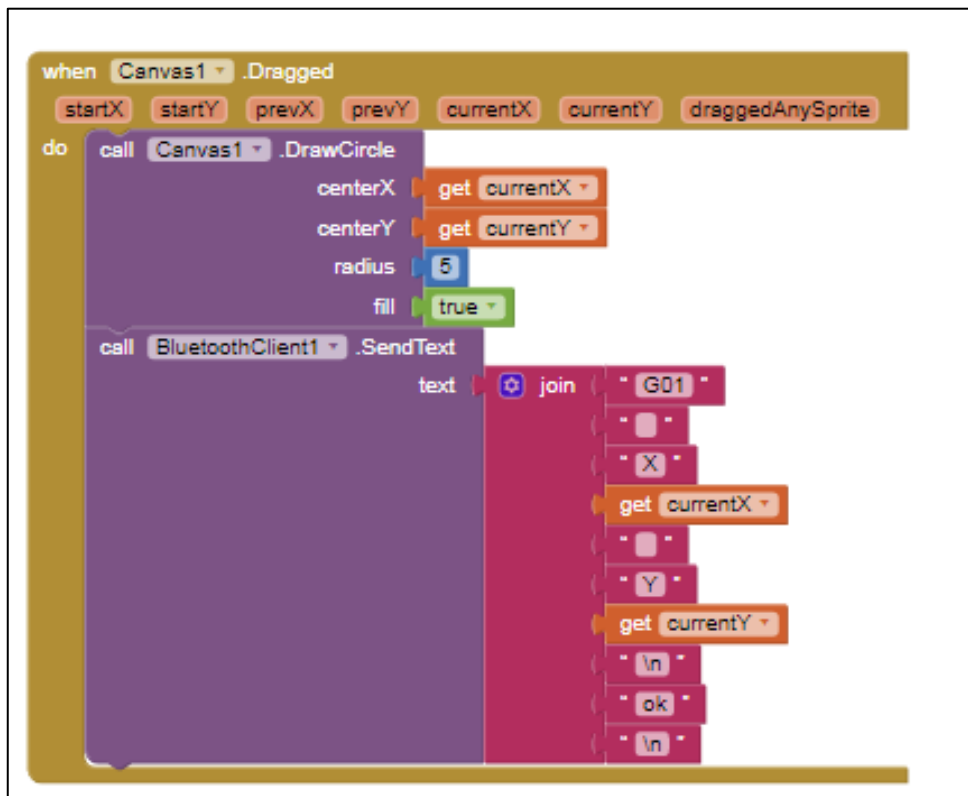


Figura. 4.36. Programación de Canvas.

Fuente: App Inventor 2, (2018).

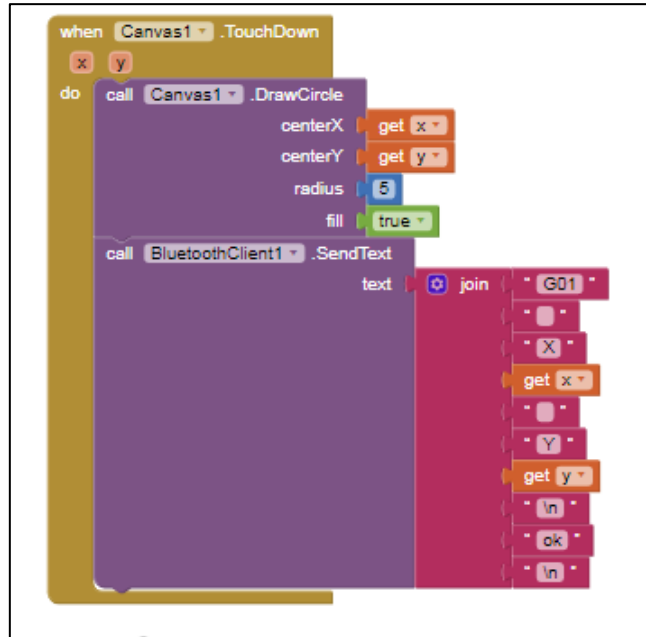


Figura. 4.37. Programación de Canvas.

Fuente: App Inventor 2, (2018).

- **Envío de comandos G**, se recogen los datos escritos en el text box, guardados en una variable llamada PROF y se los envía por la serial, al utilizar el sensor de movimiento del smartphone. (Figura, 4.38)

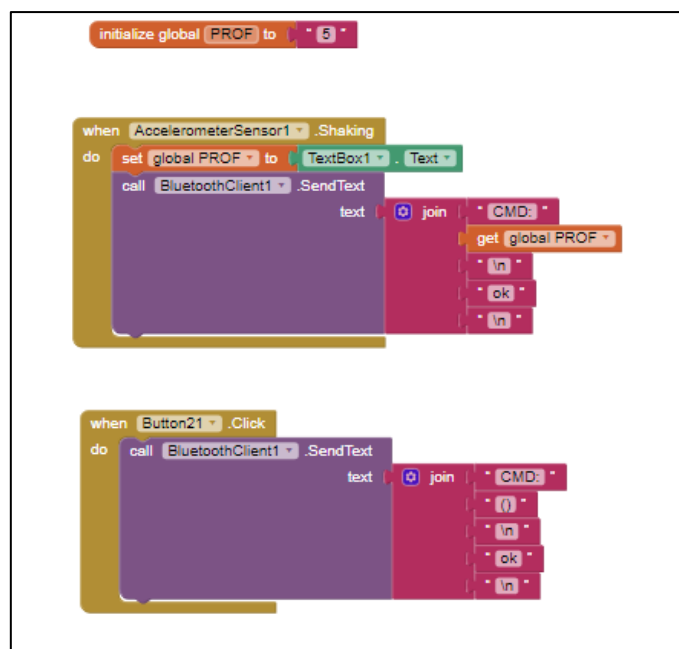
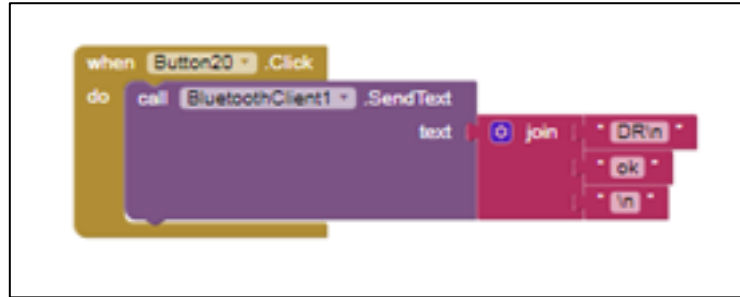


Figura. 4.38. Envío de comandos G.

Fuente: App Inventor 2, (2018).

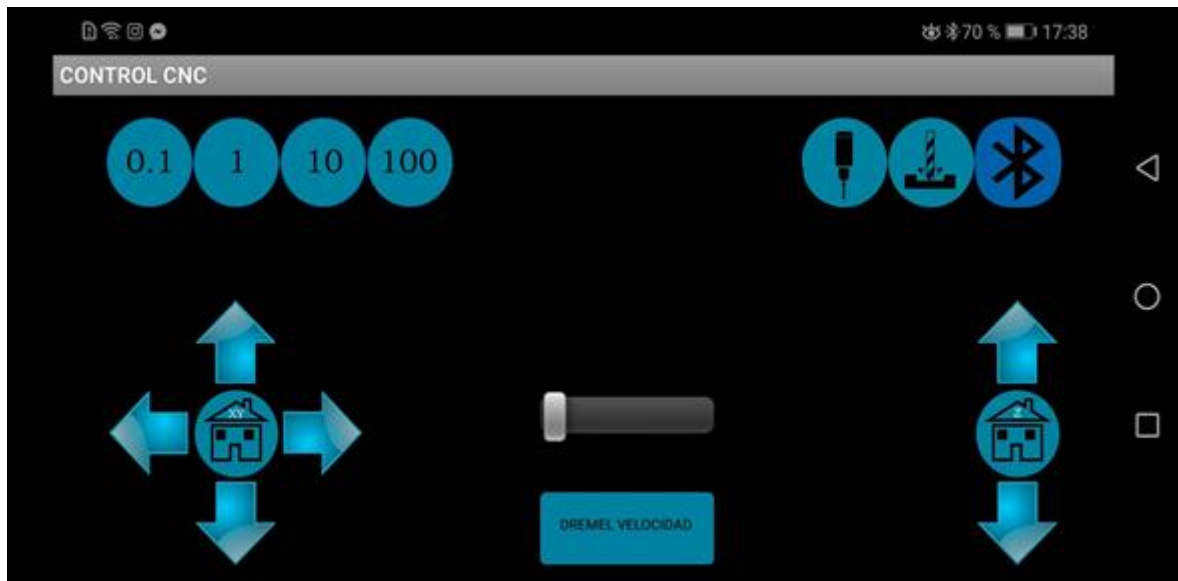
- **Control de encendido y apagado desde la segunda interfaz**, con este botón se puede encender o apagar la herramienta eléctrica (Dremel). (Figura 4.39).



**Figura. 4.39. Control (Dremel).**

Fuente: App Inventor 2, (2018).

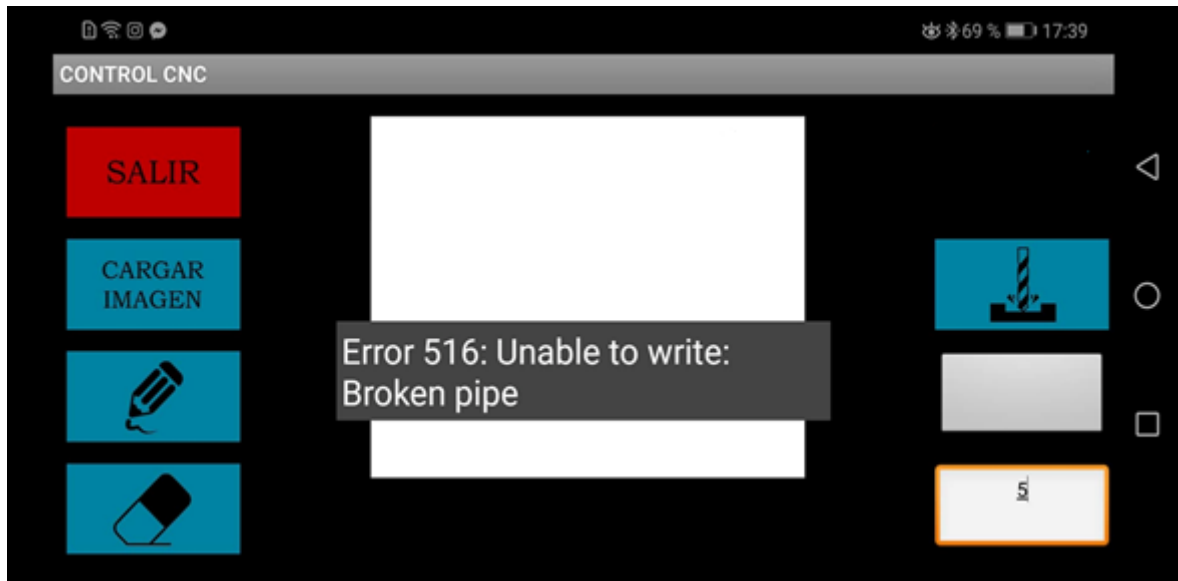
Finalmente se puede observar el resultado de la programación y hacer sus respectivas pruebas de funcionalidad, en la figura 4.40 y 4.41 se puede observar el resultado del diseño de la aplicación.



**Figura. 4.40. Interfaz principal del control de desplazamiento direccional.**

Fuente: App Inventor 2, (2018).



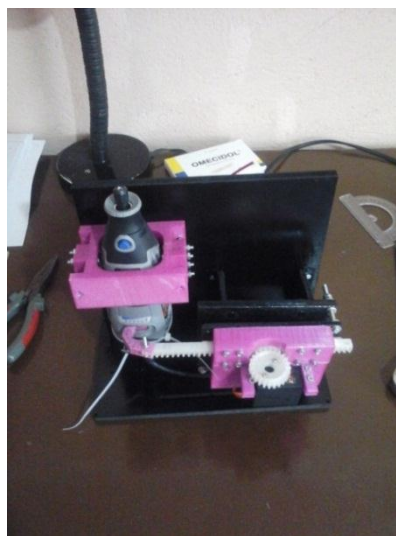


**Figura. 4.41.** Interfaz secundaria del control de desplazamiento direccional.

Fuente: App Inventor 2, (2018).

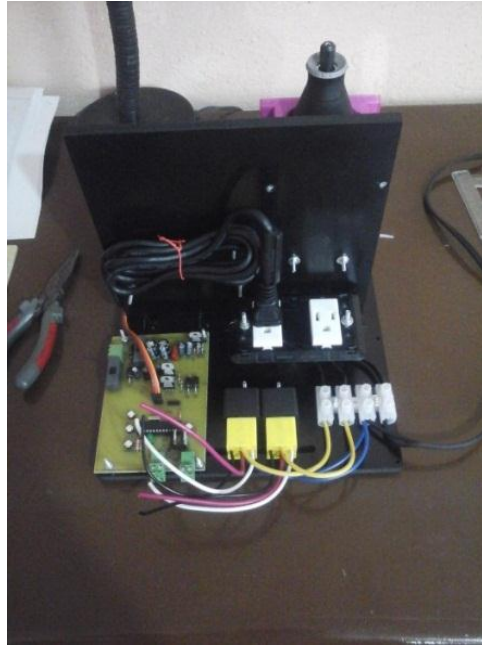
## 4.2 Implementación

Una vez listo todo lo necesario a nivel de hardware, software y acople, es la hora de montarlo todo para poner en marcha el proyecto. Primero se realiza la elaboración de un módulo para colocar allí la herramienta giratoria, circuito de control electrónico, circuito eléctrico, esto se lo puede apreciar en las siguientes figuras: 4.42 y 4.43.



**Figura. 4.42.** Montaje del Dremel y servomotor en el módulo.

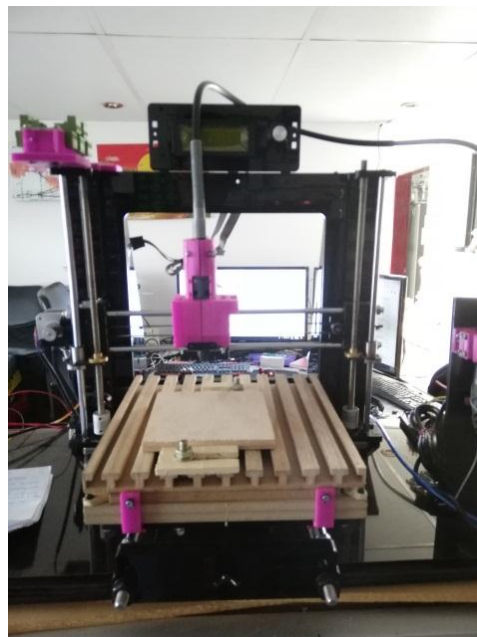
Fuente: Elaborado por el autor.



**Figura. 4.43. Montaje del circuito electrónico y eléctrico en el módulo.**

Fuente: Elaborado por el autor.

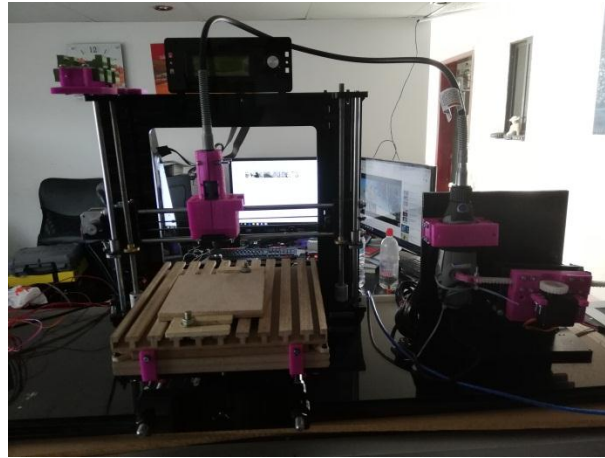
Una vez listo el módulo, es necesario colocar los acoples para la CNC fresadora sobre la estructura de la impresora 3D. Estos acoples son: cabezal de fresado, plataforma de (corte/grabado) y acople para borneras, (Figura 4.44).



**Figura. 4.44. Acoples para la CNC fresadora.**

Fuente: Elaborado por el autor.

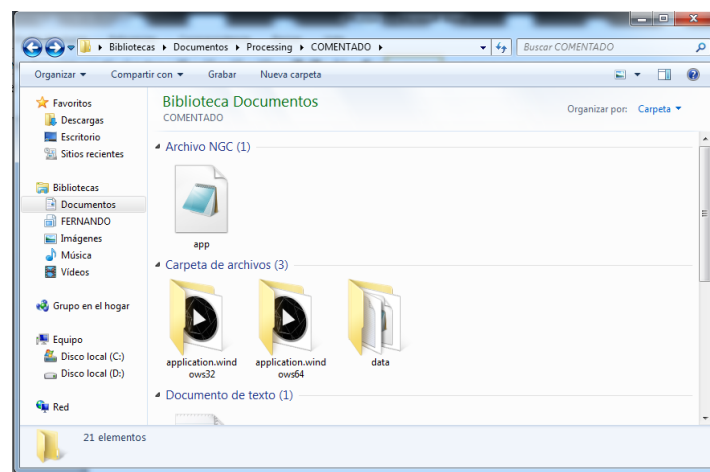
Después de colocar los acoples en la impresora 3D y haber armado el módulo de la CNC, se pueden conectar ambas estructuras por medio del flex del Dremel. (Figura 4.45)



**Figura. 4.45. Estructura final para la CNC de (corte/grabado).**

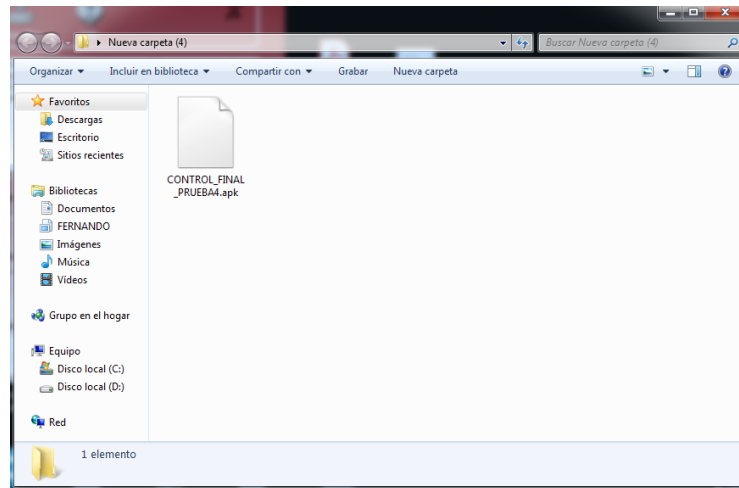
Fuente: Elaborado por el autor.

A nivel de software, la implementación se realiza al momento de subir el programa al Atmega 328, proceso que ya fue previamente demostrado, por otra parte el software realizado en Processing es exportado como un archivo ejecutable para el sistema operativo Windows tanto para 32 como para 64 bits, (figura 4.46). Y de igual forma sucede con la aplicación realizada en App Inventor, al momento de generar un paquete de instalación para el smartphone, (figura 4.47).



**Figura. 4.46. Archivos ejecutables para la interfaz CNC.**

Fuente: Elaborado por el autor.



**Figura. 4.47. Paquete de instalación del control para smartphone.**

Fuente: Elaborado por el autor.

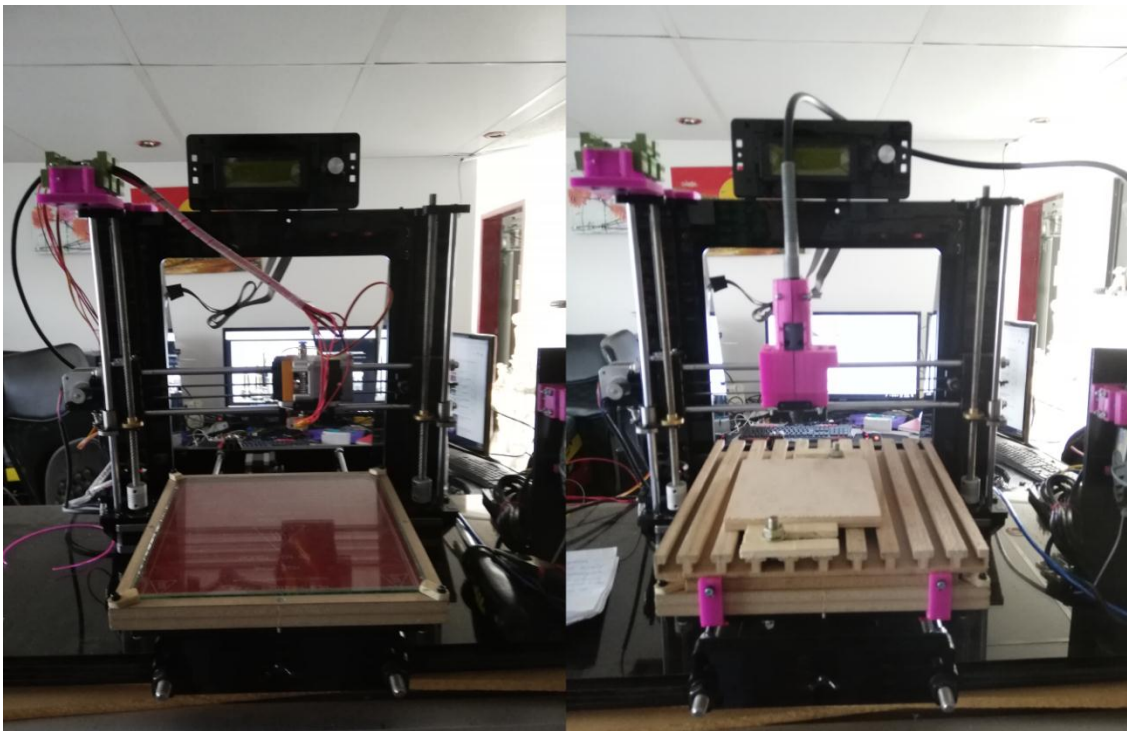
El aporte que brinda esta implementación, es el siguiente, con la colocación de pocos elementos fijos en la estructura de la impresora, esta puede brindar funciones de (corte/grabado) sin perder sus funciones de impresión, por lo cual solo hace falta retirar la plataforma de corte/grabado para poder utilizar la cama de impresión caliente, así mismo solo se debe retirar el cabezal de fresado y colocar nuevamente el cabezal de extrusión para poder imprimir figuras volumétricas. Tomando en cuenta, que al retirar el cabezal de fresado se retiraría todo el módulo de la fresadora, también se libera a la impresora del módulo CNC de fresado completamente, de esta forma la impresora trabajaría libremente.

