

**Grado en Ingeniería de Tecnologías Industriales**

**Trabajo de Fin de Grado**

El presente trabajo, titulado ***DISEÑO DE UN SISTEMA DE BOBINADO AUTOMÁTICO BASADO EN HARDWARE LIBRE***, constituye la memoria correspondiente a la asignatura Trabajo de Fin de Grado que presenta Dª. ***Marta Martín Bravo*** como parte de su formación para aspirar al Título de Graduado/a en Ingeniería de Tecnologías Industriales. Este trabajo ha sido realizado en ***Escuela Superior de Ciencias Experimentales y Tecnología*** en el ***Departamento de Matemática Aplicada, Ciencia e Ingeniería de los Materiales y Tecnología Electrónica*** bajo la dirección de ***Felipe Machado Sánchez***.

Móstoles, 22 de septiembre de 2018

AGRADECIMIENTOS

RESUMEN

En el Área de Tecnología Electrónica, el elevado coste de una bobinadora automática comercial obligaba a devanar manualmente las bobinas que necesitaban para sus proyectos de investigación. Este tedioso trabajo puede durar horas y no garantiza la precisión que las características de la bobina requieren.

Por esta razón y sumado al hecho de que sólo harían uso de ella de forma ocasional, no compensaría hacer tal desembolso de dinero, y surgió la necesidad de diseñar un sistema de bobinado automático de bajo coste.

Este proyecto consiste en el diseño una bobinadora automática, la cual funciona mediante un sistema de movimiento de traslación y rotación. Los movimientos de la bobina han sido programados en un microcontrolador (placa Arduino) que determina los giros de dos motores paso a paso.

Cabe destacar que el diseño del soporte es modular e intercambiable, y permite una gran adaptabilidad del dispositivo a distintos diámetros y longitudes de bobina, en función de las características requeridas por el usuario.

Para el desarrollo del proyecto, se ha hecho uso de software y hardware libre. Por ello, cualquier persona podría replicar la bobinadora, ya que todas las herramientas utilizadas son de libre acceso y los dispositivos electromecánicos son comerciales, económicos y la documentación de estos dispositivos está accesible para todo el mundo.

Este TFG constituye un prototipo de sistema de devanado automático para bobinas accesible, al que podríamos añadirle mejoras en cuanto a (INCLUIR AL FINAL). Todos los diseños, tanto de códigos como de piezas para impresión 3D, han sido compartidos en GitHub [7].

Por último, cabe destacar que durante el desarrollo del TFG la autora ha profundizado en sus conocimientos de programación de microcontroladores, diseño CAD, electrónica y mecánica, además de su formación en nuevos ámbitos como la impresión 3D.

ABREVIATURAS

ATX Advanced Technology Extended – Tecnología avanzada extendida

BOM Bill of materials – Lista de materiales

CA Corriente alterna

CC Corriente continua

CNC Computer Numerical Control – Ordenador de Control Numérico

DIY Do it yourself – Hazlo tú mismo

f.e.m Fuerza electromotriz

GND Ground – Puesta a tierra

I2C Inter-Integrated Circuit – Circuito inter-integrado

IDE Integrated Development Environment – Entorno de desarrollo integrado

LCD Liquid Crystal Display – Pantalla de cristal líquido

LED Light Emitting Diode - Diodo emisor de luz

NC Normalmente Cerrado

NA Normalmente Abierto

PLA Ácido Poliáctico

RAM Random Access Memory - Memoria de acceso aleatorio

SDA Serial Data – Envío de datos en serie

SCL Serial Clock – Envío de la señal de reloj en serie

USB Universal Serial Bus - Bus Universal en Serie

V Voltios

TABLA DE ILUSTRACIONES

Ilustración 1. Bobinas de un transformador 10

Ilustración 2. Bobina de cobre 10

Ilustración 3. Líneas de flujo magnético 10

Ilustración 4. Dimensiones de la bobina 13

Ilustración 5. Bobinadora comercial 14

Ilustración 6. Bobinadora con estructura impresa en 3D 15

Ilustración 7. Diseño de bobinadora automática con estructura metálica 16

Ilustración 8. Sistema de bobinado automático 22

Ilustración 9. Componentes del sistema de bobinado 23

Ilustración 10. Esquema general de la bobinadora 24

Ilustración 11. Interfaz de Repetier Host 28

Ilustración 12. Piezas impresas en 3D fallidas 29

Ilustración 13. Soporte principal 30

Ilustración 14. Conjunto del soporte principal y las carcasas 31

Ilustración 15. Carcasas del soporte principal 31

Ilustración 16. Soporte de los motores 32

Ilustración 17. Módulo alzador 32

Ilustración 18. Conjunto del soporte principal con carcasas, soporte motor y módulos alzadores 33

