

## **Grabadora de madera**

Alumnas: Amparo Appiolaza y Victoria Castillo

### **Índice**

#### **Introducción**

- 1.1. Objetivo del Proyecto
- 1.2. Descripción General del Proyecto
- 1.3. Alcances y Limitaciones del Proyecto

#### **Antecedentes y Fundamentos Teóricos**

- 2.1. Tecnología CNC y Control de Máquinas
- 2.2. ¿Qué es GRBL y su Aplicación en el Control de Máquinas?
- 2.3. G-code y su Uso en el Control de la Grabadora Láser

#### **Materiales y Métodos**

- 3.1. Componentes Hardware Utilizados
- 3.2. Software y Framework Utilizados
  - 3.2.1. ESP-IDF
  - 3.2.2. GRBL-ESP32
  - 3.2.3. PlatformIO
- 3.3. Esquema de Conexiones y Diagramas del Proyecto

## **Introducción**

### **1.1. Objetivo**

Nuestro proyecto tiene como objetivo aprobar la materia. Mediante la fabricación de una máquina láser, la cual podrá grabar madera con diseños.

### **1.2.Descripción del proyecto**

Su funcionamiento va a estar basado en que el Esp32 va a controlar cómo opera la máquina, cada motor paso a paso va a controlar el movimiento del cabezal, uno en el eje Y y otro en eje X, el sensor de temperatura supervisa la temperatura del láser y también del grabado para evitar sobrecalentamientos y el sensor de fin de carreras evita que la máquina sobrepase su área de trabajo.

### **1.3.Alcances y Limitaciones del Proyecto**

Las principales limitaciones que enfrentamos fueron los costos y la búsqueda de materiales, el modelado 3D y el ensamblaje de la placa. En cuanto a los materiales, uno de los principales inconvenientes fue encontrar el láser adecuado a un precio económico. En el caso del modelado 3D, logramos obtener los modelos a un precio relativamente bajo (1,20 €), pero nos vimos en la necesidad de ajustar la escala, lo cual nos tomó tiempo debido a diversas complicaciones al utilizar TinkerCAD. Por último, en el proceso de armado de la placa, tuvimos que adaptarla a una CNC Shield y conectarla mediante bornes a los sensores finales de carrera. Este paso nos presentó desafíos adicionales, ya que carecíamos de los conocimientos técnicos previos, lo que nos obligó a realizar una investigación más profunda.

## **Antecedentes y Fundamentos Teóricos**

### **2.1. Tecnología CNC y Control de Máquinas**

La CNC (control numérico por computadora) se refiere al control automatizado de maquinaria y herramientas, como taladros, herramientas de mandrinar, láseres y tornos.

### **2.2. ¿Qué es GRBL y su Aplicación en el Control de Máquinas?**

GRBL es un firmware de código abierto diseñado específicamente para controlar máquinas CNC (Control Numérico por Computadora) utilizando un microcontrolador, generalmente un Arduino Uno. Este software permite que el Arduino interprete y ejecute comandos G-code, que son instrucciones estándar utilizadas en el control de máquinas CNC y fresadoras.

GRBL se ejecuta en una placa Arduino con un microcontrolador Atmega328 y se comunica con los motores de la máquina CNC a través de señales digitales. Ofrece las siguientes características:

- Interpreta código G (G-code), el lenguaje de programación más común en CNC.
- Soporta hasta 3 ejes de coordenadas (X, Y, Z) y permite controlar motores paso a paso (stepper motors).
- Permite configurar parámetros como la velocidad de avance, la aceleración y la deceleración.

- Soporta varios tipos de motores y
- controladores de motores.

En nuestro caso, tuvimos que usar un grbl adaptado para ESP32, utilizando PlatformIO para poder programar nuestro dispositivo.

### **2.3. G-code y su Uso en el Control de la Grabadora Láser**

El G-code es un lenguaje de programación más utilizado en el área de máquinas CNC. Que en términos básicos es un conjunto de comandos que le indican a la máquina una serie de movimientos, en otras palabras les dice “que hacer” y “cómo hacerlo”. Los comandos suelen empezar con una letra, seguida de un número y una serie de instrucciones.

Su uso en las grabadoras láser ayuda en el movimiento del cabezal del láser junto con su potencia, velocidad de corte, la activación y desactivación del láser, las secuencias y el control de la altura. También nos permite la automatización lo que reduce los errores humanos y nos da una mejor optimización.

## **Materiales y Métodos**

### **3.1. Componentes Hardware Utilizados**

- Placa de cobre
- Actuadores:
  - 2 motores paso a paso
- Sensores:
  - 4 sensores fines de carrera: Limit Switch
- ESP32
- Modelado 3D
- Herramienta de grabado

### **3.2. Software y Framework Utilizados**

- Comunicación I2C
- Platformio

#### **3.2.1. ESP-IDF**

ESP-IDF (Espressif IoT Development Framework) es el entorno de desarrollo oficial de Expressif para sus microcontroladores como es el caso del ESP32. Este framework proporciona las herramientas para programar y compilar proyectos en Esp utilizando interfaces de comunicación WIFI y Bluetooth. Lo decidimos emplear porque lleva las librerías integradas, contiene compatibilidad con FreeRTOS, control detallado de hardware y funciones específicas de microcontrolador con periféricos I2C.

### **3.2.2. GRBL-ESP32**

GRBL-ESP32 es una versión de GRBL adaptada para manejar microcontroladores ESP32. La cual decidimos utilizarla porque nos permite gestionar procesos de manera simultánea y es más fácil la integración. El programa que decidimos utilizar es LaserGRBL. El cual funciona subiendo un archivo, y el propio programa lo convierte en G-code y le da las instrucciones.

### **3.2.3. PlatformIO**

Decidimos utilizar Platformio integrado con Visual Studio por su compatibilidad con ESP32.