De igual forma a nivel de software la CNC fresadora utiliza su propia interfaz de trabajo, por lo cual la impresora 3D se encuentra libre de utilizar cualquier software para desempeñar su labor. Adicional a esto, tanto la interfaz para Windows, como el control remoto para Android, cuentan con gráficos muy intuitivos para facilitar su manejo.

Por otra parte el modulo que alberga las herramientas eléctricas es pequeño y compacto, por lo cual puede ser transportado fácilmente junto con la impresora. En conjunto la implementación del proyecto, resulta de fácil acceso y sin ningún tipo de exigencia extraordinaria para entender su funcionamiento y posterior utilización, esto gracias a que a nivel electrónico, eléctrico y de programación se procuró facilitar todo para el usuario.

### 4.3 Pruebas de funcionamiento

Las pruebas de funcionamiento son extremadamente importantes, ya que por este medio se puede validar el trabajo de titulación, es así que, la primera prueba de funcionalidad surge al momento de desmontar y montar los acoples que faciliten el intercambio de funcionalidades para la impresora 3D. Como se puede observar en la figura 4.48, la transformación que sufre la impresora para pasar de su función a la función de corte/grabado es poco invasiva y de fácil acceso.



**Figura. 4.48. Estructura de la impresora 3D vs CNC fresadora.**

Fuente: Elaborado por el autor.

Para poner a prueba el funcionamiento a nivel de software en la PC (interfaz CNC), se determinaron los siguientes escenarios, primero, comprobación del funcionamiento del movimiento de los ejes X,Y, con su respectivo home para la fresadora CNC (figura 4.49), segundo, comprobación del movimiento del eje Z con su respectivo home (figura 4.50), en este punto, hay que tener en cuenta que los movimientos antes mencionados no se ejecutarán sin haber elegido su nivel de avance entre: 0,1 – 1 – 10 – 100 mm (figura 4.51), tercero, envío de código G a la impresora, este proceso se realizó enviando un código a

través de la casilla y a continuación pulsando “enter” para enviarlo (figura 4.52), cuarto, control de velocidad del Dremel, este proceso se realizó moviendo el slider de velocidad y luego pulsando el botón de envío (figura 4.53), quinto, encendido y apagado del Dremel, esto se logró únicamente pulsando el botón que llevan su nombre (figura 4.54), sexto, visualización de la posición en los ejes X, Y, Z. estas posiciones se pueden apreciar en la casilla que se encuentra en la interfaz (figura 4.55), séptimo, con el botón “archivo” se procede a elegir el archivo modificado y el archivo a ser modificado, a continuación en la (figura 4.56) se muestra el botón que servirá para seleccionar el archivo a ser modificado y el archivo modificado, (figura 4.57) se puede apreciar la reinterpretación que hizo el software al cambiar los valores de profundidad gracias a la utilización de los botones, “original” y “modificado”, octavo, cuando el archivo ha sido subido el software automáticamente lo reinterpreta y está listo para efectuar el corte/grabado y este proceso se logra pulsando el botón “inicio” (figura 4.58), finalmente, existe un círculo de color verde en la parte superior izquierda de la interfaz, su función es indicar que el programa está listo para ser utilizado, esto al momento de abrir el software (figura 4,59).

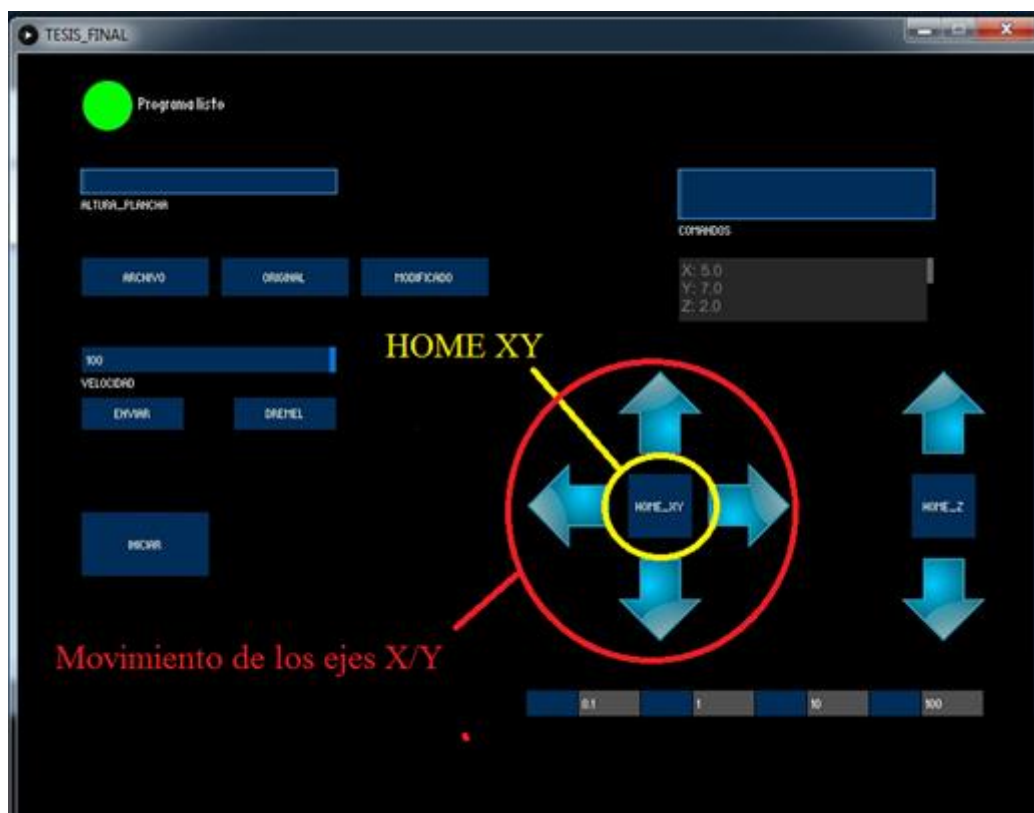


Figura. 4.49. Comprobación del movimiento de los ejes X, Y.

Fuente: Elaborado por el autor.



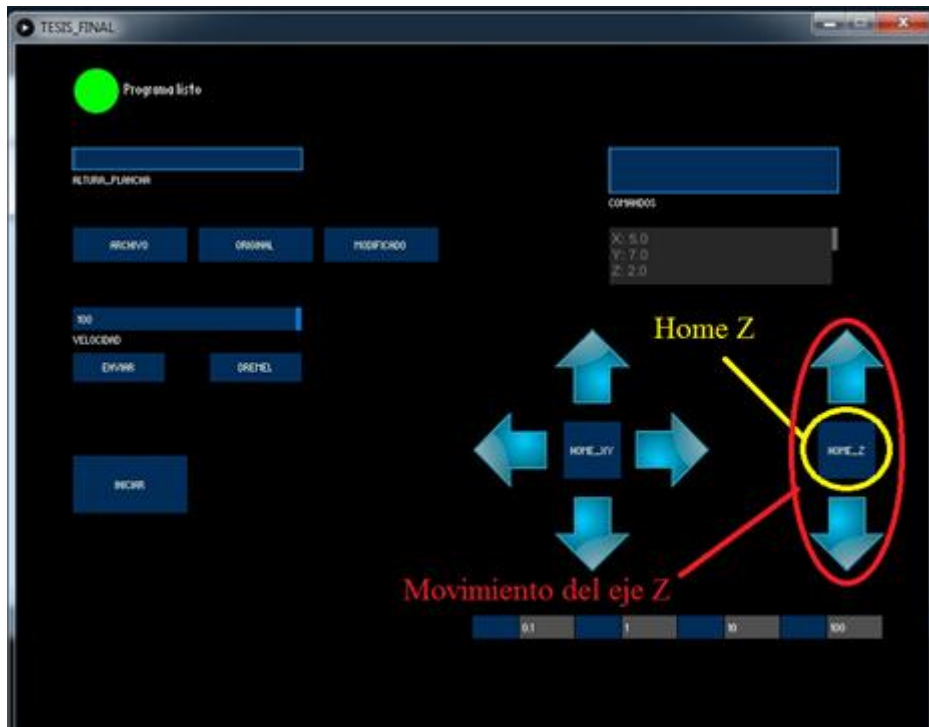


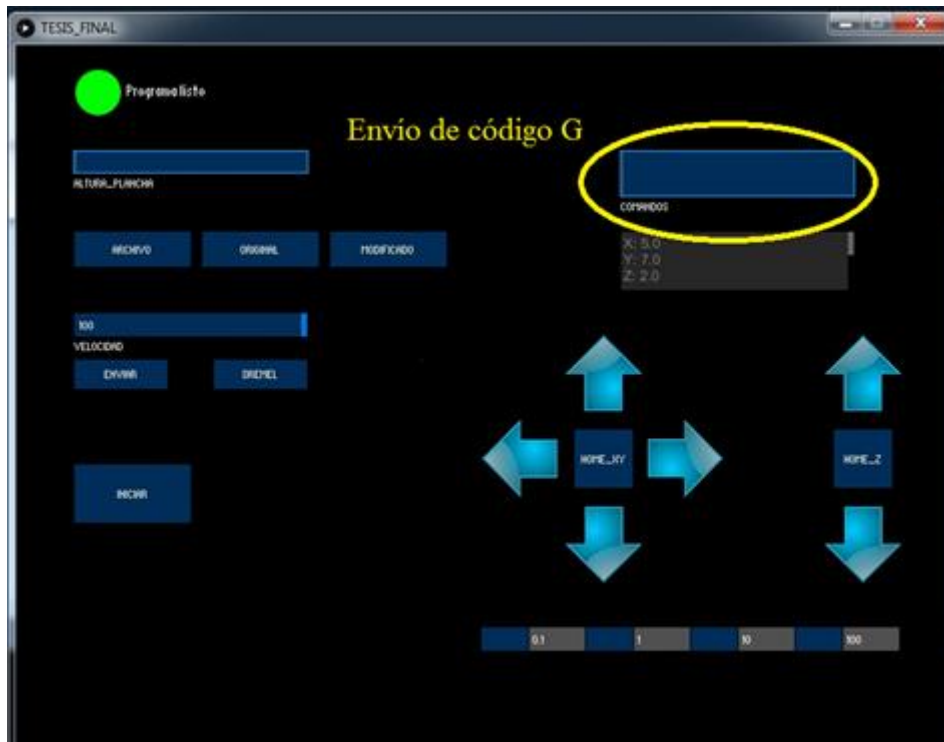
Figura. 4.50. Comprobación del movimiento del eje Z.

Fuente: Elaborado por el autor.



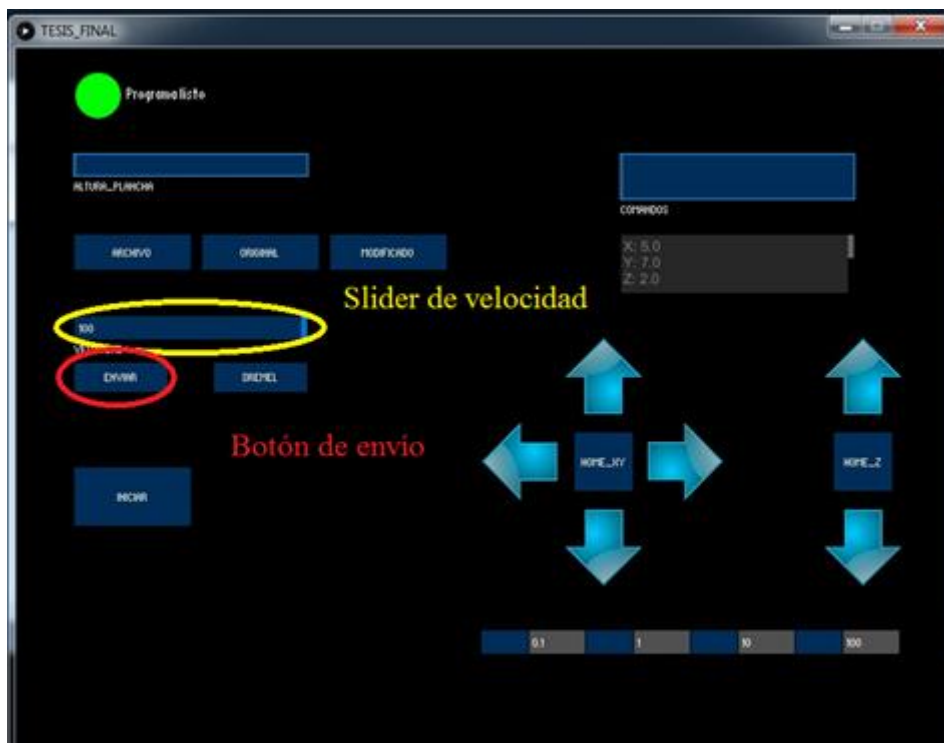
Figura. 4.51. Paso previo al movimiento de los ejes.

Fuente: Elaborado por el autor.



**Figura. 4.52. Envío de código G.**

Fuente: Elaborado por el autor.



**Figura. 4.53. Control de velocidad.**

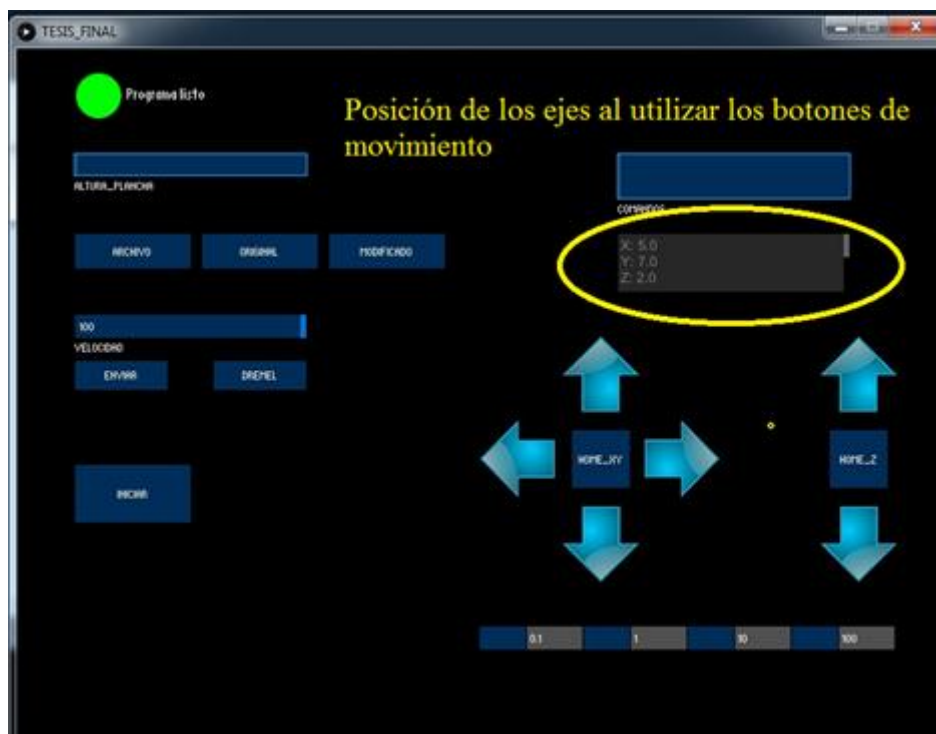
Fuente: Elaborado por el autor.





**Figura. 4.54. Encendido y apagado del Dremel.**

Fuente: Elaborado por el autor.



**Figura. 4.55. Posición actual de los ejes.**

Fuente: Elaborado por el autor.

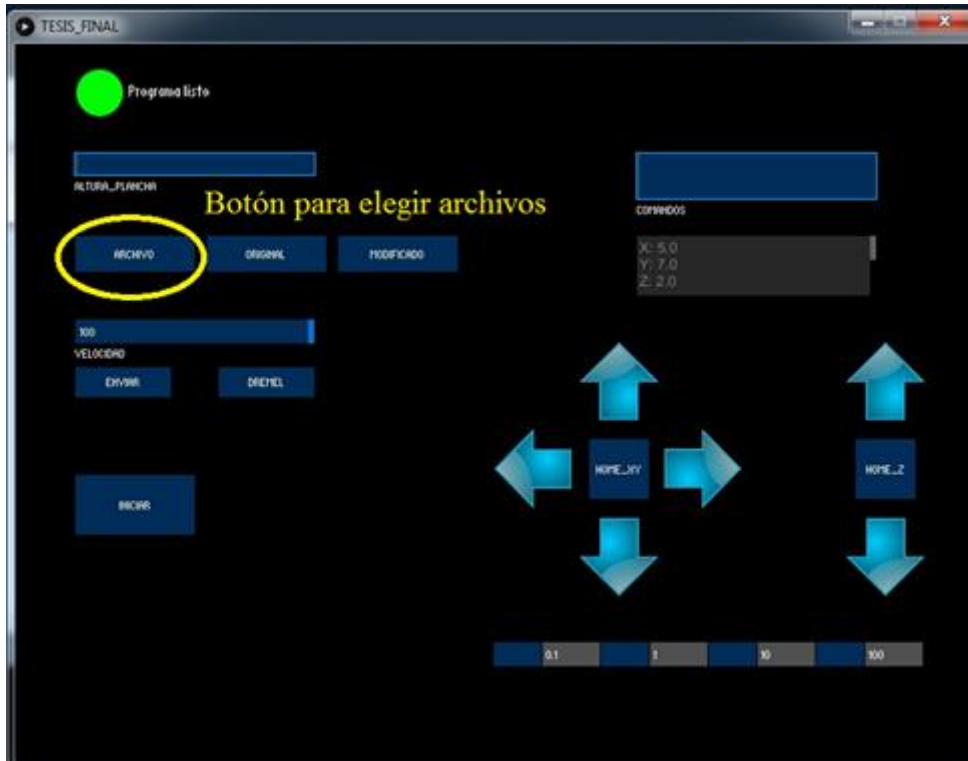


Figura. 4.56. Botón para seleccionar archivos.

Fuente: Elaborado por el autor.

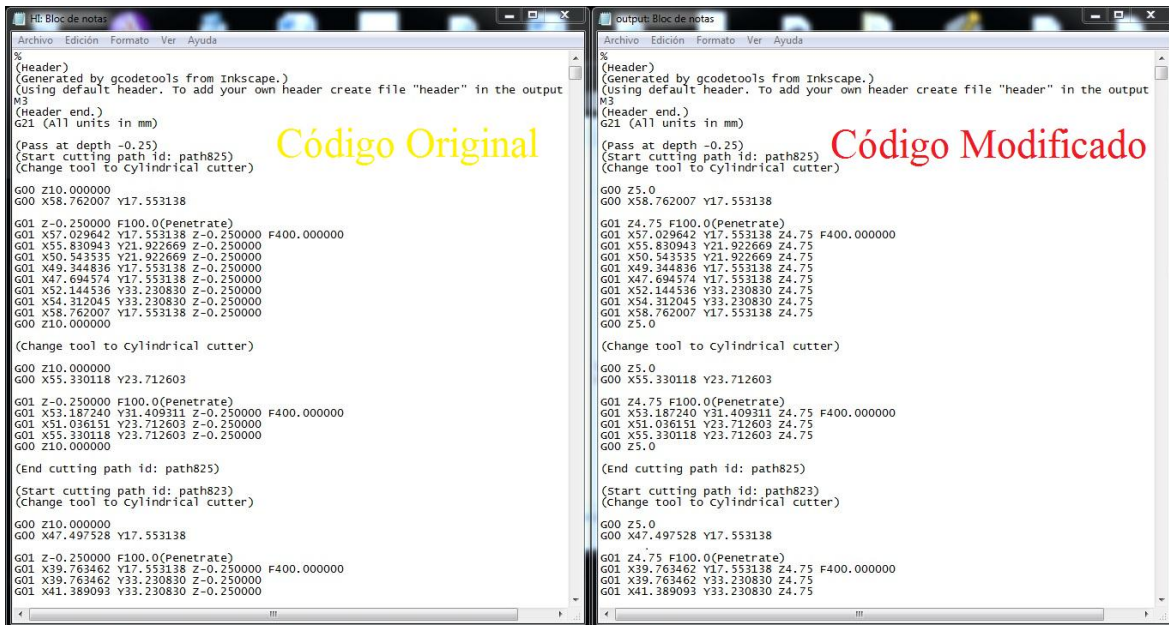


Figura. 4.57. Reinterpretación de código.