Ilustración 19. Soporte del eje 33

Ilustración 20. Tensionador de la correa 34

Ilustración 21. Soporte del final de carrera 34

Ilustración 22. Tensionador del hilo 35

Ilustración 23. Características de la bobina 36

Ilustración 24. Esquema placa Arduino UNO 38

Ilustración 25. Proyectos de ejemplo 39

Ilustración 26. Seleccionar modelo de placa 40

Ilustración 27. Esquema IDE 40

Ilustración 28. Funciones botones 41

Ilustración 29. Esquema de funcionamiento del final de carrera (NC) 42

Ilustración 30. Descripción general del sistema 43

Ilustración 31. Pantalla LCD con botones 44

Ilustración 32. Secuencia de pantallas 45

Ilustración 33. Arquitectura de tipo master-slave 46

Ilustración 34. Conexión entre placas Arduino UNO 48

Ilustración 35. Motor paso a paso Nema 17 Wantai 50

Ilustración 36. Cables de salida del motor paso a paso 51

Ilustración 37. Conexión de los cables de los motores paso a paso 51

Ilustración 38. CNC Shield 52

Ilustración 39. Esquema de conexiones en la placa CNC Shield 52

Ilustración 40. Esquema de conexión de los motores en la placa CNC Shield 53

Ilustración 41. Pines de conexión de la placa Arduino UNO 54

Ilustración 42. Pololu A4988 55

Ilustración 43. Esquema de conexiones del Pololu A4988 55

ÍNDICE

[1. INTRODUCCIÓN 10](#_Toc32057916)

[1.1. BOBINAS: CONCEPTOS GENERALES 10](#_Toc32057917)

[1.1.1 PRINCIPIO DE FUNCIONAMIENTO 11](#_Toc32057918)

[1.2. ESTADO DEL ARTE 15](#_Toc32057919)

[1.2.1. BOBINADORAS COMERCIALES 15](#_Toc32057920)

[1.2.2. BOBINADORAS DIY 16](#_Toc32057921)

[1.1.2 MODELO 2: Estructura de componentes metálicos 17](#_Toc32057922)

[1.3. ESQUEMA DE LA MEMORIA 18](#_Toc32057923)

[2. OBJETIVOS 19](#_Toc32057924)

[2.1. ESTRUCTURA DEL PLAN DE TRABAJO 19](#_Toc32057925)

[2.1.1. Definición de especificaciones 19](#_Toc32057926)

[2.1.2. Estudio de alternativas 19](#_Toc32057927)

[2.1.3. Adquisición de materiales 20](#_Toc32057928)

[2.1.4. Documentación de los avances 20](#_Toc32057929)

[2.1.5. Aprendizaje del software 20](#_Toc32057930)

[2.1.6. Diseño mecánico e impresión de las piezas 20](#_Toc32057931)

[2.1.7. Montaje del sistema 20](#_Toc32057932)

[2.1.8. Programación del control de los motores paso a paso 20](#_Toc32057933)

[2.1.9. Calibración del sistema 21](#_Toc32057934)

[2.1.10. Elaboración de la memoria 21](#_Toc32057935)

[3. SOLUCIÓN TÉCNICA 22](#_Toc32057936)

[3.1. Descripción general del prototipo 23](#_Toc32057937)

[3.2. Movimientos del sistema 24](#_Toc32057938)

[3.2.1. Movimiento de traslación 24](#_Toc32057939)

[3.2.2. Movimiento de rotación 24](#_Toc32057940)

[3.2.3. Relación entre movimientos 25](#_Toc32057941)

[3.3. Sistema mecánico 26](#_Toc32057942)