Fuente: Elaborado por el autor.

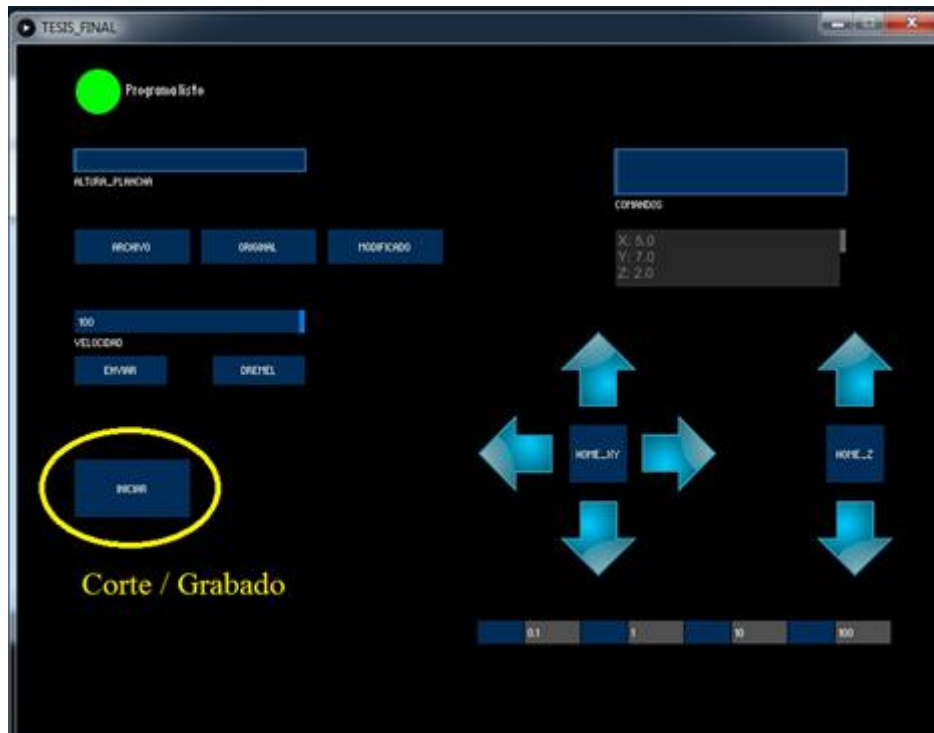


Figura. 4.58. Enviar a cortar / grabar.

Fuente: Elaborado por el autor.

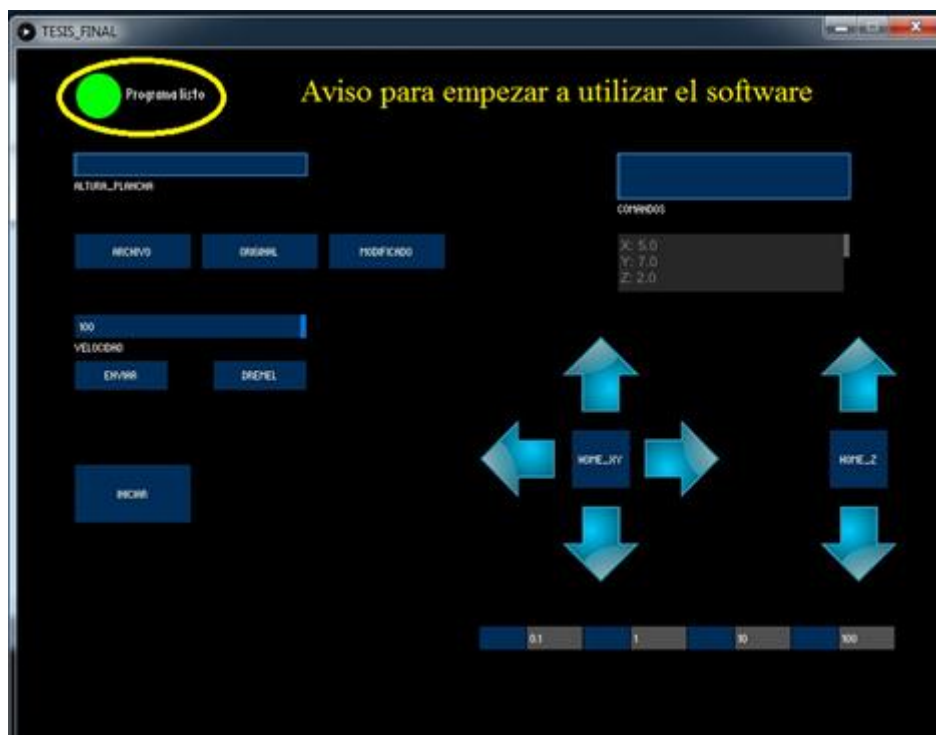
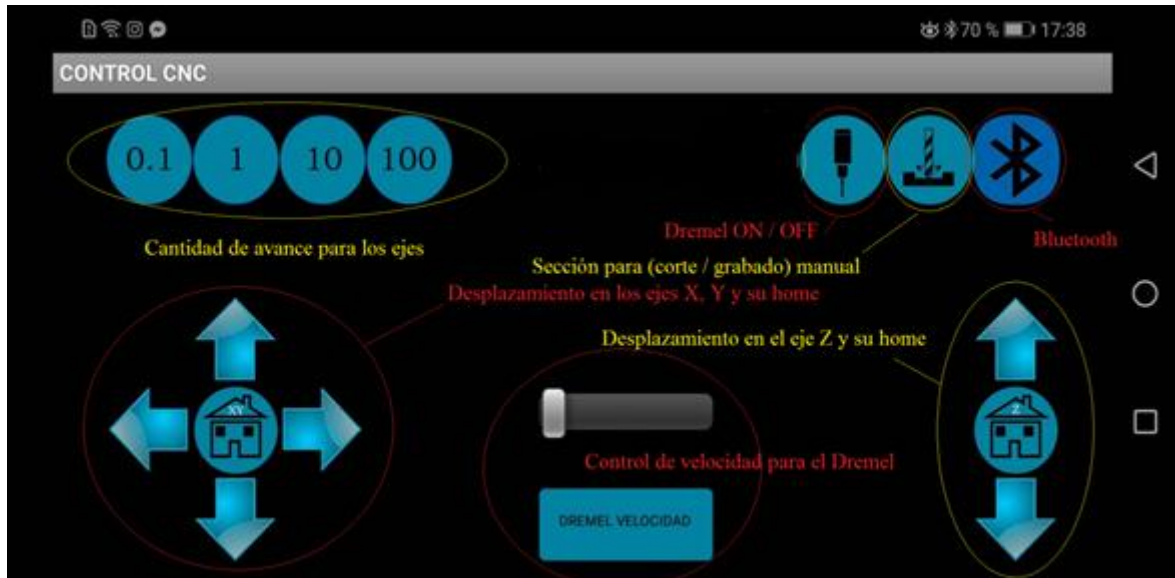


Figura. 4.59. Aviso para empezar a utilizar el software.

Fuente: Elaborado por el autor.

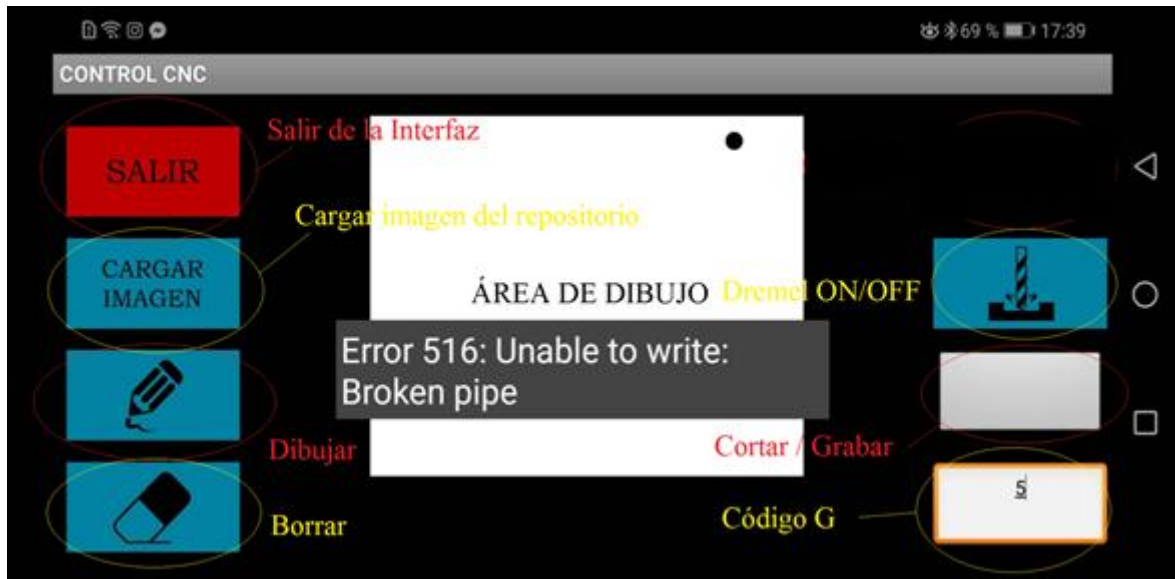
De una manera similar, se puso a prueba la interfaz de la aplicación, la cual es la encargada del desplazamiento direccional para la fresadora a través de un control remoto, como se muestra a continuación en la (figura 4.60) la aplicación ofrece una interfaz amigable para el usuario.



**Figura. 4.60. Desplazamiento direccional para la fresadora.**

Fuente: Elaborado por el autor.

Como un valor agregado, se extendió el desarrollo de la aplicación, la cual brinda la oportunidad de dibujar en ella para que la CNC ejecute sus movimientos, cargar una imagen del repositorio del smartphone para tener una imagen de referencia, además de contar con un cuadro de texto disponible para enviar código G, por otra parte cuenta con la posibilidad de encender o apagar las herramientas. Debido a que fue un valor extra para este trabajo de titulación y por motivos de tiempo esta interfaz no se desarrolló al 100%, pero queda como una constancia de lo que se puede hacer con una aplicación de smartphone para una CNC de corte y grabado. A continuación, se especifican las funciones que fueron probadas en ella (figura 4.61).



**Figura. 4.61. Valor agregado interfaz #2.**

Fuente: Elaborado por el autor.

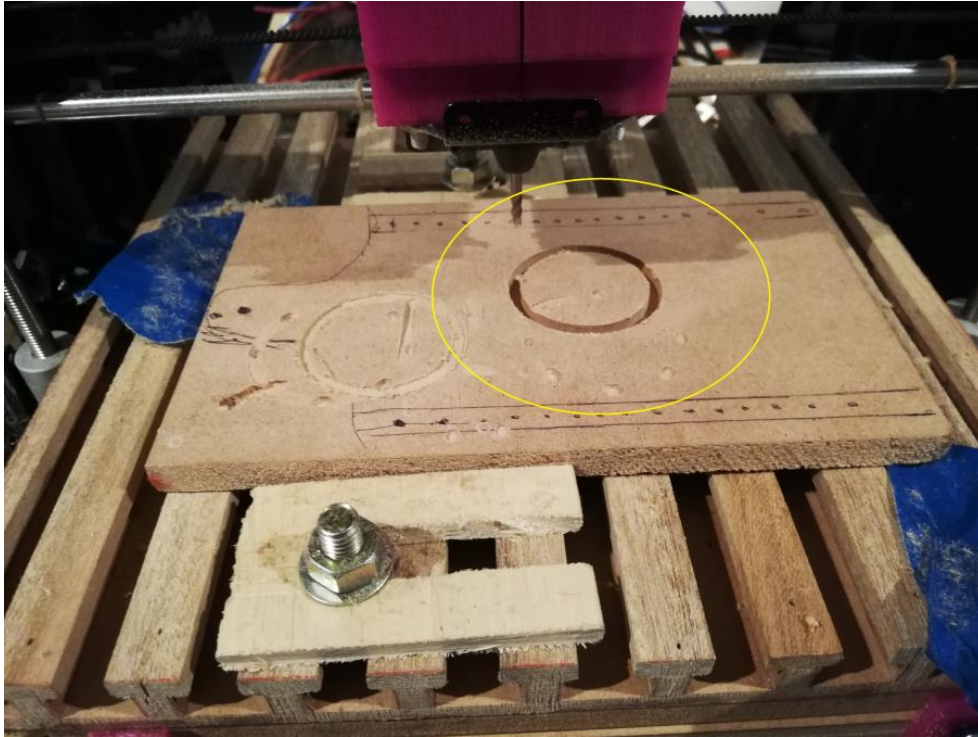
Por último para validar el funcionamiento de (corte/grabado), se realizaron pruebas desde el software de la PC (interfaz) hacia la CNC fresadora. Obteniendo los siguientes resultados: Previo a esta prueba es necesario generar el código G que brinda el software Inkscape de cualquier tipo de imagen. En la (figura 4.62) se puede apreciar el tipo de grabado que realiza la máquina, por otro lado en la (figura 4.63) se observa el tipo de corte que realizó la máquina.



**Figura. 4.62. Grabado realizado por la CNC fresadora.**

Fuente: Elaborado por el autor.





**Figura. 4.63. Corte brindado por la CNC fresadora.**

Fuente: Elaborado por el autor.

#### **4.4 Análisis de resultados**

Los resultados obtenidos son altamente satisfactorios, ya que se cumplieron con todos los objetivos estipulados en el presente proyecto de titulación, además se realizó un pequeño aporte adicional al explorar otras funcionalidades que puede brindar interesantes prestaciones para una CNC de corte y grabado.

Por otra parte, queda demostrada la posibilidad de emplear la infraestructura de una impresora 3D, para generar las funcionalidades que brinda una CNC fresadora en materiales de baja densidad.

##### **4.4.1 Modificación de la infraestructura**

Primero, el proceso de desmontaje del cabezal de extrusión resulta fácil, debido a que gracias a las borneras colocadas en la parte superior izquierda de la impresora, la desconexión de todo el cableado eléctrico es muy simple. Además la sujeción del cabezal

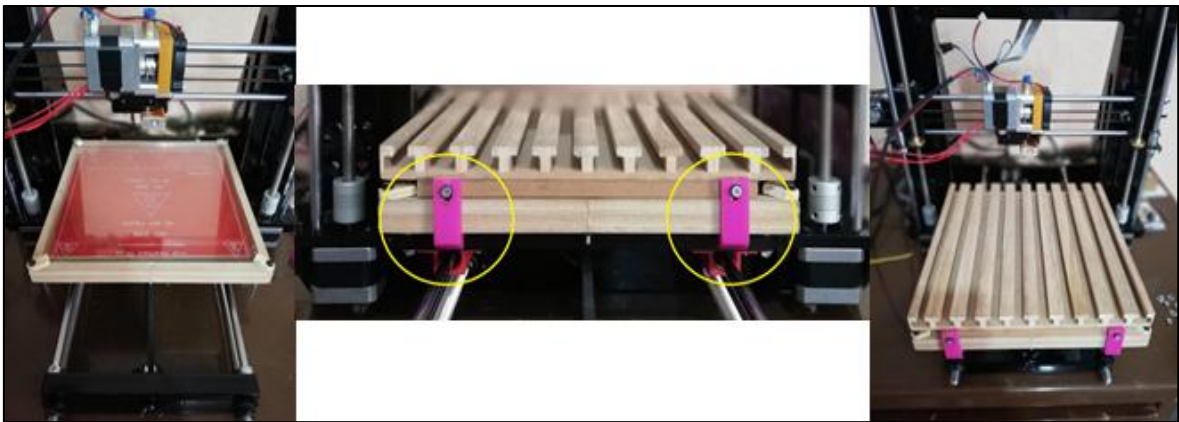
de fresado al igual que el de extrusión se realiza por medio del ajuste de tornillos. En la figura 4.64 se puede apreciar el cambio de cabezal en la infraestructura de la impresora.



**Figura. 4.64. Intercambio de cabezal.**

Fuente: Elaborado por el autor.

Segundo, la colocación de la plataforma de (corte/grabado) sobre la plataforma de impresión no representa mayor esfuerzo, simplemente se la coloca en posición y es ajustada por los acoples de sujeción diseñados en Solidworks por medio de tornillos. A continuación en la figura 4.65 se puede apreciar la colocación de la plataforma para (corte/grabado).



**Figura. 4.65. Colocación de plataforma para fresado.**

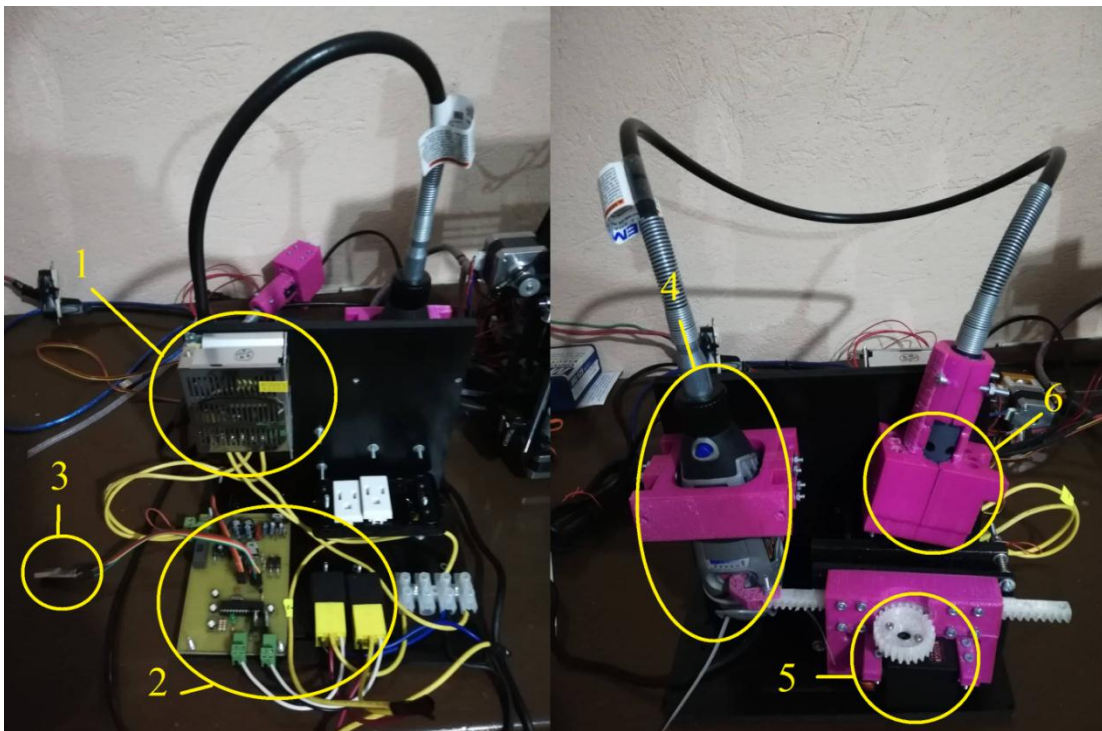
Fuente: Elaborado por el autor.

#### 4.4.2 Detalle técnico

La estructura del módulo que contiene tanto a la herramienta giratoria como a los circuitos de control eléctrico y electrónico, cuenta con la siguiente descripción técnica:

1. Fuente de alimentación CNC: 120 AC
2. Circuito de control
3. Puerto USB
4. Dremel
5. Servomotor MG995
6. Cabezal de fresado

En la figura 4.66 se puede observar cada uno de los puntos antes descritos.



**Figura. 4.66. Estructura del módulo.**

Fuente: Elaborado por el autor.

Además es importante conocer los siguientes detalles para el correcto funcionamiento del módulo de fresado CNC.

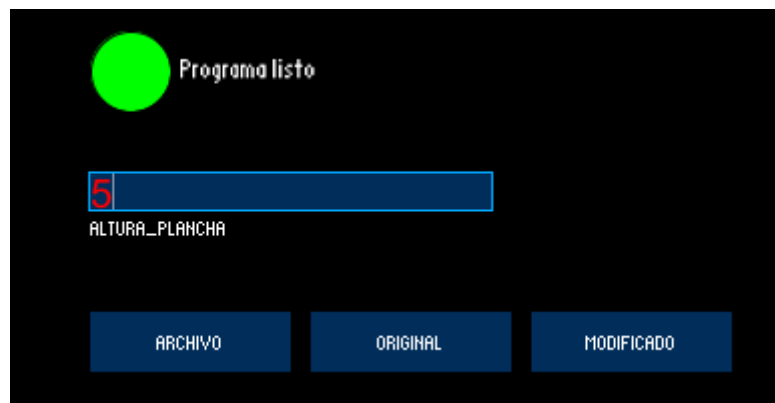


- El sistema está adecuado para funcionar en la infraestructura de una impresora 3D.
- Diámetro de fresa máximo: 3mm para ser colocado en el mandril del flex.
- Necesidad de dos puertos tipo USB (módulo CNC e impresora 3D).
- El módulo CNC utiliza un enchufe para AC.

#### 4.4.3 Reinterpretación del código G

La reinterpretación del código G se presenta al momento de ingresar el archivo “.ngc” proveniente del software Inkscape en la interfaz de la fresadora CNC, en donde, es leído y posteriormente se reemplazan los siguientes valores: primero, la línea que contiene el código “G00 Z\*\*\*\*\*” es reemplazado por los valores ingresados en la casilla de la interfaz llamada “Altura de la plancha”, luego, a este valor se le resta la línea de código (Pass at depth -\*\*\*\*) la cual contiene un número negativo que representa la cantidad de penetración que se realizara en un intervalo de movimientos determinados.