[3.3.1. Estructura 26](#_Toc32057943)

[3.3.2. Transmisión del movimiento 27](#_Toc32057944)

[3.3.3. Impresión 3D 28](#_Toc32057945)

[3.3.3.2. Repetier Host 29](#_Toc32057946)

[3.3.4. Componentes impresos 30](#_Toc32057947)

[3.3.5. Carrete de la bobina objetivo 36](#_Toc32057948)

[3.4. Sistema electrónico de control 38](#_Toc32057949)

[3.4.1. Arduino UNO 38](#_Toc32057950)

[3.4.2. Interruptor de final de carrera 43](#_Toc32057951)

[3.4.3. Descripción del sistema 44](#_Toc32057952)

[3.5. Sistema electrónico de potencia 51](#_Toc32057953)

[3.5.1. Motores paso a paso 51](#_Toc32057954)

[3.5.2. Placa CNC Shield V3.51 53](#_Toc32057955)

[1.1.1 Pololu A4988 56](#_Toc32057956)

[3.6. Especificaciones del diseño 58](#_Toc32057957)

[3.7. Líneas futuras 58](#_Toc32057958)

[4. CONCLUSIONES 59](#_Toc32057959)

[5. BIBLIOGRAFÍA 60](#_Toc32057960)

[6. APÉNDICES 62](#_Toc32057961)

[6.1. Presupuesto de una bobinadora profesional 62](#_Toc32057962)

[6.2. Lista de materiales adquiridos 63](#_Toc32057963)

# INTRODUCCIÓN

La finalidad de este TFG es el diseño y la construcción de una bobinadora automática. Este dispositivo de bobinado no tendrá una precisión equiparable a una comercial, pero será suficiente para las bobinas con las que se va a trabajar en el Área de Tecnología Electrónica.

Esta bobinadora puede tener diversos usos, como enrollar hilo de costura o hilo de pescar, pero el objetivo de este proyecto es devanar bobinas de hilo de cobre.

El funcionamiento general de las bobinadoras consiste en los movimientos giratorio y traslacional simultáneos de un cilindro. El movimiento traslacional puede realizarlo el cilindro que se quiere devanar o bien la bobina suministradora de hilo.

La relación entre estos dos movimientos dependerá de las características que el usuario defina inicialmente para su bobina, como el grosor del hilo, el diámetro y la longitud de la bobina o el número de capas.

Puesto que las bobinas son componentes imprescindibles de los circuitos magnéticos, la realización de este proyecto supondrá una gran ayuda al Área de Tecnología Electrónica para devanar las bobinas de sus proyectos, con las características que requieran, en un tiempo muy reducido y a un coste mínimo.

## BOBINAS: CONCEPTOS GENERALES

Una bobina es un componente eléctrico pasivo que incluye un alambre aislado, el cual se arrolla en forma de hélice. Esto le permite almacenar [energía](https://definicion.de/energia) en un campo magnético a través de un fenómeno conocido como autoinducción. Generalmente, está formada por devanado de cobre arrollado sobre un núcleo ferromagnético.

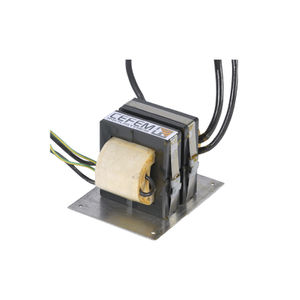


Ilustración 1. Bobinas de un transformador



Ilustración 2. Bobina de cobre

### PRINCIPIO DE FUNCIONAMIENTO

El funcionamiento de una bobina se basa en el fenómeno de la inducción electromagnética, para explicarlo, se introducen algunos conceptos básicos [1]:

El flujo magnético , determina el número de líneas de campo que atraviesan una superficie S sometida a la acción de un campo magnético y depende del producto vectorial del campo magnético , y el vector normal a un diferencial de la superficie :

Ecuación 1



Ilustración 3. Líneas de flujo magnético

Teniendo en cuenta esta ecuación, si queremos modificar el flujo magnético que atraviesa una espira podemos:

* Variar el ángulo entre la fuente de y la espira.
* Cambiar la superficie de la espira.
* Si está generado por la inducción de una corriente I, podemos variar dicha corriente y, de esta forma, modificar .

El campo magnético inducido en un solenoide se calcula mediante la siguiente ecuación:

Ecuación 2

Siendo la permitividad magnética, el número de espiras, la longitud del solenoide e la intensidad que circula por el solenoide.

El fenómeno de la inducción electromagnética se basa en la Ley de Faraday-Lenz, por la cual se puede inducir una corriente I mediante una variación en el campo magnético y viceversa.

La Ley de Faraday establece que la fuerza electromotriz (f.e.m.) inducida en una bobina es directamente proporcional a la variación del flujo magnético en el tiempo.

La Ley de Lenz determina que la f.e.m y la corriente I inducidas en una espira o lazo tienden a oponerse al cambio que las genera.

Ecuación 3

En el caso de una bobina, la depende del número de vueltas N del hilo de cobre:

Ecuación 4

En concreto, en las bobinas se produce una autoinducción, es decir, una variación de la intensidad de la corriente produce un campo magnético variable, que da lugar a una fuerza electromotriz inducida y una corriente inducida que se opone a la corriente inicial inductora.

#### CIRCUITO INDUCTOR

Un circuito inductor es un circuito que consiste en un [conductor eléctrico](https://es.wikipedia.org/wiki/Conductor_el%C3%A9ctrico) enrollado alrededor de un núcleo (ya sea de aire o de [hierro](https://es.wikipedia.org/wiki/Hierro)). El fenómeno de autoinducción surge cuando el inductor y el [inducido](https://es.wikipedia.org/wiki/Inducido) constituyen el mismo elemento.

Cuando por un circuito circula una corriente eléctrica, alrededor se crea un [campo magnético](https://es.wikipedia.org/wiki/Campo_magn%C3%A9tico). Si varía la corriente, dicho campo también varía y, según la [Ley de Faraday](https://es.wikipedia.org/wiki/Ley_de_Faraday)-Lenz, en el circuito se produce una fuerza electromotriz o voltaje inducido, denominado fuerza electromotriz autoinducida.

Según lo explicado anteriormente, el flujo magnético es proporcional al campo magnético , y éste es, a su vez proporcional a la corriente I inductora. El flujo magnético se relaciona con la corriente a través del coeficiente de autoinducción L o inductancia.

Ecuación 5

Siendo, la sección de la bobina y la densidad de espiras:

Ecuación 6

Aplicando la Ecuación 1 a un solenoide y sustituyendo en ella la Ecuación 2, la inductancia de una bobina sería:

Ecuación 7

Siendo el flujo magnético el producto de la inductancia por la corriente eléctrica, de la Ecuación 7 podemos deducir:

Ecuación 8

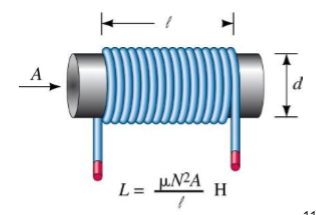


Ilustración 4. Dimensiones de la bobina

Por lo tanto, según lo obtenido en la *Ecuación 8*, la inductancia de una bobina es proporcional a la longitud del hilo de cobre, al área de la bobina y proporcional al cuadrado al número de vueltas. El valor de viene determinado por el núcleo de la bobina, puede ser un material ferromagnético o incluso aire.

## ESTADO DEL ARTE

### BOBINADORAS COMERCIALES

Existe una gran variedad de bobinadoras en el mercado con gran precisión. El precio de una bobinadora comercial ronda los 20.355 euros, un precio muy elevado que no compensa al Departamento para el uso que se le va a dar.



Ilustración 5. Bobinadora comercial

Puesto que para las aplicaciones que va a tener este dispositivo no necesitamos tanta precisión, queda justificado el desarrollo de este proyecto.

Especificaciones de esta bobinadora comercial:

* Rango de diámetros del hilo: 0.02 -2.00 mm
* Diámetro máximo de la bobina: 180 mm
* Longitud máxima de la bobina: 300 mm
* Diámetro del husillo de bobinado: 10 mm
* 2 rangos de velocidades: 0 - 1000 rpm y 0 - 4500 rpm
* Tolerancia de paso entre espiras: 0.01 mm
* Tolerancia de rotación: 2 grados

El presupuesto de la bobinadora se puede consultar en el apéndice 6.1.