A continuación se presenta el análisis de un caso de reinterpretación de código para una figura geométrica (cuadrado). Primero en la interfaz de la CNC se ingresa el grosor de la plancha en (mm), para este caso será de 5mm, figura 4.67.



**Figura. 4.67. Altura de la plancha.**

Fuente: Elaborado por el autor.

Segundo, se selecciona el archivo a modificarse y el archivo modificado dando clic en el botón “archivo”.

Tercero, dando clic en los botones “original” y “modificado” de la interfaz de la CNC se puede constatar la reinterpretación del código G. En la figura 4.68, se pueden observar los valores obtenidos en el archivo original proveniente del software Inkscape, poniendo mayor énfasis en las líneas de código “G00 Z\*\*\*\*\*” y (Pass at depth -\*\*\*\*), ya que estas serán las líneas modificadas posteriormente.

```
%
(Header)
(Generated by gcodetools from Inkscape.)
(Using default header. To add your own header create file "header" in the output
M3
(Header end.)
G21 (All units in mm)
(Pass at depth -0.25)
(Start cutting path id: rect815)
(Change tool to cylindrical cutter)
G00 Z3.000000
G00 X16.630859 Y44.601560

G01 Z-0.250000 F100.0(Penetrate)
G01 X16.630859 Y17.386720 Z-0.250000 F400.000000
G01 X48.380859 Y17.386720 Z-0.250000
G01 X48.380859 Y44.601560 Z-0.250000
G01 X16.630859 Y44.601560 Z-0.250000
G00 Z3.000000

(End cutting path id: rect815)
(Pass at depth -0.5)
(Start cutting path id: rect815)
(Change tool to cylindrical cutter)
G00 Z3.000000
G00 X16.630859 Y44.601560

G01 Z-0.500000 F100.0(Penetrate)
G01 X16.630859 Y17.386720 Z-0.500000 F400.000000
G01 X48.380859 Y17.386720 Z-0.500000
G01 X48.380859 Y44.601560 Z-0.500000
G01 X16.630859 Y44.601560 Z-0.500000
G00 Z3.000000

(End cutting path id: rect815)
(Pass at depth -0.75)
(Start cutting path id: rect815)
(Change tool to cylindrical cutter)
G00 Z3.000000
G00 X16.630859 Y44.601560
```

Figura. 4.68. Archivo original.

Fuente: Elaborado por el autor.

Para este instante se puede observar que: G00 Z3.000000 y (Pass at depth -0,25) siendo este último el valor que se va restando para cada bloque de movimientos. En la figura 4.69, se puede observar como estos valores cambian siendo que, G00 Z3.000000 es reemplazado por G00 Z5.0, valor que fue ingresado previamente (5mm), así mismo el valor de (Pass at depth -0,25) se mantiene ya que será restado a cada bloque de movimientos.

```

output: Bloc de notas
Archivo Edición Formato Ver Ayuda
%
(Header)
(Generated by gcodetools from Inkscape.)
(Using default header. To add your own header create file "header" in the output
M3
(Header end.)
G21 (All units in mm)
(Pass at depth -0.25)
(Start cutting path id: rect815)
(Change tool to cylindrical cutter)
G00 Z5.0
G00 X16.630859 Y44.601560

G01 Z4.75 F100.0(Penetrare)
G01 X16.630859 Y17.386720 Z4.75 F400.000000
G01 X48.380859 Y17.386720 Z4.75
G01 X48.380859 Y44.601560 Z4.75
G01 X16.630859 Y44.601560 Z4.75
G00 Z5.0

(End cutting path id: rect815)

(Pass at depth -0.5)
(Start cutting path id: rect815)
(Change tool to cylindrical cutter)
G00 Z5.0
G00 X16.630859 Y44.601560

G01 Z4.5 F100.0(Penetrare)
G01 X16.630859 Y17.386720 Z4.5 F400.000000
G01 X48.380859 Y17.386720 Z4.5
G01 X48.380859 Y44.601560 Z4.5
G01 X16.630859 Y44.601560 Z4.5
G00 Z5.0

(End cutting path id: rect815)

(Pass at depth -0.75)
(Start cutting path id: rect815)
(Change tool to cylindrical cutter)

G00 Z5.0
G00 X16.630859 Y44.601560

```

Figura. 4.69. Archivo modificado.

Fuente: Elaborado por el autor.

## CONCLUSIONES

Una vez culminado el proyecto de titulación, se han llegado a las siguientes conclusiones:

- Para el diseño del sistema de fresado se puede concluir que, los elementos diseñados en Solidworks, se acoplaron cómodamente a la estructura de la impresora, haciendo posible el intercambio de funcionalidades de una manera fácil y cómoda, además, el modulo creado para contener la tarjeta electrónica, el circuito eléctrico y el servomotor junto con el Dremel, es compacto y práctico al momento de ser trasladado. Por otra parte, la utilización de un accesorio para extender la boquilla del Dremel hasta la estructura de la impresora, hizo más liviano el cabezal de fresado, liberando así de peso extra en el carrete de movimiento en el eje X.
- Por otra parte el software diseñado para la comunicación CAD fresadora, cumplió con todas las prestaciones establecidas en los objetivos específicos, como: movimiento de los ejes X, Y, Z, colocación de dichos ejes a su home, control de encendido/apagado y velocidad del Dremel, envío de código G, visualización del posicionamiento de los ejes X, Y, Z, cargar archivos de corte/grabado, reinterpretación de código G y finalmente enviar a ejecutar el corte/grabado, todo desde una misma interfaz.
- Así mismo la implementación del sistema, tanto a nivel físico como de software, estableció todos los parámetros estipulados en los objetivos del proyecto, haciendo todo lo posible para facilitar la migración de una funcionalidad a otra, pensando siempre en la comodidad del usuario.
- Además en la actualidad, la tecnología está cada vez más al alcance de la mano y la implementación del control de desplazamiento direccional, para la fresadora a través de un control remoto, aprovechó las prestaciones que brinda una de las tecnologías que se encuentra en la actualidad más cerca del ser humano, el cual es, el smartphone. En el transcurso de la investigación para la generación del contro

remoto, en búsqueda de cumplir con las funciones estipuladas en el objetivo de este proyecto, se fueron creando nuevas inquietudes y una sed de conocimiento, por lo cual se exploraron ámbitos adicionales, que generaron una pequeña aportación extra para el presente trabajo de titulación.

- Finalmente la validación fue sin duda un proceso muy importante para la construcción del proyecto, ya que, se la realizó a nivel de software y hardware para comprobar el correcto funcionamiento de cada uno de estos ámbitos.

## RECOMENDACIONES

- Para el diseño de acoples o elementos en el Solidworks, es importante tener muy en cuenta que se debe contar con las medidas adecuadas de cada figura a diseñar, preferentemente en mm, al momento de imprimirlas siempre estar pendiente de la fabricación de las piezas para evitar cualquier desperfecto en ellas.
- Para desarrollar el software de comunicación CAD – fresadora, es importante investigar y apoyarse en las prestaciones básicas que brindan otro tipo de software parecido al que se va a implementar.
- Al momento de implementar el diseño del fresado y software en la impresora volumétrica existente, es importante analizar como este sistema se está acoplando a la infraestructura para así poder corregir cualquier desperfecto.
- Es importante comenzar con la programación a la cual se va a conectar una aplicación, ya que así se podrán definir parámetros que facilitarán el proceso de su diseño.
- Por otra parte, para la ejecución de la validación del proyecto, se recomienda contar con todo tipo de herramientas, repuestos, y tiempo necesario para realizar este trabajo con toda tranquilidad.

## BIBLIOGRAFÍA

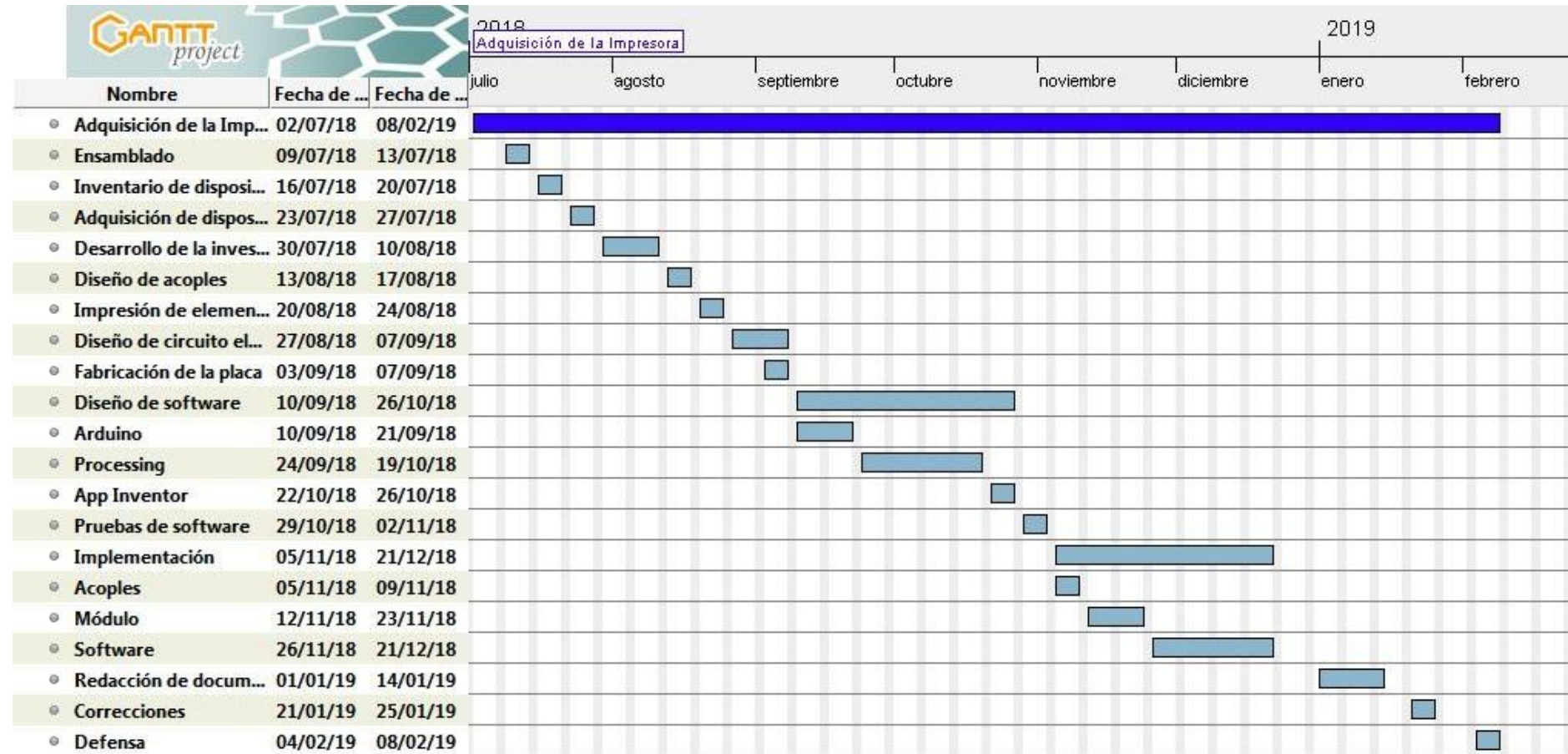
- Aguilar, J. (Sin fecha): *11-Fresado*, Recuperado de [http://biblio3.url.edu.gt/Libros/2013/pro\\_ma/11.pdf](http://biblio3.url.edu.gt/Libros/2013/pro_ma/11.pdf)
- Atmel (2016): *ATmega328/P: Datasheet complete*. Recuperado de [https://cdn.sparkfun.com/assets/c/a/8/e/4/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P\\_Datasheet.pdf](https://cdn.sparkfun.com/assets/c/a/8/e/4/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf)
- Boylestad, R. y L. Nashelsky (2009): *Electrónica: teoría de circuitos y dispositivos electrónicos*, México, Pearson Educación, Décima edición.
- Dávila, M. (Sin fecha): *Investigación tecnológica: Extractado del libro “La Investigación tecnológica, Limusa Noriega Editores, García Córdoba”*, Colombia, Uniminuto. Recuperado de <http://www.e-logicasoftware.com/tutoriales/conferencias/investigacion-tecnologica-y-aplicada/160525-investigacion-tecnologica-manuel-davila.doc>
- García, F. (2018): *La investigación tecnológica: Investigar, idear e innovar en ingenierías y ciencias sociales*, México, Editorial Limusa, Segunda edición.
- GEEETECH (2016): *User’s Manual of GT2560*, Recuperado de <https://www.geeetech.com/Documents/Users%20Manual%20of%20GT2560.pdf>
- Gordón, A. y Agencias (s/f): *La impresión 3D conquista rápidamente nuevas áreas*, El Comercio. Recuperado de <https://www.elcomercio.com/actualidad/impresion-3d-conquista-rapidamente-nuevas.html>
- Herrera, L. (2018): *Implementar un sistema de manejo lineal para una mesa de control numérico computarizado*, Tesis de pregrado, Universidad Tecnológica Israel, Ecuador.
- Jiménez, R. (Sin fecha): *Ingeniería de manufactura: Control Numérico por Computadora (CNC)*, Recuperado de <http://materias.fi.uba.ar/7565/U4-control-numerico-por-computadora.pdf>
- Lascano, R. (2018): *Diseño y construcción de una máquina de corte y grabado láser en materiales de baja densidad para un centro de diseño gráfico*, Tesis de pregrado, Universidad Tecnológica Israel, Quito – Ecuador.

- Marlin P. Jones & Assoc., INC. (Sin fecha): *31150-MP MG995 High Speed Servo Actuator*, Lake Park, Florida. Recuperado de <https://www.mpja.com/download/31150mp.pdf>
- Martínez, A. y A. Salinas (2017): *Metodología de la investigación / MEI401 (Guía didáctica del estudiante)*, México, Universidad Latinoamericana, S.C.
- Palella, S. y F. Martins (2012): *Metodología de la investigación cuantitativa*, Caracas, Fondo editorial de la Universidad Pedagógica Experimental Libertador, Tercera edición, Primera reimpresión.
- Schvab, L. (2011): *Máquinas y herramientas: Guía didáctica*, Buenos Aires – Argentina, Ministerio de Educación.
- Semiconductor Components Industries, LLC. (2014): *TIP120, TIP121, TIP122 (NPN); TIP125, TIP126, TIP127 (PNP): Plastic Medium-Power Complementary Silicon Transistors*. Recuperado de <https://www.onsemi.com/pub/Collateral/TIP120-D.PDF>
- Texas Instruments Incorporated (1997–2016): *LM317 3-Terminal Adjustable Regulator*. Recuperado de <http://www.ti.com/lit/ds/slvs044x/slvs044x.pdf>
- Torres, J. (Sin fecha): *Diseño asistido por ordenador*, Granada, Universidad de Granada. Recuperado de <https://lsi.ugr.es/~cad/teoria/Tema1/RESUMENTEMA1.PDF>
- Turriago, A. (2018): *Implementación de una tarjeta de control para una mesa mediante Control Numérico Computarizado*, Tesis de pregrado, Universidad Tecnológica Israel, Ecuador.
- Yuni, J. y C. Urbano (2014): *Técnicas para Investigar: Recursos Metodológicos para la Preparación de Proyectos de Investigación*, Argentina, Editorial Brujas, Segunda edición.



# **ANEXOS**

## ANEXO 1. Cronograma de actividades

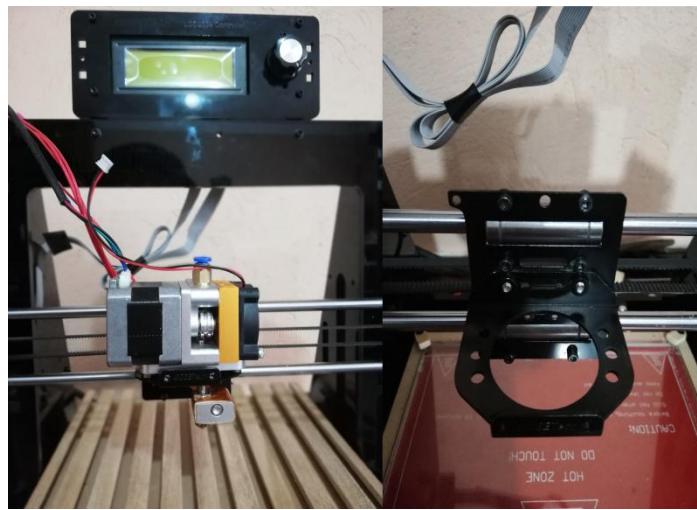


## ANEXO 2. Montaje de elementos diseñados en Solidworks

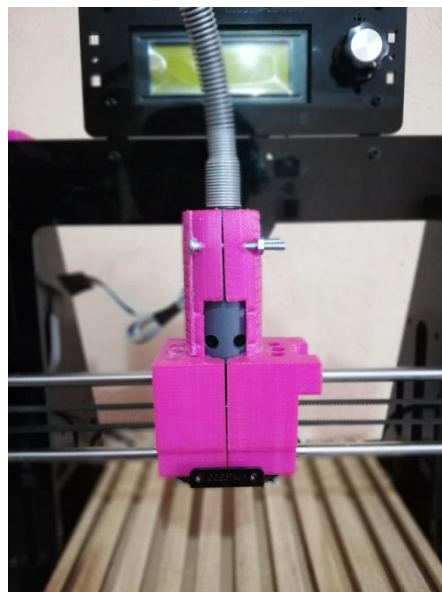
Como se observó en el capítulo 4, se diseñaron varios elementos para que sea posible el proceso de (corte/grabado) utilizando la misma estructura de una impresora 3D. A continuación se detalla su montaje sobre la impresora y en el módulo para la CNC.

### Montaje del cabezal de la fresadora sobre la infraestructura de la impresora 3D.

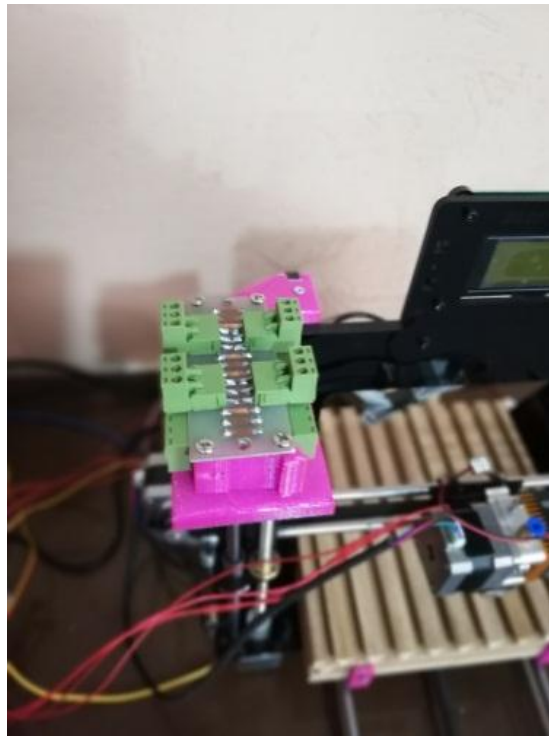
- Primero es necesario retirar el cabezal de extrusión.



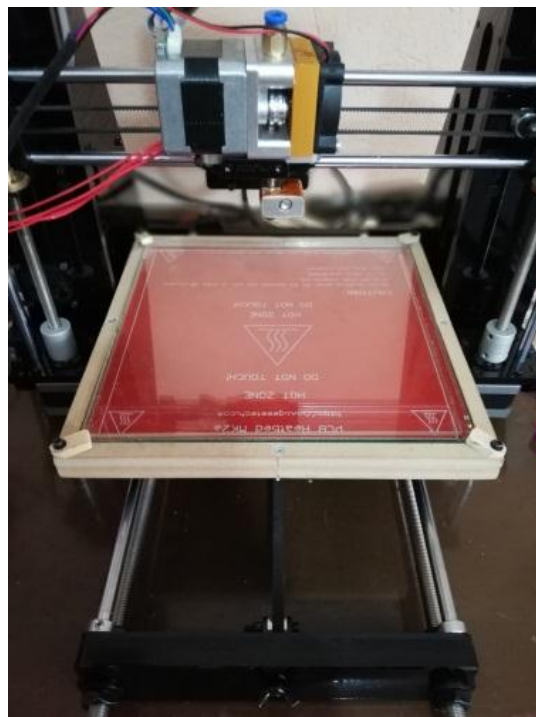
- Segundo se coloca en su lugar el cabezal de fresado (figura 4.2 - 4.3).