### BOBINADORAS DIY

En la etapa de diseño de la bobinadora, se estudiaron distintas alternativas de bobinadoras DIY (Do it yourself o hazlo tú mismo) para tomar ideas y tener referencias. Algunos de los modelos de bobinadoras se exponen en los siguientes apartados.

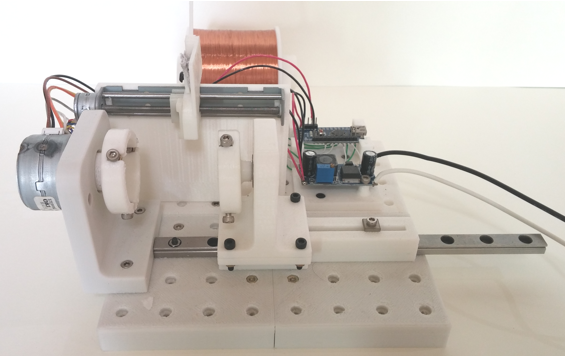


Ilustración 6. Bobinadora con estructura impresa en 3D

La característica principal de este modelo de bobinadora es que tiene una estructura compuesta completamente por piezas fabricadas mediante impresión 3D.

Este diseño modular permite devanar bobinas de distintas longitudes y, cambiando el soporte, distintos diámetros. La fabricación de la mayoría de las piezas por impresión 3D reduciría el coste del diseño.

Sin embargo, este proceso hubiese incrementado el tiempo de fabricación, pues son muchas piezas y las tolerancias entre ellas para garantizar el ajuste son muy delicadas y requieren mucha precisión por parte de la impresora 3D.

Este diseño utiliza un microcontrolador (placa Arduino nano) para el control de los motores paso a paso. Todos los diseños de piezas y códigos utilizados se encuentran subidos en GitHub [17].

### MODELO 2: Estructura de componentes metálicos



Ilustración 7. Diseño de bobinadora automática con estructura metálica

Esta bobinadora está pensada para devanar bobinas de longitudes pequeñas, de unos 2 – 3 cm, pero modificando algunas de sus piezas podrían devanarse bobinas de longitudes mayores.

Su estructura está compuesta por guías, perfiles de aluminio, que hacen de este modelo una alternativa sencilla, de precio moderado y fácil de obtener. Esta opción reduce el tiempo de fabricación y de montaje respecto a la bobinadora anterior.

Se valoró diseñar una bobinadora con unas características similares, en cuanto al diseño estructural y al movimiento.

Al estudiar en detalle cada uno de sus componentes, una gran parte de ellos se habrían tenido que comprar, como el sistema de guías, el husillo del bobinado, el soporte de la bobina suministradora, etc., lo que hubiese producido un aumento considerable en el coste final de la bobinadora. Además, este diseño no garantiza la modularidad o intercambiabilidad de las piezas ya que tenemos muy restringidas las medidas de cada pieza y esto también limita el rango de diámetros de bobinas que se pueden devanar.

En cuanto al control de los motores paso a paso, esta bobinadora también utiliza un microcontrolador, en concreto una placa Arduino MEGA, y funciona con el software Marlin, usualmente utilizado en las impresoras 3D, basado en G-Code.

El funcionamiento de este sistema y los componentes utilizados los explica en este vídeo [18].

## ESQUEMA DE LA MEMORIA

La memoria se compone de nueve capítulos y algunos apéndices:

* *Introducción:* Se definen algunos conceptos básicos sobre el funcionamiento de una bobina y se justifica el desarrollo del proyecto debido al alto coste de las bobinadoras comerciales.
* *Objetivos:* en este apartado se definen los objetivos principales y secundarios del TFG.
* *Solución técnica:* Se exponen los pasos que se han seguido para alcanzar los objetivos del proyecto. También se enumeran los problemas que han surgido durante su desarrollo y cómo se han solucionado.
* *Conclusiones:* En este capítulo se discuten los resultados obtenidos en el proyecto, los objetivos alcanzados y líneas futuras para la mejora del prototipo.
* *Bibliografía.*
* *Apéndices:* Está compuesto por varios subcapítulos en los que se añade información sobre el código utilizado para la programación de los motores, una BOM (Bill of materials) y presupuesto del material empleado y el mapa de pines de conexión entre la placa Arduino y la CNC Shield.