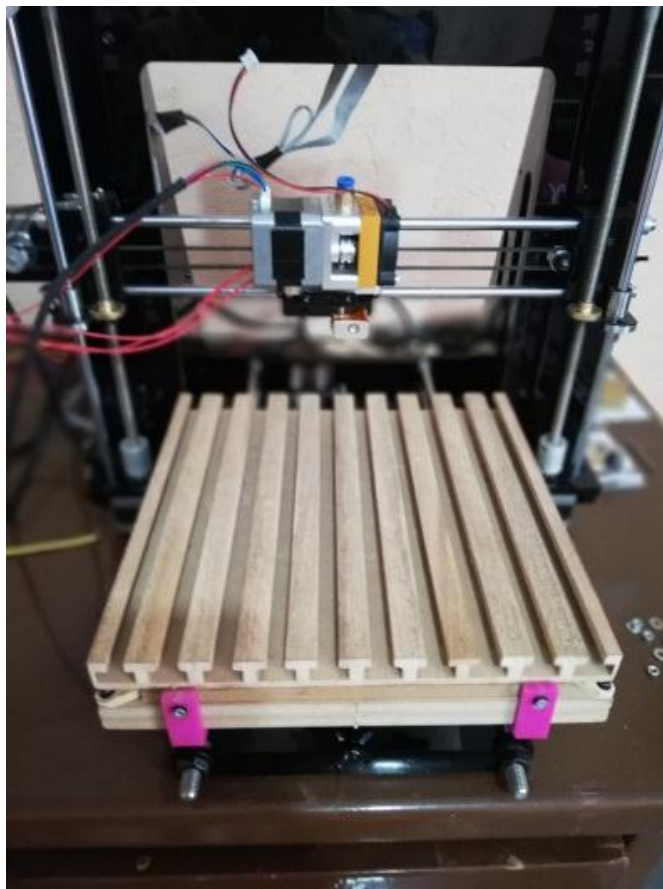
**Montaje del acople para borneras en la estructura de la impresora 3D (figura 4.4).**



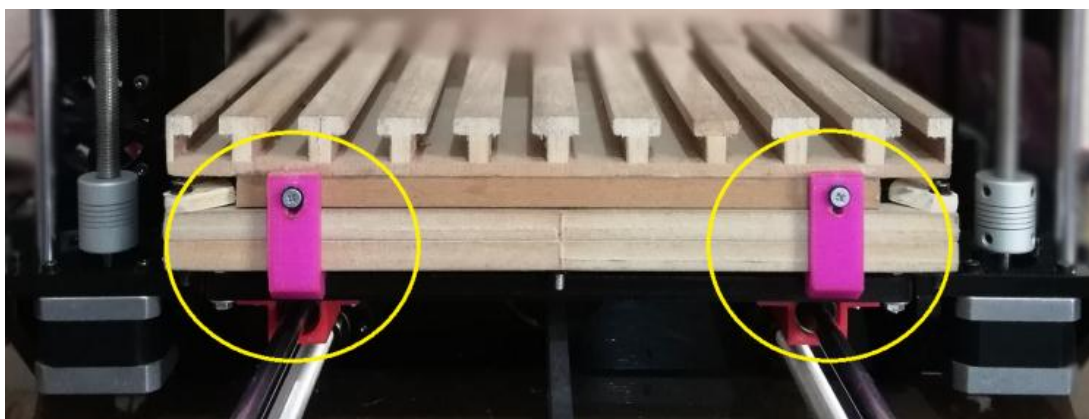
**Nueva plataforma colocada para ejecutar las funciones de la impresora 3D (figura 4.5).**



**Plataforma para corte y grabado sobre la estructura de la impresora 3D (figura 4.6).**

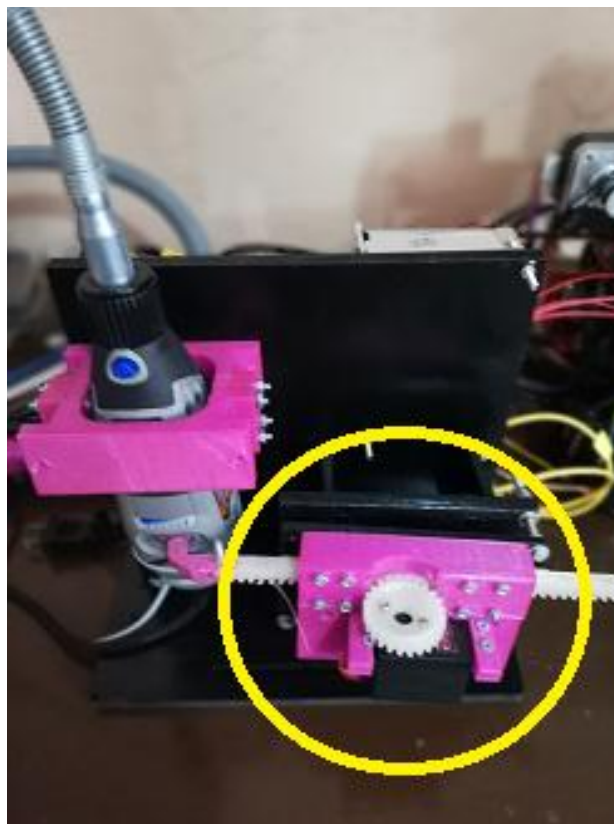


**Colocación de acoples para fijar la plataforma de corte/grabado sobre la plataforma de impresión (figura 4.7).**

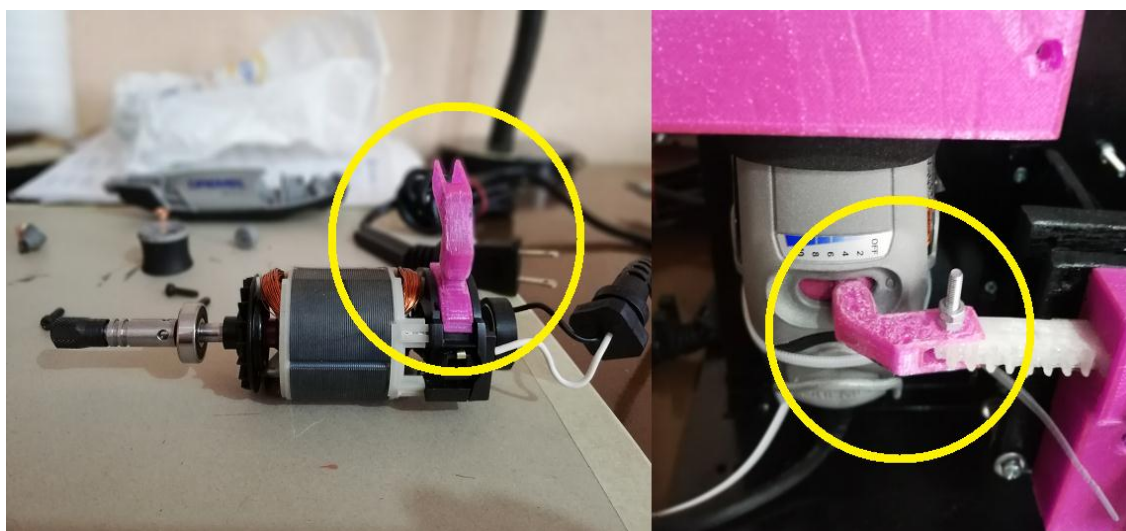




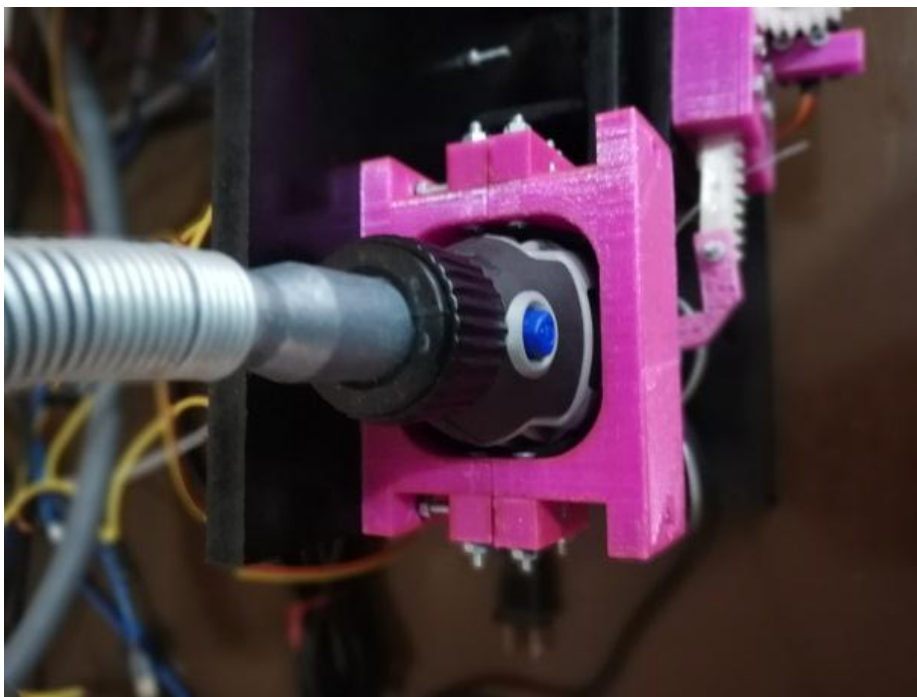
**Montaje de la plataforma de sujeción para el servomotor en el módulo de la CNC (figura 4.8).**



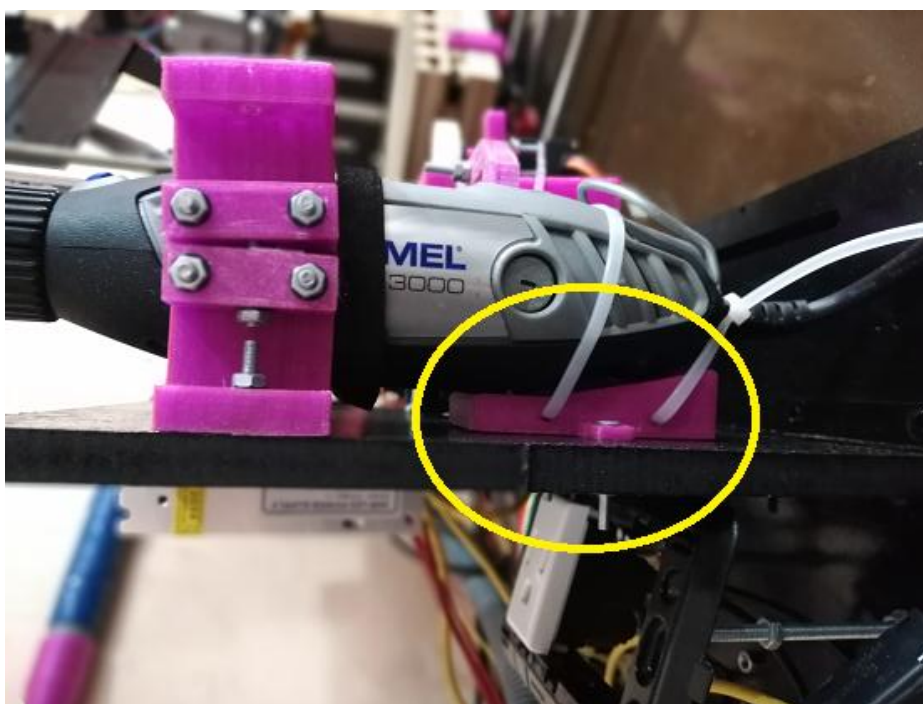
**Montaje de la perilla modelada en Solidworks dentro del Dremel (figura. 4.9).**



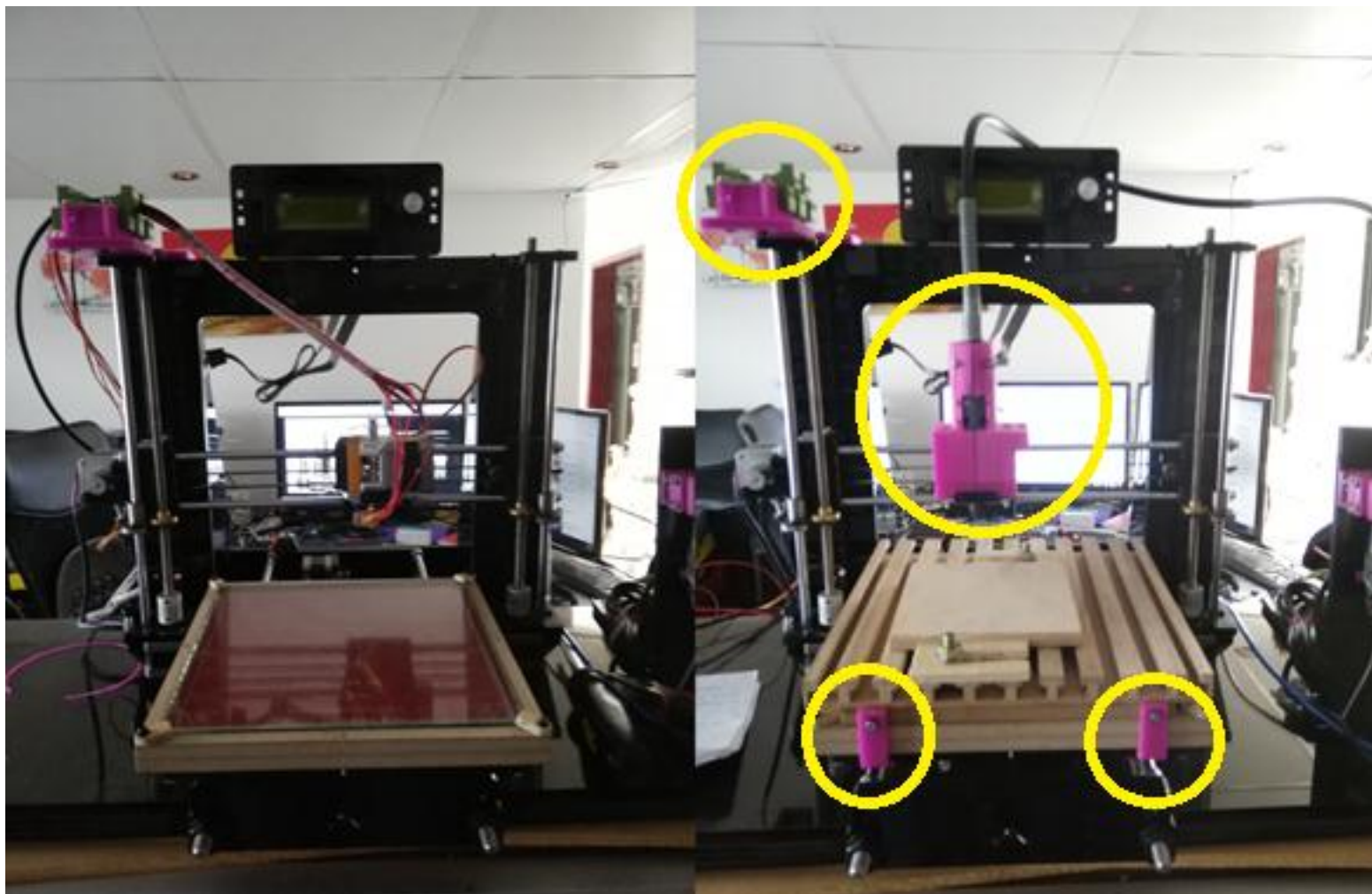
**Empleo de soportes superiores en la estructura del módulo para la sujeción del Dremel (figura. 4.10).**



**Empleo de soporte inferior en la estructura del módulo para la sujeción del Dremel (figura. 4.11).**



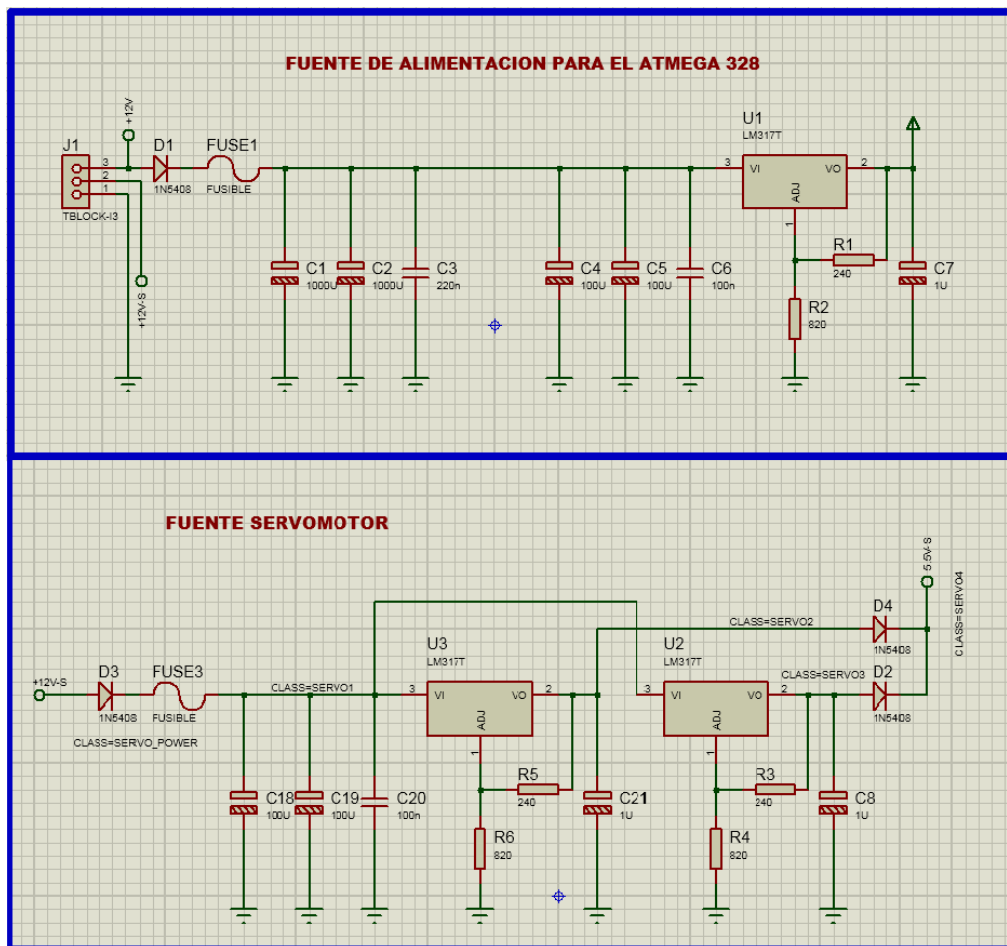
**ANEXO 3. Impresora 3D antes y después de la colocación de acoples.**



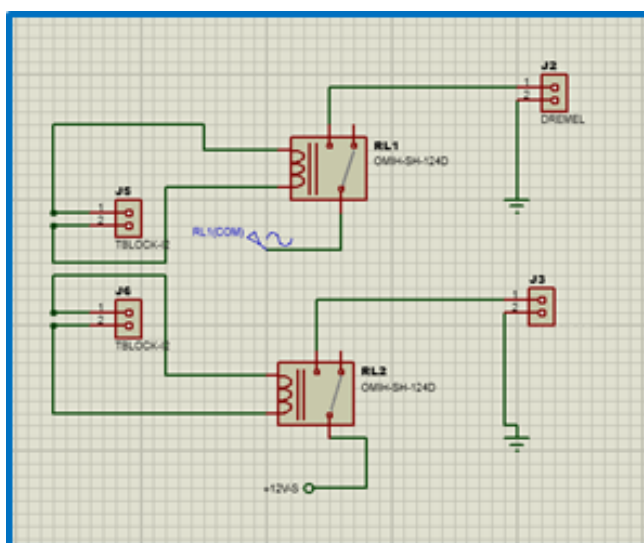


## ANEXO 4. Diagramas electrónicos y eléctricos

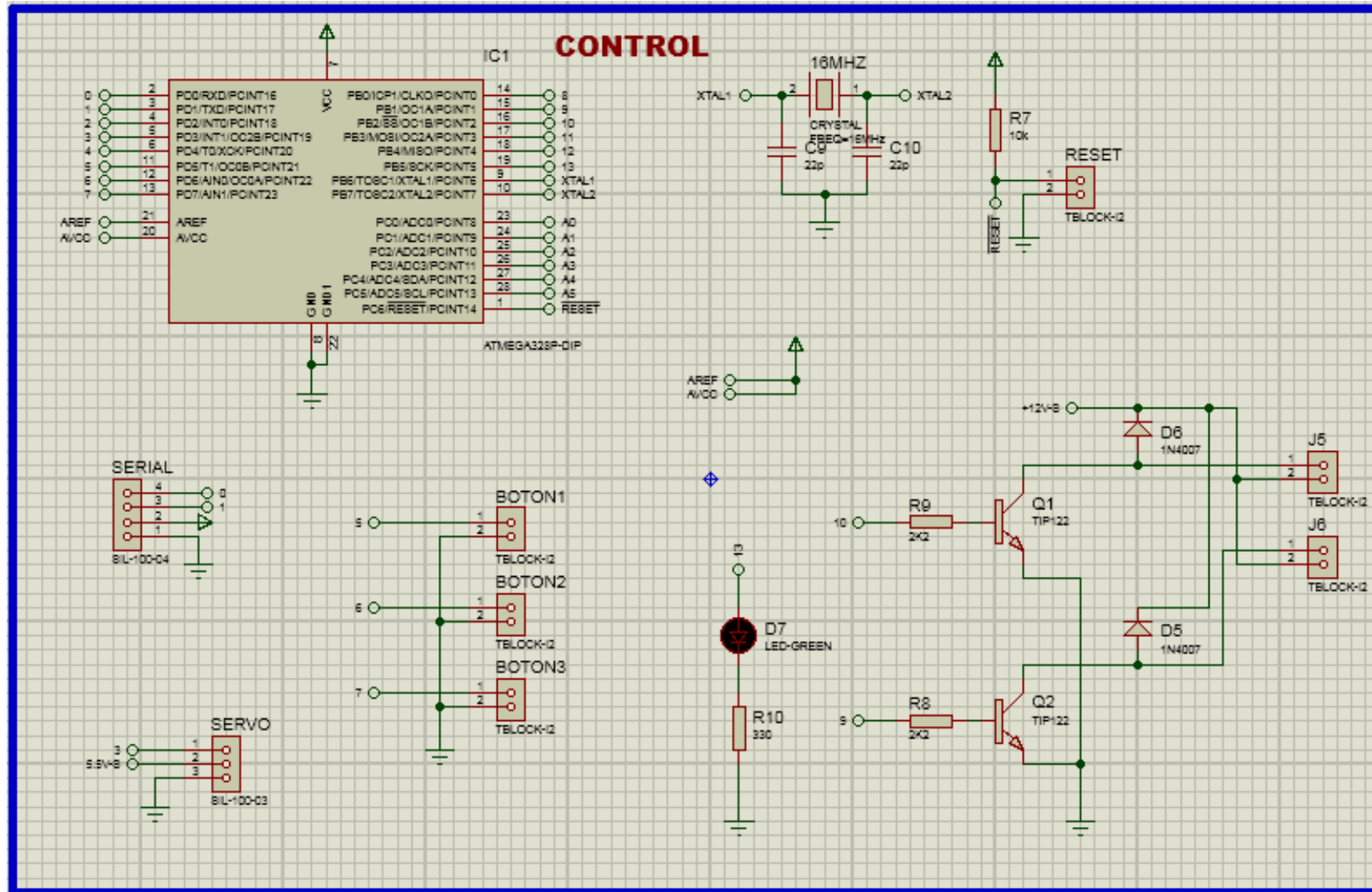
### Diagrama de alimentación:



### Diagrama eléctrico:



Circuito de control:



## ANEXO 5. Programación en Arduino

### Programación bajo la (IDE) Arduino

- Inclusión de las bibliotecas a ser utilizadas: MsTimer2.h y Servo.h.

```
#include <MsTimer2.h>
#include <Servo.h>
```

- Definición de constantes y sus respectivos puertos:

```
#define rele1 9
#define rele2 10
#define led 13
#define servo 3
#define boton1 5
#define boton2 6
#define boton3 7
```

- Definición de: salidas, entradas, estados altos y bajos.

```
#define output OUTPUT
#define input INPUT_PULLUP
#define ON HIGH
#define OFF LOW
```

- Definición de variables.

```
Servo myservo;
boolean flagRele = false;
String inputString = "";
boolean stringComplete = false;
int ledState = LOW;
int val =0;
```

- Inicialización de variables, bibliotecas y funciones.

```
void setup() {

    inicializar_serial();
    configuracion_pines_salida();
    configuracion_pines_entrada();
```

```

    MsTimer2::set(500, timer_led);
    MsTimer2::start();
    myservo.attach(servo);
    myservo.write(45);
}

```

- Bucle de repetición: en el bucle se actualizan los valores dentro de las funciones de los comandos seriales y las funciones de los botones.

```

void loop() {
    comandos_Seriales();
    funciones_botones();
}

```

- Configuración de pines de entrada:

```

void configuracion_pines_entrada(){

    pinMode(boton1,input);
    pinMode(boton2,input);
    pinMode(boton3,input);
}

```

- Configuración de pines de salida:

```

void configuracion_pines_salida(){

    pinMode(rele1,output);
    pinMode(rele2,output);
    pinMode(led,output);
    pinMode(servo,output);
}

```

- Configuración del puerto serial: en él se establece la velocidad de la comunicación, así como la cantidad de memoria reservada para el flujo de información del puerto serial, y finalmente se imprime un mensaje para saber que el puerto esta inicializado.

```

void inicializar_serial(){

    Serial.begin(9600);
    inputString.reserve(200);
    Serial.println("*****");
    Serial.println("*Control iniciado*");
}

```

```

    Serial.println("*****");
}

```

- Ejecución del evento serial. El evento serial se encarga de recibir, almacenar y procesar la información que llega al puerto serial.

```

void serialEvent() {

    while (Serial.available()) {
        char inChar = (char)Serial.read();
        inputString += inChar;
        if (inChar == '\r') {
            stringComplete = true;
        }
    }
}

```

- Inicialización del contador, el cual es aplicado para controlar el encendido y apagado de un led que mostrará si el Atmega 328 se encuentra en funcionamiento.

```

void timer_led(void){

    if (ledState == LOW) {
        ledState = HIGH;
    }
    else {
        ledState = LOW;
    }
    digitalWrite(led, ledState);
}

```

- Función del botón, estos botones se encargarán del incremento o disminución de la velocidad del Dremel por medio de la manipulación del estado de un servomotor.

```

void funciones_botones(){

    if(!digitalRead(boton1)){
        while(!digitalRead(boton1)){delay(1);}
        val=val+1;
        if(val >= 120){val=120;}
        myservo.write(val);
        Serial.println(val);
        delay(100);
    }
}

```

```

if(!digitalRead(boton2)){
  while(!digitalRead(boton2)){delay(1);}
  val=val-1;
  if(val <= 30){val=30;}
  myservo.write(val);
  Serial.println(val);
  delay(100);
}
if(!digitalRead(boton3)){
  while(!digitalRead(boton3)){delay(1);}
  Serial.println("aca");
  if(digitalRead(rele2) == ON){
    digitalWrite(rele2,OFF);
    delay(500);}
  else{digitalWrite(rele2,ON);
    delay(500);}
  delay(500);
}
}
}

```

- Ejecución de los comandos en la serial, aquí se ejecutan las palabras claves asignadas a cada una de las funciones del programa, como por ejemplo: DON hace referencia al estado encendido del Dremel o DOFF hacer referencia al estado apagado del Dremel, por otra parte, para el control de velocidad, se rastrea la cadena de datos de llegada al puerto serial en espera de la letra S, en esta se extraerá el valor de la velocidad transformándolo a un número entero y asignándole un valor entre 45 y 120 grados.

```

void comandos_Seriales(){

  if(stringComplete){
    Serial.println(inputString);
    inputString.trim();

    if(inputString.equals("DON")){
      digitalWrite(rele1,ON);
    }

    if(inputString.equals("DOFF")){
      digitalWrite(rele1,OFF);
    }

    if(inputString.substring(0,1)=="S"){
      String auxs = inputString.substring(1);
      int auxi = auxs.toInt();

```

```
        if(auxi >= 0 && auxi < 101){
            auxi= map(auxi, 0, 100, 45, 120);
            Serial.println(auxi);
            myservo.write(auxi);
        }
    }
    inputString = "";
    stringComplete = false;
}
}
```

## ANEXO 6. Programación en Processing

- **Importación de librerías.**

```
import processing.serial.*;
import controlP5.*;
import static javax.swing.JOptionPane.*;
```

- **Objetos** a ser utilizados en la programación. Los objetos son los encargados de brindar un estado y comportamiento, estos serán empleados para generar diferentes funciones en el programa.

```
Serial impresora, variador, bluetooth;
ControlP5 cp5;
Slider velo;
Textfield tfcomando;
RadioButton r1;
Textarea myTextarea;
```

- **Especificación de variables tipo entero.**

```
int whichKey = -1;
int inByte = -1;
int VELOCIDAD = 45;
int inicio=0;
int vp0=1; //bluetooth vp0
int vp1=0; //impresora vp1
int vp2=2; //variador vp2
int INTRO=0;
int index = 0;
int PosIZ = -1;
int PosDE = -1;
int PosAD = -1;
int PosAT = -1;
int PosAR = -1;
int PosAB = -1;
int PosHXY = -1;
int PosHZ = -1;
int PosAS = -1;
int PosDR = -1;
int PosJ0 = -1;
int PosJ1 = -1;
int PosJ2 = -1;
```



```

int PosJ3 = -1;
int PosVL = -1;
int PosOK = -1;
int PosGComand = -1;
int PosIntro = -1;
int contador =0;
int PosCMD=-1;
int G00Z=0,gg=0;

```

- **Especificación de variables tipo flotante.**

```

float vr=0.0;
float px=0.00;
float py=0.00;
float pz=0.00;
float seguridad=0.00,seguridadt=0.00;
float ats=0.00;
float paso = 0.00;
float altura=0.00;

```

- **Especificación de variables tipo string.**

```

String texto="";
String texto2="";
String textValue = "";
String Scomando = "";
String auxt2 = "";
String[] lines,lines1,sSalida;
String Archivo="";
String Archivo2="";
String[] input;
String DirectorioOutput="";

```

- **Especificación de variables tipo boolean.**

```

boolean flagCompleto = false;
boolean flagCirculoVerde=false;
boolean finicio=false;
boolean finicio2=false;
boolean fd=false;
boolean fa=false;
boolean fready=false;

```

- **Especificación de salida de texto.**

```
PrintWriter output,output2;
```

- **Inicialización de variables**, objetos y funciones.

```
void setup() {
    size(800, 600);
    ObjetosCP5();
    ObjetosSerial();
    lines = loadStrings("cuadrado.txt");
    output = createWriter("data/app.gnc");
    output2 = createWriter("data/app1.txt");
}
```

- **Bucle de repetición:** en el bucle se actualizan los valores de la ventana a desplegarse, así como la información brindada de las posiciones en X, Y, Z, también las funciones del puerto serial son actualizadas al igual que un aviso de inicio del programa.

```
void draw() {

    background(0);

    if(fready){
        fill(0,255,0);
        ellipse(50,50,50,50);
        fill(255,255,255);
        text("Programa listo",25,45);
    }

    myTextarea.setText( "X: "+px+"\n'
                        +"Y: "+py+"\n'
                        +"Z: "+pz+"\n'
                        );

    if(flagCompleto == true){
        validacion_M114();
        flagCompleto=false;
    }

    if(!finicio2){
        inicio++;
    }
}
```

```

    }

    if(inicio>200){
        finicio=true;
        finicio2=true;
        inicio=0;
        println("inicio");
        fready=true;
    }
}

```

- **Configuración de inicio para el evento serial.** Asignación de nombres a los puertos y velocidad de comunicación.

```

void ObjetosSerial(){

    printArray(Serial.list());
    String portName = Serial.list()[vp1];
    String portName1 = Serial.list()[vp2];
    String portName2 = Serial.list()[vp0];
    impresora = new Serial(this, portName, 250000);
    variador = new Serial(this, portName1, 9600);
    bluetooth = new Serial(this, portName2, 9600);
}

```

- **Ejecución del evento serial.** El evento serial se encarga de recibir, almacenar y procesar la información que llega al puerto serial.

```

void serialEvent(Serial impresora) {

    inByte = impresora.read();
    texto += char(inByte);
    int pos =texto.indexOf("ok");
    if(pos>0){
        flagCompleto = true;
        texto2=texto;
        texto="";
        pos=0;
    }
}

```

- **Creación de objetos CP5:** en esta ventana se encuentran las configuraciones de cada uno de los objetos utilizados, así como: su color, tamaño, ubicación, etc.

```

void ObjetosCP5(){

    cp5 = new ControlP5(this);

    PFont font = createFont("arial",20);
cp5.addTextfield("ALTURA_PLANCHA")
    .setPosition(50,90)
    .setSize(200,20)
    .setFont(font)
    .setFocus(true)
    .setColor(color(255,0,0))
    ;

cp5.addButton("ARCHIVO")
    .setValue(0)
    .setPosition(50,160)
    .setSize(100,30)
    ;
cp5.addButton("ORIGINAL")
    .setValue(0)
    .setPosition(160,160)
    .setSize(100,30)
    ;
cp5.addButton("MODIFICADO")
    .setValue(0)
    .setPosition(270,160)
    .setSize(100,30)
    ;

velo=cp5.addSlider("VELOCIDAD")
    .setPosition(50,230)
    .setRange(0,100)
    .setValue(128)
    .setSize(200, 20)
    .setSliderMode(Slider.FLEXIBLE)
    ;

cp5.getController("VELOCIDAD")
.getCaptionLabel()
.align(ControlP5.LEFT, ControlP5.BOTTOM_OUTSIDE)
.setPaddingX(0);

cp5.addButton("enviar")
    .setValue(0)

```

```

        .setPosition(50,270)
        .setSize(80,25)
        ;
cp5.addTextfield("comandos")
    .setPosition(520,90)
    .setSize(200,40)
    .setFont(createFont("arial",20))
    .setFocus(true)
    .setColor(color(255,0,0))
    ;
cp5.addButton("Iniciar")
    .setValue(0)
    .setPosition(50,360)
    .setSize(100,50);
    //.setColor(color(255,0,0))
    ;
cp5.addButton("atras")
    .setPosition(467,243)
    .setImages(loadImage("arribaout.jpg"), loadImage("arribaover.jpg"),
loadImage("arribaok.jpg"))
    .updateSize()
    ;
cp5.addButton("adelante")
    .setPosition(467,390)
    .setImages(loadImage("abajout.jpg"), loadImage("abajoover.jpg"),
loadImage("abajook.jpg"))
    .updateSize()
    ;
cp5.addButton("derecha")
    .setPosition(537,316)
    .setImages(loadImage("derechaout.jpg"), loadImage("derechaover.jpg"),
loadImage("derechaok.jpg"))
    .updateSize()
    ;
cp5.addButton("izquierda")
    .setPosition(395,316)
    .setImages(loadImage("izquierdaout.jpg"), loadImage("izquierdaover.jpg"),
loadImage("izquierdaok.jpg"))
    .updateSize()
    ;
cp5.addButton("arriba")
    .setPosition(690,243)
    .setImages(loadImage("atrasout.JPG"), loadImage("atrasover.JPG"),
loadImage("atrasok.JPG"))
    .updateSize()

```

```

;
cp5.addButton("abajo")
    .setPosition(690,390)
    .setImages(loadImage("adelanteout.jpg"), loadImage("adelanteover.jpg"),
loadImage("adelanteok.jpg"))
    .updateSize()
;
r1 = cp5.addRadioButton("radioButton")
    .setPosition(400,500)
    .setSize(40,20)
    .setColorForeground(color(120))
    .setColorActive(color(255))
    .setColorLabel(color(255))
    .setItemsPerRow(5)
    .setSpacingColumn(50)
    .addItem("0.1",1)
    .addItem("1",2)
    .addItem("10",3)
    .addItem("100",4)
;
for(Toggle t:r1.getItems()) {
    t.getCaptionLabel().setColorBackground(color(255,80));
    t.getCaptionLabel().getStyle().moveMargin(-7,0,0,-3);
    t.getCaptionLabel().getStyle().movePadding(7,0,0,3);
    t.getCaptionLabel().getStyle().backgroundWidth = 45;
    t.getCaptionLabel().getStyle().backgroundHeight = 13;
}

myTextarea = cp5.addTextarea("txt")
    .setPosition(520,160)
    .setSize(200,200)
    .setFont(createFont("arial",12))
    .setLineHeight(14)
    .setColor(color(128))
    .setColorBackground(color(255,100))
    .setColorForeground(color(255,100));
;
myTextarea.setWidth(200);
myTextarea.setHeight(50);

cp5.addButton("HOME_XY")
    .setValue(0)
    .setPosition(480,330)
    .setSize(50,50)

```

```

;

cp5.addButton("HOME_Z")
  .setValue(0)
  .setPosition(703,330)
  .setSize(50,50)
;

cp5.addButton("DREMEL")
  .setValue(0)
  .setPosition(170,270)
  .setSize(80,25)
;
}

```

- **Validación M114**, en esta validación se extraen los datos que llegan al puerto serial provenientes de la impresora y del Bluetooth del teléfono, para luego ser procesados y enviados a los puertos de la impresora y del módulo (variador).

```

void validacion_M114(){

texto2=texto2.trim();
println(texto2.trim());
String s1="";
String s2="";
String s3="";
String s4="";
int is1 = 0;
int is2 = 0;
int is3 = 0;
int is4 = 0;

is1=texto2.indexOf("X:");
is2=texto2.indexOf("Y:");
is3=texto2.indexOf("Z:");
is4=texto2.indexOf("E:");

if(is1>0 && is1<is2 && is2<is3 && is3<is4){
  s1=texto2.substring(is1+2,is2);
  s2=texto2.substring(is2+2,is3);
  s3=texto2.substring(is3+2,is4);

  px=int(s1);
  py=int(s2);
  pz=int(s3);
}
}

```

```
texto2="";  
}  
  
PosIZ = texto2.indexOf("IZ");  
PosDE = texto2.indexOf("DE");  
PosAD = texto2.indexOf("AD");  
PosAT = texto2.indexOf("AT");  
PosAR = texto2.indexOf("AR");  
PosAB = texto2.indexOf("AB");  
PosHXY = texto2.indexOf("HXY");  
PosHZ = texto2.indexOf("HZ");  
  
PosAS = texto2.indexOf("AS");  
PosDR = texto2.indexOf("DR");  
PosJ0 = texto2.indexOf("J0");  
PosJ1 = texto2.indexOf("J1");  
PosJ2 = texto2.indexOf("J2");  
PosJ3 = texto2.indexOf("J3");  
PosVL = texto2.indexOf("VL");  
PosOK = texto2.indexOf("ok");  
PosGComand = texto2.indexOf("G");  
PosIntro = texto2.indexOf("\n");  
PosCMD = texto2.indexOf("CMD");  
  
String sIZ="";  
String sDE="";  
String sAD="";  
String sAT="";  
String sAR="";  
String sAB="";  
String sHXY="";  
String sHZ="";  
String sAS="";  
String sDR="";  
String sJ0="";  
String sJ1="";  
String sJ2="";  
String sJ3="";  
String sVL="";  
//String sOK="";  
String sGComand="";  
String Scanvas="";
```



```

if(PosCMD!=-1){

    println("PosCMD: "+PosCMD);
    println("PosIntro: "+PosOK);
    sGComand=texto2.substring(PosCMD+4,PosOK);
    sGComand = sGComand.trim();
    println("CMD:"+sGComand+" L: "+sGComand.length());
    println(sGComand);
    if(sGComand.equals("")){
        println("CMD:: "+sGComand);
        output.flush(); // Write the remaining data
        output.close(); // Finish the file
        contador=0;

        String x ="";
        String[] aux = loadStrings("data/app.gnc");

        while(contador<aux.length){
            println(aux[index]);
            x=aux[contador];
            println("A: "+ x);
            impresora.write(x);
            impresora.write(13);
            impresora.write(10);
            delay(400);
            contador++;
        }
    }else if(sGComand.equals("$")){
        output = createWriter("data/app.gnc");
    }
    else{

        impresora.write(sGComand);
        impresora.write(13);
        impresora.write(10);

        contador=0;
    }
    contador=0;
    texto2="";
    PosGComand = -1;
    PosCMD = -1;
    PosOK = -1;
}

```

```

}

if(PosGComand!=-1){

    println("PosGComand: "+PosGComand);
    println("PosGComand: "+PosIntro);

    try{
        sGComand=texto2.substring(PosGComand,PosIntro);
        Scanvas = Scanvas + sGComand;

    }catch(Exception e){
        println("error"+e);

    }
    int rango = Scanvas.length();
    println("rango: "+rango);

    if(rango>0){
        println("contador: "+contador);
        output.println(Scanvas);
        output2.println(Scanvas);

        rango =0;
    }

    texto2="";

}

if(PosIZ!=-1){

    println("PosIZ: "+PosIZ);
    sIZ=texto2.substring(PosIZ,PosIZ+2);
    sIZ=sIZ.trim();
    println("sIZ: "+sIZ);

    if(sIZ.equals("IZ")){
        println("Ilego: "+sIZ);
        texto2="";
        impresora.write("M114");
        impresora.write(13);
        impresora.write(10);
        px=px-vr;
    }
}

```

```

println("px: "+px);
String aux="G0 X"+px;
println(aux);
impresora.write(aux);
impresora.write(13);
impresora.write(10);
println("ok");
if(px<=0){px=0;println("px: "+px);}
}
}

if(PosDE!=-1){

println("PosDE: "+PosDE);
sDE=texto2.substring(PosDE,PosDE+2);
sDE=sDE.trim();
println("sDE: "+sDE);

if(sDE.equals("DE")){
println("llego: "+sDE);
texto2="";
impresora.write("M114");
impresora.write(13);
impresora.write(10);
px=px+vr;
println("px: "+px);
String aux="G0 X"+px;
println(aux);
impresora.write(aux);
impresora.write(13);
impresora.write(10);
println("ok");
if(px>200){px=200;println("px: "+px);}
}
}

if(PosAD!=-1){
println("PosAD: "+PosAD);
sAD=texto2.substring(PosAD,PosAD+2);
sAD=sAD.trim();
println("sAD: "+sAD);

if(sAD.equals("AD")){
println("llego: "+sAD);
}
}

```

```

texto2="";
impresora.write("M114");
impresora.write(13);
impresora.write(10);
py=py+vr;
if(py<=180){
    String aux="G0 Y"+py;
    println(aux);
    impresora.write(aux);
    impresora.write(13);
    impresora.write(10);
}
if(py>200){py=200;println("py: "+py);}
}
}

if(PosAT!=-1){

    println("PosAT: "+PosAT);
    sAT=texto2.substring(PosAT,PosAT+2);
    sAT=sAT.trim();
    println("sAT: "+sAT);

    if(sAT.equals("AT")){
        println("Ilego: "+sAT);
        texto2="";
        impresora.write("M114");
        impresora.write(13);
        impresora.write(10);
        py=py-vr;
        String aux="G0 Y"+py;
        println(aux);
        impresora.write(aux);
        impresora.write(13);
        impresora.write(10);
        if(py<=0){py=0;println("py: "+py);}
    }
}

if(PosAR!=-1){

    println("PosAR: "+PosAR);
    sAR=texto2.substring(PosAR,PosAR+2);
    sAR=sAR.trim();