# OBJETIVOS

El objetivo principal de este proyecto es el diseño y la fabricación de una bobinadora automática para devanar bobinas con dimensiones variables definidas por el usuario (longitud y número de capas).

Para cumplir este objetivo principal, se establecen otros objetivos secundarios:

1. El dispositivo debe ser **económico** y sencillo, utilizando tecnologías de bajo coste y fácil adquisición, como hardware y software libre e impresión 3D.
2. El dispositivo debe ser **totalmente autónomo**, no depender de un ordenador ni de otros dispositivos externos, para facilitar su utilización.
3. El dispositivo debe tener la capacidad de **adaptarse a distintos tipos de bobinas** mediante el uso de componentes modulares intercambiables para las bobinas de mayores dimensiones.
4. El **devanado** de la bobina debe ser lo suficientemente **preciso y resistente**.

## ESTRUCTURA DEL PLAN DE TRABAJO

### Definición de especificaciones

En primer lugar, se establecieron los objetivos del proyecto, definiendo las características generales del sistema de bobinado y los requisitos que debía satisfacer.

### Estudio de alternativas

Una vez definidas las especificaciones, se contemplaron distintas alternativas para la construcción de la bobinadora. Finalmente, se eligieron las características que cumplían con los objetivos establecidos, reducían el tiempo y coste de fabricación y la complejidad del diseño.

### Adquisición de materiales

En esta fase se adquirieron todas las piezas, componentes tanto electrónicos como mecánicos, y herramientas software necesarios para el desarrollo del proyecto.

### Documentación de los avances

Con el avance del proyecto, se iban documentando los procedimientos utilizados, explicando cada problema que surgía y la solución adoptada. Esta etapa abarca prácticamente toda la realización del proyecto.

### Aprendizaje del software

Para lograr los objetivos definidos inicialmente fue necesario el aprendizaje de distintos programas y plataformas como FreeCAD, GitHub, Arduino y Repetier Host. También fue necesaria la familiarización con componentes electromecánicos con los que no se había trabajado anteriormente.

### Diseño mecánico e impresión de las piezas

En esta fase se lleva a cabo el diseño de la parte mecánica con todos los componentes y las piezas imprimibles, optimizándolas en cantidad de material y espacio utilizado. Posteriormente se imprimen en la impresora 3D.

### Montaje del sistema

Una vez impresas todas las piezas y adquiridos el resto de los materiales, se comienza a montar la estructura. Posteriormente, cuando se programa el código de los motores, se montan los soportes con los elementos electrónicos.

### Programación del control de los motores paso a paso

El control de los motores se programa en la placa Arduino UNO, junto con el módulo de expansión CNC Shield y los drivers para el control de corriente que llega a los motores.

### Calibración del sistema

Tras el montaje del sistema, se realiza una calibración para garantizar el tensionado del hilo y el ajuste correcto de los tornillos para evitar vibraciones. También se modifica la relación de movimientos entre motores para ajustar la separación entre espiras durante el bobinado.

### Elaboración de la memoria

Finalmente, para garantizar el entendimiento, se estructuró y se redactó toda la información documentada anteriormente en la memoria y los pasos que se siguieron, añadiendo las conclusiones obtenidas de este proyecto.

# SOLUCIÓN TÉCNICA

En este capítulo se describe el prototipo de bobinadora que se ha diseñado para este proyecto.

En primer lugar, se describen de forma general las características principales del prototipo y los distintos subsistemas que lo componen.

A continuación, se explica en cada apartado en detalle el funcionamiento de cada uno de los subsistemas que integran la bobinadora, las diversas alternativas que se plantearon en cuanto a materiales y componentes y, finalmente, las soluciones adoptadas.

Las elecciones que se han tomado se han hecho en base a construir un sistema de bobinado automático de la forma más eficiente posible: con un diseño que permita el devanado de bobinas con distintos tamaños, a un coste y un tiempo de desarrollo no muy elevados.