```

```

println("sAR: "+sAR);

if(sAR.equals("AR")){
    println("llego: "+sAR);
    texto2="";
    impresora.write("M114");
    impresora.write(13);
    impresora.write(10);
    pz=pz+vr;
    println("pz: "+pz);
    String aux="G0 Z"+pz;
    println(aux);
    impresora.write(aux);
    impresora.write(13);
    impresora.write(10);
    println("ok");
    if(pz>200){pz=200;println("pz: "+pz);}
}
}

if(PosAB!=-1){

    println("PosAB: "+PosAB);
    sAB=texto2.substring(PosAB,PosAB+2);
    sAB=sAB.trim();
    println("sAB: "+sAB);

    if(sAB.equals("AB")){
        println("llego: "+sAB);
        texto2="";
        impresora.write("M114");
        impresora.write(13);
        impresora.write(10);
        pz=pz-vr;
        println("pz: "+pz);
        String aux="G0 Z"+pz;
        println(aux);
        impresora.write(aux);
        impresora.write(13);
        impresora.write(10);
        println("ok");
        if(pz<=0){pz=0;println("pz: "+pz);}
    }
}
}

```

```

if(PosHXY!=-1){

    println("PosHXY: "+PosHXY);
    sHXY=texto2.substring(PosHXY,PosHXY+3);
    sHXY=sHXY.trim();
    println("sHXY: "+sHXY);

    if(sHXY.equals("HXY")){
        println("llego: "+sHXY);
        texto2="";
        String HXY="G28 X Y";
        println(HXY);
        impresora.write(HXY);
        impresora.write(13);
        impresora.write(10);
    }
}

if(PosHZ!=-1){

    println("PosHZ: "+PosHZ);
    sHZ=texto2.substring(PosHZ,PosHZ+2);
    sHZ=sHZ.trim();
    println("sHZ: "+sHZ);

    if(sHZ.equals("HZ")){
        println("llego: "+sHZ);
        texto2="";
        String HZ="G28 Z";
        println(HZ);
        impresora.write(HZ);
        impresora.write(13);
        impresora.write(10);
    }
}

if(PosDR!=-1){
    println("PosDR: "+PosDR);
    sDR=texto2.substring(PosDR,PosDR+2);
    sDR=sDR.trim();
    println("sDR: "+sDR);

    if(sDR.equals("DR")){

```

```

println("Ilego: "+sDR);
texto2="";
if(fd==true){
  delay(100);
  fd=false;
  String sd="DON";
  println(sd);
  impresora.write("M117 dremel on");
  impresora.write(13);
  impresora.write(10);

  variador.write(sd);
  variador.write(13);
  variador.write(10);

  bluetooth.write(sd);
  bluetooth.write(13);
  bluetooth.write(10);

  println("ok");
}else{
  delay(100);
  fd=true;
  String sd="DOFF";
  println(sd);
  impresora.write("M117 dremel off");
  impresora.write(13);
  impresora.write(10);

  variador.write(sd);
  variador.write(13);
  variador.write(10);

  bluetooth.write(sd);
  bluetooth.write(13);
  bluetooth.write(10);

  println("ok");

}
}
}

if(PosJ0!=-1){

```

```
println("PosJ0: "+PosJ0);
sJ0=texto2.substring(PosJ0,PosJ0+2);
sJ0=sJ0.trim();
println("sJ0: "+sJ0);

if(sJ0.equals("J0")){
    println("llego: "+sJ0);
    texto2="";
    vr=0.1;
}
}

if(PosJ1!=-1){

    println("PosJ1: "+PosJ1);
    sJ1=texto2.substring(PosJ1,PosJ1+2);
    sJ1=sJ1.trim();
    println("sJ1: "+sJ1);

    if(sJ1.equals("J1")){
        println("llego: "+sJ1);
        texto2="";
        vr=1;
    }
}

if(PosJ2!=-1){

    println("PosJ2: "+PosJ2);
    sJ2=texto2.substring(PosJ2,PosJ2+2);
    sJ2=sJ2.trim();
    println("sJ2: "+sJ2);

    if(sJ2.equals("J2")){
        println("llego: "+sJ2);
        texto2="";
        vr=10;
    }
}

if(PosJ3!=-1){

    println("PosJ3: "+PosJ3);
```



```

sJ3=texto2.substring(PosJ3,PosJ3+2);
sJ3=sJ3.trim();
println("sJ3: "+sJ3);

if(sJ3.equals("J3")){
    println("llego: "+sJ3);
    texto2="";
    vr=100;
}
}

if(PosVL!=-1){

    println("PosVL: "+PosVL+" PosOK: "+PosOK);
    sVL=texto2.substring(PosVL+2,PosOK);
    sVL=sVL.trim();
    println("sVL: "+sVL);
    int velocidadDremel=int(sVL);

    println("iVL: "+velocidadDremel);

    if(velocidadDremel>=0 && velocidadDremel<=100){
        println("llego: "+velocidadDremel);
        texto2="";
        impresora.write("M117 VELOCIDAD "+velocidadDremel);
        impresora.write(13);
        impresora.write(10);
        String envio="S" + velocidadDremel;
        println(envio);
        variador.write(envio);
        variador.write(13);
        variador.write(10);
    }
}

texto2="";
PosIZ = -1;
PosDE = -1;
PosAD = -1;
PosAT = -1;
PosAR = -1;
PosAB = -1;
PosHXY = -1;
PosHZ = -1;

```

```

PosAS = -1;
PosDR = -1;
PosJ0 = -1;
PosJ1 = -1;
PosJ2 = -1;
PosJ3 = -1;
PosVL = -1;
PosOK = -1;
PosGComand = -1;
PosIntro = -1;
PosCMD = -1;
}

```

- **Evento comando**, es el encargado de enviar por la serial de la impresora el código escrito en el cuadro de texto.

```

public void comandos(String comando) {
    if(finicio){
        println("a button event from comando ");
        println(comando);

        if(comando!=""){
            String aux=comando;
            println(aux);
            impresora.write(aux);
            impresora.write(13);
            impresora.write(10);
            println("okC");
        }
    }
}

```

- **Botón búsqueda archivos**, este botón se encarga de buscar los archivos que serán utilizados tanto para modificar el código G, como el archivo donde se guardarán los cambios realizados.

```

public void ARCHIVO(boolean a) {
    if(finicio){
        println("a button event from ARCHIVO "+a);
        selectInput("Selecciona el Archivo a Modificar:", "fileSelected");
        delay(500);
        selectInput("Selecciona el Archivo Modificado", "fileSelected2");
        delay(500);
    }
}

```

```

}
}

```

- **Botón abrir archivo original**, este botón se encarga de mostrar el archivo generado por el CAD.

```

public void ORIGINAL(boolean a) {
    if(finicio){
        println("a button event from ORIGINAL "+a);
        launch(Archivo);
    }
}

```

- **Botón abrir archivo modificado**, este botón se encarga de mostrar el archivo modificado por el software.

```

public void MODIFICADO(boolean a) {
    if(finicio){
        println("a button event from MODIFICADO "+a);
        launch(Archivo2);
    }
}

```

- **Eventos cuadros de texto**, este evento se encarga de recibir el grosor del material a ser trabajado en mm.

```

public void ALTURA_PLANCHA(String a) {
    if(finicio){
        println("a button event from MODIFICADO "+a);
        if(a!=""){
            altura = int(a);
            if(altura>0 && altura<=50){
                println("altura aceptada: "+altura);
            }else{
                showMessageDialog(null, "Ingresa valores entre 1 y 50", "ERROR",
                ERROR_MESSAGE);
            }
        }
    }
}

```

- **Seleccionar archivos y procesar**, en este punto el archivo seleccionado el cual fue entregado por el CAD, ingresa a un proceso en el cual sus valores de profundidad

son cambiados y restados de la altura de la plancha para que la CNC pueda entender el proceso de corte/grabado.

```

void fileSelected(File selection) {
  if (selection == null) {
    println("Window was closed or the user hit cancel.");
  } else {
    println("User selected " + selection.getAbsolutePath());
    Archivo = selection.getAbsolutePath();
    input = loadStrings(Archivo);
    println("La ruta seleccionada es: "+Archivo);
    while(index < input.length){
      //println("gnc: "+input[index]);
      String a1 = "";
      String a2 = "";
      String a3 = "";
      String a4 = "";
      String a5 = "";

      String depth = "(Pass at depth ";
      int vdepth = depth.length();
      int c1 =input[index].indexOf(depth);
      int c2 =input[index].indexOf(")");
      if(c1!=-1 && c2!=-1){
        String sdepth =input[index].substring(c1+vdepth,c2).trim();
        //println("valor depth string: "+sdepth);
        float fdepth=0.00;
        fdepth = float(input[index].substring(c1+vdepth,c2));

        paso=altura+fdepth;

        println("contador: "+index+" valor profundidad: "+fdepth+" paso: "+paso);

      }

      int c3 =input[index].indexOf("G00 Z");
      if(c3!=-1){

        a3=input[index];
        a3=a3.substring(0,c3+5)+altura;

        //println(a3);
        input[index]=a3;
        saveStrings("data/output.ngc", input);
      }
    }
  }
}

```

```

}

int c5 =input[index].indexOf("Z-");
if(c5!=-1){
    a5=input[index];
    int vf =a5.indexOf("F");
    int vi =a5.indexOf("I");
    String apaso="";
    String dpaso="";
    String epaso="";

    if(vf>0){
        apaso=a5.substring(0,c5);
        dpaso=a5.substring(vf);
        epaso=a5.substring(c5,c5+1)+pasos+" ";
        a5=apaso+epaso+dpaso;
    }else
    if(vi>0){
        apaso=a5.substring(0,c5);
        dpaso=a5.substring(vi);
        epaso=a5.substring(c5,c5+1)+pasos+" ";
        a5=apaso+epaso+dpaso;
    }else{
        apaso=a5.substring(0,c5);
        epaso=a5.substring(c5,c5+1)+pasos+" ";
        a5=apaso+epaso;
    }
}

println(a5);
input[index]=a5;
saveStrings("data/output.ngc", input);
}

```

```

index = index + 1;

```

```

}

```

```

G00Z=0;
gg=0;
seguridadt=0.00;

```

```

}
}

```

```

void fileSelected2(File selection) {

```

```

if (selection == null) {
    println("Window was closed or the user hit cancel.");
} else {
    println("User selected " + selection.getAbsolutePath());
    Archivo2 = selection.getAbsolutePath();
}
}

```

- **Objeto cursores** es el encargado de ejecutar los movimientos de la CNC: arriba, abajo, derecha, izquierda, etc.

```

public void adelante() {
    if(finicio){
        impresora.write("M114");
        impresora.write(13);
        impresora.write(10);
        py=py+vr;
        if(py<=180){
            String aux="G0 Y"+py;
            println(aux);
            impresora.write(aux);
            impresora.write(13);
            impresora.write(10);
            println("ok");
        }
        if(py>200){py=200;println("py: "+py);}
    }
}

public void atras() {
    if(finicio){
        impresora.write("M114");
        impresora.write(13);
        impresora.write(10);
        py=py-vr;
        String aux="G0 Y"+py;
        println(aux);
        impresora.write(aux);
        impresora.write(13);
        impresora.write(10);
        println("ok");
        if(py<=0){py=0;println("py: "+py);}
    }
}

public void derecha() {

```

```
if(finicio){
    impresora.write("M114");
    impresora.write(13);
    impresora.write(10);
    px=px+vr;
    println("px: "+px);
    String aux="G0 X"+px;
    println(aux);
    impresora.write(aux);
    impresora.write(13);
    impresora.write(10);
    println("ok");
    if(px>200){px=200;println("px: "+px);}
}
}
public void izquierda() {
    if(finicio){
        impresora.write("M114");
        impresora.write(13);
        impresora.write(10);
        px=px-vr;
        println("px: "+px);
        String aux="G0 X"+px;
        println(aux);
        impresora.write(aux);
        impresora.write(13);
        impresora.write(10);
        println("ok");
        if(px<=0){px=0;println("px: "+px);}
    }
}
public void arriba() {
    if(finicio){
        impresora.write("M114");
        impresora.write(13);
        impresora.write(10);
        pz=pz+vr;
        println("pz: "+pz);
        String aux="G0 Z"+pz;
        println(aux);
        impresora.write(aux);
        impresora.write(13);
        impresora.write(10);
        println("ok");
    }
}
```

```

        if(pz>200){pz=200;println("pz: "+pz);}
    }
}
public void abajo() {
    if(finicio){
        impresora.write("M114");
        impresora.write(13);
        impresora.write(10);
        pz=pz-vr;
        println("pz: "+pz);
        String aux="G0 Z"+pz;
        println(aux);
        impresora.write(aux);
        impresora.write(13);
        impresora.write(10);
        println("ok");
        if(pz<=0){pz=0;println("pz: "+pz);}
    }
}
}

```

- **Objeto RadioButton** es el encargado de generar los avances en los cursores entre cantidades de: 0,1 – 1 – 10 – 100.

```

void radioButton(int a) {
    if(finicio){
        println("a radio Button event: "+a);
        if(a==1){
            vr=0.1;
        }
        if(a==2){
            vr=1;
        }
        if(a==3){
            vr=10;
        }
        if(a==4){
            vr=100;
        }
        if(a!=1 && a!=2 && a!=3 && a!=4){
            vr=0;
        }
    }
}
}
}

```



- **Home Z**, en esta programación se envía el comando de home en el eje z hacia la impresora.

```
public void HOME_Z() {
    if(finicio){
        String HZ="G28 Z";
        println(HZ);
        impresora.write(HZ);
        impresora.write(13);
        impresora.write(10);
        println("ok");
    }
}
```

- **Home XY**, en esta programación se envía el comando de home en el eje x, y hacia la impresora.

```
public void HOME_XY() {
    if(finicio){
        String HXY="G28 X Y";
        println(HXY);
        impresora.write(HXY);
        impresora.write(13);
        impresora.write(10);
        println("ok");
    }
}
```

- **Dremel**, esta sección se encarga de enviar el estado encendido o apagado hacia el puerto serial del variador.

```
public void DREMEL() {
    if(finicio){
        if(fd==true){
            delay(100);
            fd=false;
            String sd="DON";
            println(sd);
            impresora.write("M117 dremel on");
            impresora.write(13);
            impresora.write(10);

            variador.write(sd);
        }
    }
}
```

```

    variador.write(13);
    variador.write(10);    )

    //bluetooth.write(sd);
    //bluetooth.write(13);
    //bluetooth.write(10);

    println("ok");
}else{
    delay(100);
    fd=true;
    String sd="DOFF";
    println(sd);
    impresora.write("M117 dremel off");
    impresora.write(13);
    impresora.write(10);

    variador.write(sd);
    variador.write(13);
    variador.write(10);

    //bluetooth.write(sd);
    //bluetooth.write(13);
    //bluetooth.write(10);

    println("ok");

}
}
}

```

- **Botón enviar**, aquí se puede encontrar la programación que hace posible enviar la velocidad escogida para el Dremel.

```

public void enviar() {
    if(finicio){
        println("a button event from enviar ");
        println(int(velo.getValue()));
        if(int(velo.getValue())>=0 && int(velo.getValue())<101){ //Comparo que el
valor que me genera el evento del slider
            impresora.write("M117 VELOCIDAD "+int(velo.getValue()));
            impresora.write(13);
            impresora.write(10);
            String envio="S" + int(velo.getValue());

```

```

println(envio);
variador.write(envio);
variador.write(13);
variador.write(10);
println("ok");
}
}
}

```

- **Botón Iniciar**, es el encargado de enviar el archivo de (corte/grabado) hacia la impresora

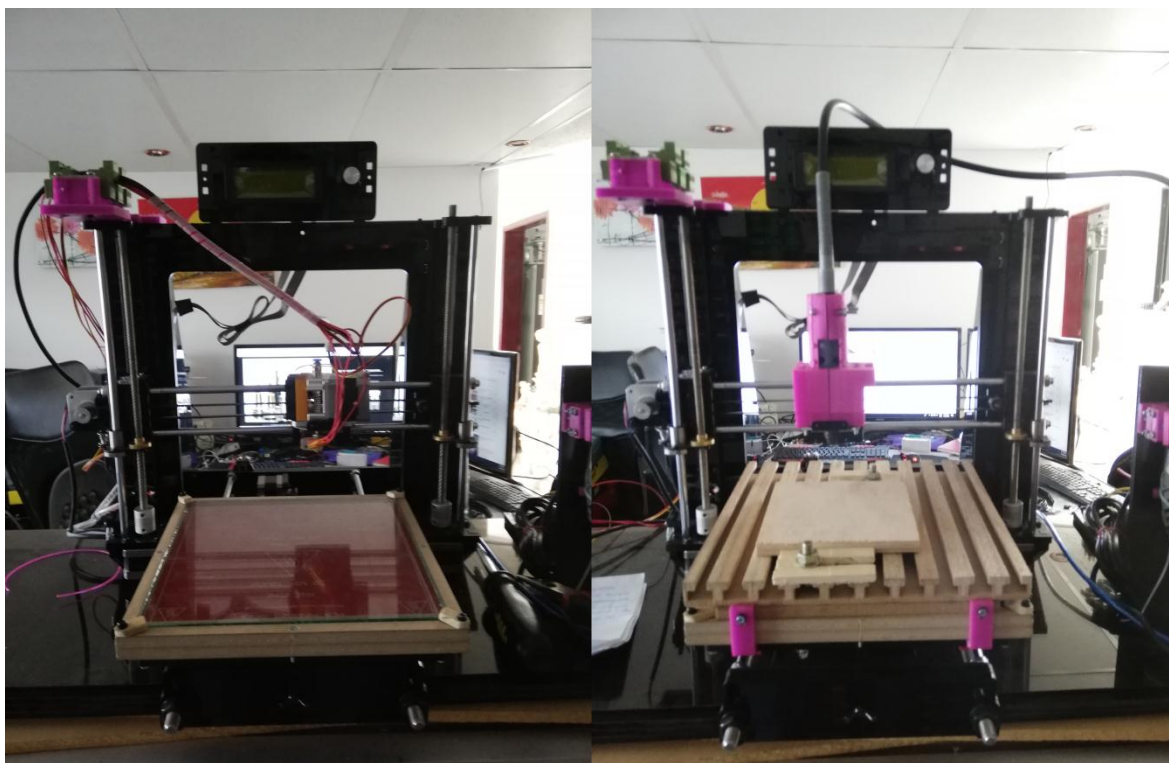
```

public void Iniciar() {
    if(finicio){
        println("1");
        lines1 = loadStrings("data/oto2.ngc");
        while(index < lines1.length){
            print("I: "+index+" ");
            String a = "";
            println(lines1[index]);
            a=lines1[index];
            println("A: "+a);
            a.trim();
            impresora.write(a);
            impresora.write(13);
            impresora.write(10);
            index = index + 1;
            delay(2000);
        }
    }
    index=0;
}

```

## ANEXO 7. Manual técnico y de usuario

### PROTOTIPO MULTIPROPÓSITO FRESADORA CNC E IMPRESORA VOLUMÉTRICA



### MANUAL TÉCNICO Y DE USUARIO

Autor: Fernando Cajo

Quito – Ecuador

2019

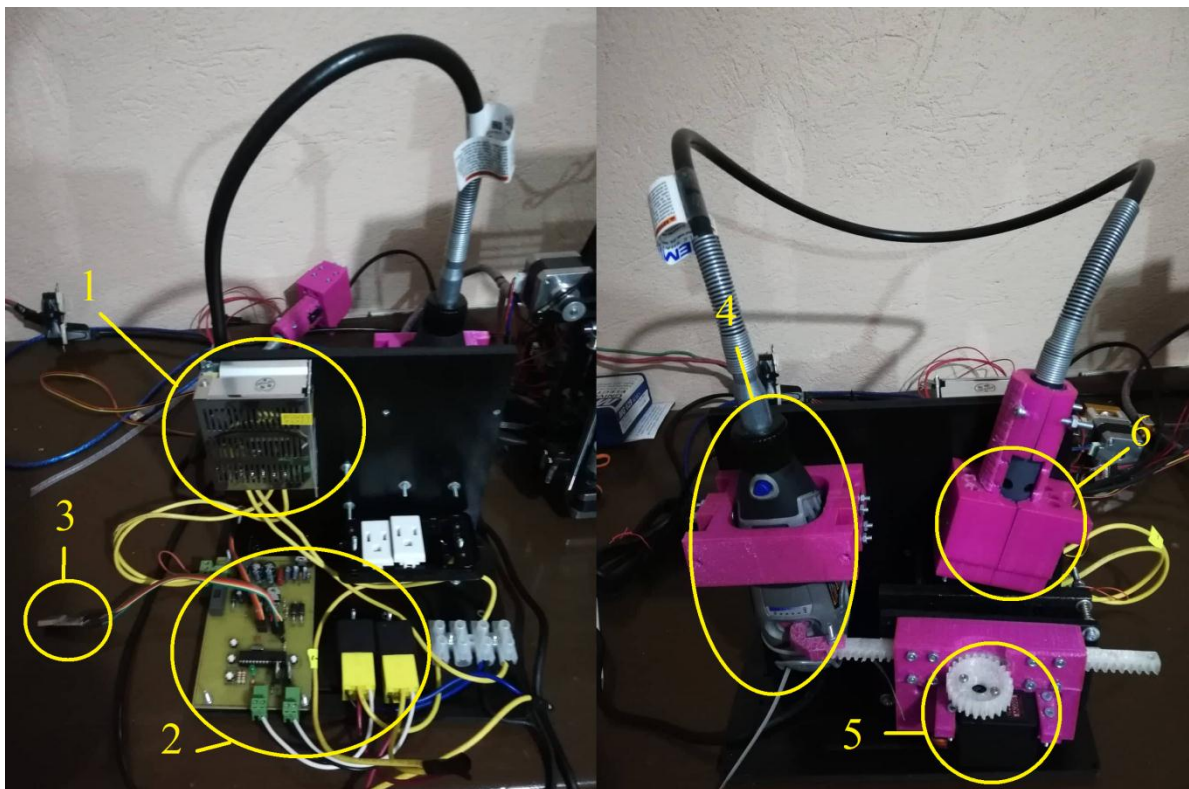
## ÍNDICE

Descripción del módulo CNC .....	2
Pasos para utilizar la impresora 3D como CNC fresadora .....	3
Pasos para utilizar la interfaz de la fresadora CNC cortadora/grabadora.....	4
Pasos para utilizar la interfaz del control remoto de la fresadora CNC .....	8

## ÍNDICE DE FIGURAS

Figura. 1. Descripción del módulo CNC .....	2
Figura. 2. Colocación de aditamentos en la impresora 3D.....	3
Figura. 3. Interfaz de la CNC cortadora/grabadora .....	4
Figura. 4. Ingreso del código M206 X0 Y0 Z? .....	4
Figura. 5. Ingreso del código M500 .....	5
Figura. 6. Ingreso del código G92 X0 Y0 Z0.....	5
Figura. 7. Ingreso del espesor del material.....	6
Figura. 8. Selección de archivos.....	6
Figura. 9. Comprobación de la reinterpretación del código .....	7
Figura. 10. Selección de velocidad del Dremel e inicio de corte/grabado .....	7
Figura. 11. Reset.....	8
Figura. 12. Control remoto de la CNC .....	8
Figura. 13. Segunda interfaz del control remoto .....	9

## Descripción del módulo CNC.



**Figura. 1. Descripción del módulo CNC.**

Fuente: Elaborado por el autor.