## Descripción general del prototipo



Ilustración 8. Sistema de bobinado automático

De forma general, el sistema de bobinado consta de las siguientes partes *(Ilustración 9):*

* Un **sistema electrónico de control**, el “cerebro” del sistema de bobinado
* Un **sistema electrónico de potencia** que ejecuta las órdenes del sistema de control
* Un **sistema mecánico** que proporciona el movimiento
* La **alimentación** del sistema de control y de la electrónica de potencia

**SISTEMA ELECTRÓNICO DE POTENCIA**

**SISTEMA MECÁNICO**

**ALIMENTACIÓN**

**SISTEMA DE ELECTRÓNICO CONTROL**

Ilustración 9. Componentes del sistema de bobinado

## Movimientos del sistema

Los movimientos realizados por este sistema de bobinado automático *(Ilustración 8)* son dos: de traslación y de rotación, y se realizan de forma simultánea.

### Movimiento de traslación

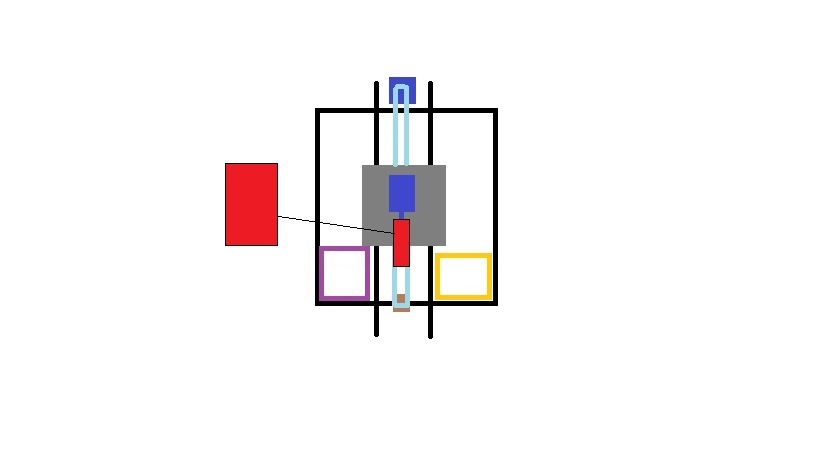
El movimiento de traslación viene proporcionado por un motor, denominado ***Slider*** *(Ilustración 10),* sujeto a la estructura de forma que el movimiento de rotación se produzca en el plano horizontal de la estructura.

Este motor transmite su movimiento al soporte al que está sujeta la bobina a través de una correa. La traslación de la bobina se produce de forma simultánea a la rotación para evitar la superposición entre espiras contiguas.

### Movimiento de rotación

El movimiento de rotación viene determinado por un motor colocado en la dirección horizontal, denominado ***Winder*** *(Ilustración 10).*

La bobina va ensamblada al motor, el cual le transmite su movimiento. Como consecuencia, la bobina a devanar tira del hilo de la bobina proveedora, transmitiéndole a su vez el movimiento de rotación.



ALIMENTACIÓN

SLIDER

WINDER

ARDUINO

BOBINA PROVEEDORA

BOBINA A DEVANAR

SOPORTE MÓVIL

Ilustración 10. Esquema general de la bobinadora

**\*\*\*ACTUALIZAR ESQUEMA**

### Relación entre movimientos

Puesto que los dos movimientos mencionados anteriormente se dan de forma simultánea, se ha de establecer una relación entre los movimientos de rotación y traslación para asegurar que el devanado de la bobina se realiza correctamente.

Las siguientes dimensiones serán seleccionadas por parte del usuario antes de iniciar el bobinado:

1. Dimensiones de la bobina:

***D*** – Diámetro de la bobina

***L*** – Longitud de la bobina

Cada vez que el motor ***Winder*** complete tres vueltas, el motor ***Slider***girará de forma que avance 0,0125‬ mm, para evitar la superposición de hilos. Este ajuste podría cambiar si se devanan hilos con un diámetro superior a 0,8 mm.

El cálculo sobre la resolución de la correa se encuentra en el apartado 3.3.2 y el de la resolución de los motores paso a paso en el apartado 3.5.1.

El motor ***Slider*** cambia de sentido cada vez que se completa una capa para optimizar el movimiento.