1. Fuente de alimentación CNC: 120 AC
2. Circuito de control
3. Puerto USB
4. Dremel
5. Servomotor MG995
6. Cabezal de fresado

El sistema está adecuado para funcionar en la infraestructura de una impresora 3D.

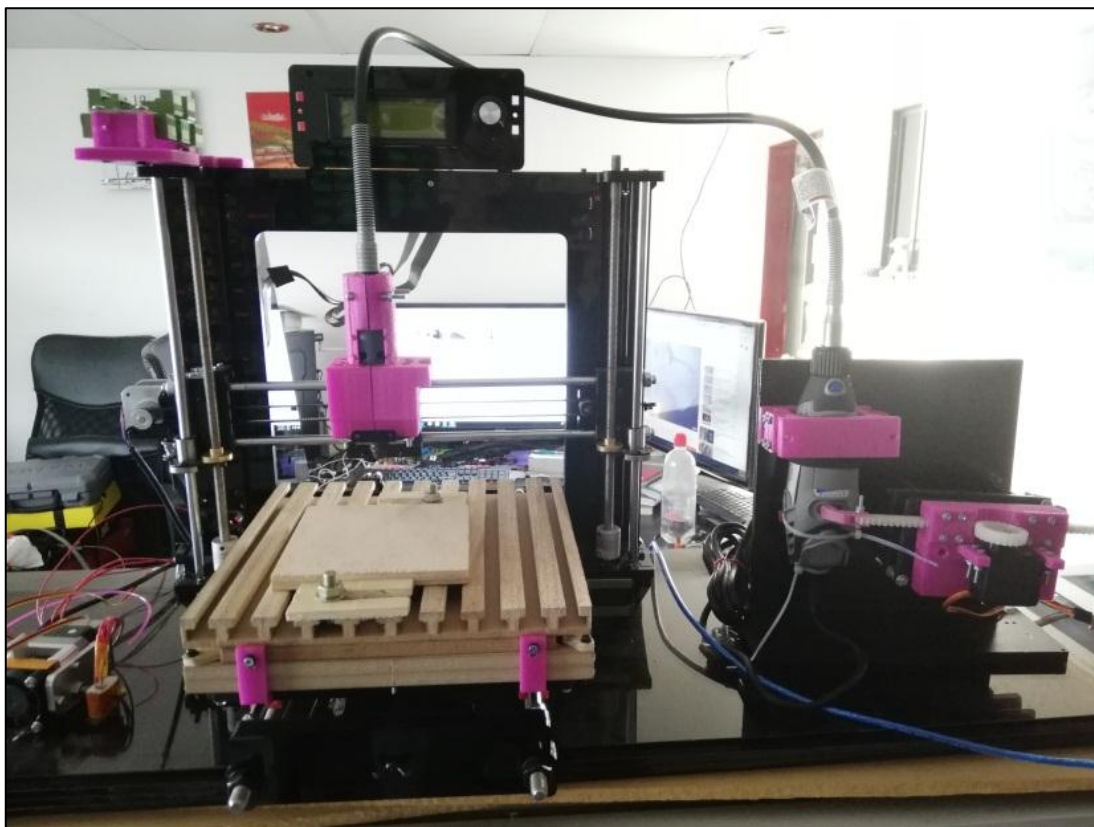
Diámetro de fresa máximo: 3mm para ser colocado en el mandril del flex

Necesidad de dos puertos tipo USB (módulo CNC e impresora 3D).

Utiliza un enchufe AC

## **Pasos para utilizar la impresora 3D como CNC fresadora.**

1. Colocar los aditamentos sobre la estructura de la Impresora 3D, como: cabezal de la fresadora junto al módulo de la CNC, plataforma de corte/grabado.

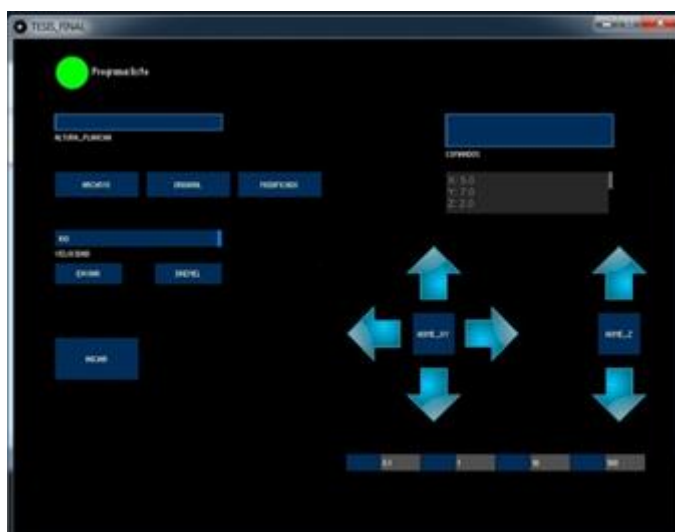


**Figura. 2. Colocación de aditamentos en la impresora 3D.**

Fuente: Elaborado por el autor.

2. Conectar a la PC los puertos USB correspondientes, tanto de la impresora como del módulo de corte/grabado.
3. Ejecutar el software de corte/grabado en la PC y esperar a que se encienda la luz verde para empezar a utilizar la interfaz.

## Pasos para utilizar la interfaz de la fresadora CNC cortadora/grabadora



**Figura. 3. Interfaz de la CNC cortadora/grabadora.**

Fuente: Elaborado por el autor.

1. Utilizando las flechas de movimiento, colocar la punta de la fresadora sobre el material a ser cortado/grabado. Este será el punto de inicio y finalización para el proceso de corte o grabado.
2. Ingresar el código M206 X0 Y0 Z?, en el cual se colocará el espesor de la pieza a ser cortada o grabada en el valor de Z.

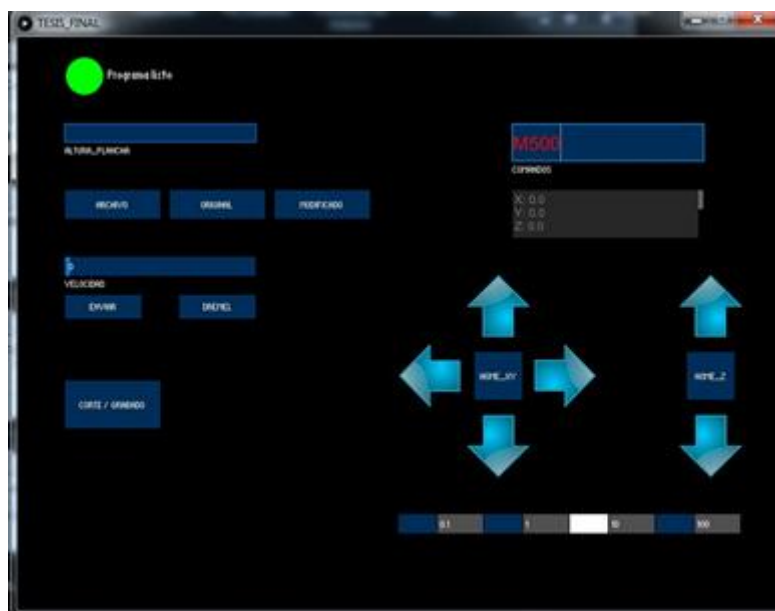


**Figura. 4. Ingreso del código M206 X0 Y0 Z?.**

Fuente: Elaborado por el autor.



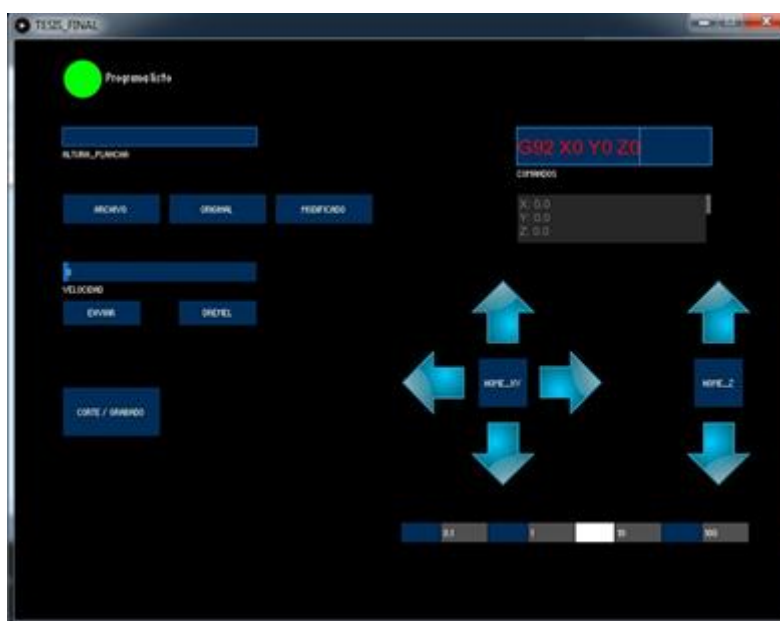
3. Ingresar el código M500 para grabar los datos ingresados anteriormente.



**Figura. 5. Ingreso del código M500.**

Fuente: Elaborado por el autor.

4. Ingresar el código G92 X0 Y0 Z0 para especificar que el punto en donde se encuentra actualmente la fresa será la referencia de X0 Y0 Z0 para la CNC.



**Figura. 6. Ingreso del código G92 X0 Y0 Z0.**

Fuente: Elaborado por el autor.

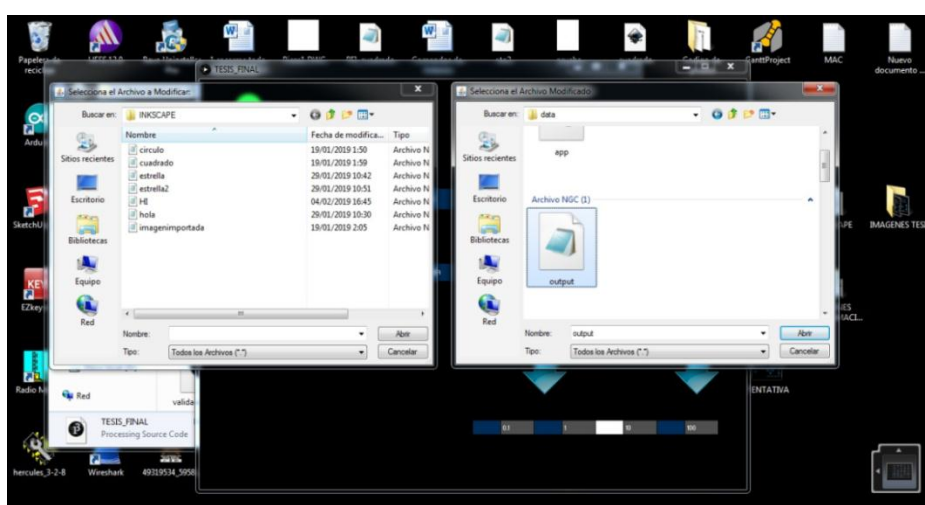
- Ingresar el espesor del material a ser cortado/grabado en mm



**Figura. 7. Ingreso del espesor del material.**

Fuente: Elaborado por el autor.

- Pulsar el botón “Archivo” y seleccionar en la ventana: “archivo modificado”, el documento llamado “output” el cual siempre se encontrará en la carpeta llamada “data” dentro del programa que se ejecutó. Por otra parte en la ventana denominada “selecciona el archivo a modificar” simplemente se selecciona el archivo generado por el software Inkscape, el cual será modificado para su posterior utilización.



**Figura. 8. Selección de archivos.**

Fuente: Elaborado por el autor.

7. Utilizando los botones “original” y “modificado” se puede asegurar y observar que el proceso de reinterpretación se ha realizado con éxito.

```

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

(Pass at depth -0.25)
(Start cutting path id: path825)
(Change tool to cylindrical cutter)
G00 Z10.000000
G00 X58.762007 Y17.553138

G01 Z-0.250000 F100.0(Penetrate)
G01 X57.028642 Y17.553138 Z-0.250000 F400.000000
G01 X55.830943 Y21.922669 Z-0.250000
G01 X50.543535 Y21.922669 Z-0.250000
G01 X49.344836 Y17.553138 Z-0.250000
G01 X47.694574 Y17.553138 Z-0.250000
G01 X52.144536 Y33.230830 Z-0.250000
G01 X54.312045 Y33.230830 Z-0.250000
G01 X58.762007 Y17.553138 Z-0.250000
G00 Z10.000000

(Change tool to cylindrical cutter)
G00 Z10.000000
G00 X55.330118 Y23.712603

G01 Z-0.250000 F100.0(Penetrate)
G01 X53.187240 Y31.409311 Z-0.250000 F400.000000
G01 X51.036151 Y23.712603 Z-0.250000
G01 X55.330118 Y23.712603 Z-0.250000
G00 Z10.000000

(End cutting path id: path825)
(Start cutting path id: path823)
(Change tool to cylindrical cutter)
G00 Z10.000000
G00 X47.497528 Y17.553138

G01 Z-0.250000 F100.0(Penetrate)
G01 X39.763462 Y17.553138 Z-0.250000 F400.000000
G01 X39.763462 Y33.230830 Z-0.250000
G01 X41.389093 Y33.230830 Z-0.250000
  
```

Figura. 9. Comprobación de la reinterpretación del código.

Fuente: Elaborado por el autor.

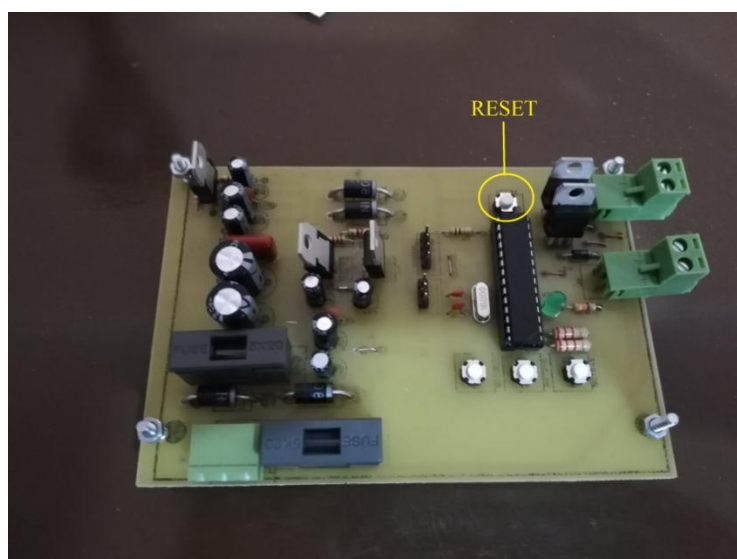
8. Se puede elegir el nivel de velocidad en el Dremel utilizando el slider llamado “velocidad”, a continuación se deberá encender el Dremel con el Botón “Dremel”. Finalmente se presiona el botón “inicio” para ejecuta la acción de corte/grabado.



Figura. 10. Selección de velocidad del Dremel e inicio de corte/grabado.

Fuente: Elaborado por el autor.

9. La placa también cuenta con un botón de emergencia para resetear el sistema. (este botón debe ser presionado luego de cerrar la interface en la PC)

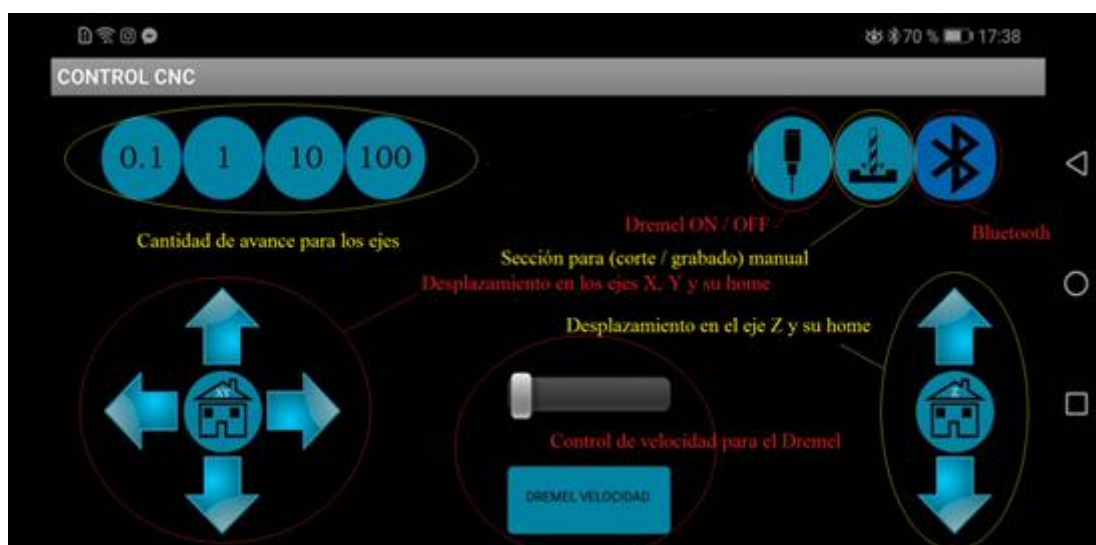


**Figura. 11. Reset.**

Fuente: Elaborado por el autor.

### Pasos para utilizar la interfaz del control remoto de la fresadora CNC.

1. Vincular vía Bluetooth la aplicación con la PC
2. La ejecución de las funciones es muy similar a la interfaz de la PC.



**Figura. 12. Control remoto de la CNC.**

Fuente: Elaborado por el autor.

## Segunda interfaz y sus funciones

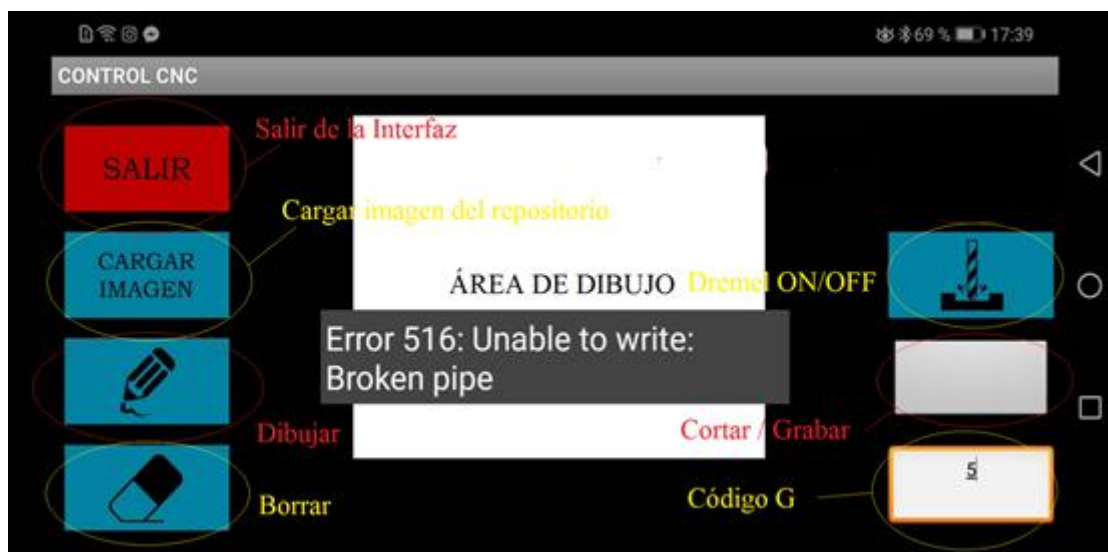


Figura. 13. Segunda interfaz del control remoto.

Fuente: Elaborado por el autor.