## Sistema mecánico

En este apartado se expone el procedimiento que se ha seguido para el montaje de la estructura y de todos los componentes mecánicos necesarios para el sistema de bobinado.

* Estructura
* Transmisión del movimiento
* Componentes impresos en 3D
* Componentes adquiridos

### Estructura

Para la construcción de la estructura se contemplaron distintas alternativas. Inicialmente se estudió la posibilidad de montar una estructura hecha con distintas piezas fabricadas por impresión 3D. Esta alternativa era muy atractiva en cuanto a coste, pero el tiempo que conllevaba el diseño de cada una de las piezas, sumado al tiempo de impresión hacían que la duración del proyecto se prolongase demasiado.

Finalmente, se decidió montar una estructura formada por perfiles de aluminio. El montaje sería mucho más rápido y permitiría hacer modificaciones en cuanto a las dimensiones empleando poco tiempo. Se utilizaron perfiles de aluminio con una sección de 15x15 mm, con las siguientes longitudes:

* 8 perfiles de 200 mm
* 4 perfiles de 100 mm

Para el ensamblaje se utilizaron 12 escuadras, tornillos DIN 912, M3 y tuercas DIN 934, M3 [6.2].

### Transmisión del movimiento

* **Movimiento traslacional:**

Para realizar el movimiento de traslación de la bobina se montaron dos ejes de 8 mm de diámetro, de acero inoxidable 304, calidad H9 [6.2].

Para convertir el movimiento de rotacional en traslacional del motor ***Slider***, se ha utilizado una correa dentada GT2 con 2 mm de espaciado entre dientes, un avance de 20 dientes por vuelta y una anchura de 6 mm.

Los motores paso a paso utilizados tienen una resolución de 1,8/paso, por lo tanto, tienen que dar 200 pasos para completar una vuelta.

La resolución de avance de la correa para este tipo de motor paso a paso se calcula de la siguiente forma:

Para garantizar la estabilidad del giro y evitar vibraciones se ha utilizado una resolución de 1/16 de paso en los motores, para completar una vuelta tendría que dar 3200 pasos:

Sin embargo, esta resolución no es real ya que el motor no es capaz de pararse cada 1/16 de paso, por lo tanto, se avanza cada 16 dieciseisavos de paso para que sea estable.

Para facilitar el movimiento del soporte principal sobre los ejes se utilizan cuatro rodamientos lineales tipo Lm8uu.

* **Movimiento rotacional:**

El movimiento de rotación para el devanado lo proporciona el motor ***Winder.*** La resolución del motor paso a paso es de **1,8/paso**.

En este movimiento también se ha utilizado una resolución de 1/16 de paso en el motor:

Como ya se ha explicado en el apartado anterior, esta resolución no es real y, por lo tanto, el ángulo mínimo de giro que se puede conseguir es de 1,8 cada 3200 dieciseisavos de paso.

### Impresión 3D

Según se iban identificando las necesidades para el correcto funcionamiento del prototipo, se fueron diseñando componentes en FreeCAD.

Una vez diseñadas las piezas, se fabricaron en una impresora 3D utilizando el software Repetier Host.

#### FreeCAD

FreeCAD es un software de código abierto de diseño de componentes en tres dimensiones. Esta aplicación permite diseñar objetos 2D a partir de bocetos o “sketches”, así como transformar un diseño 2D en una geometría 3D por extrusión, revolución y otras funcionalidades incluidas.

Una ventaja de este software es que permite el diseño paramétrico de los elementos, es decir, cada elemento diseñado es tratado como un objeto el cual no es definido únicamente por sus coordenadas espaciales (x, y, z), sino también por sus parámetros. Gracias a esta característica se han podido reutilizar algunos elementos ya diseñados y parametrizados por Felipe Machado, disponibles en su GitHub [8], tan solo ajustando sus dimensiones a las necesidades de la estructura del sistema de bobinado.

La mayoría de las piezas impresas en 3D utilizadas en este proyecto han sido diseñadas en FreeCAD.

### Repetier Host

Es un software de impresión 3D que permite la transformación de los archivos en formato *.stl* a *.gcode* para que la impresora pueda interpretarlos.

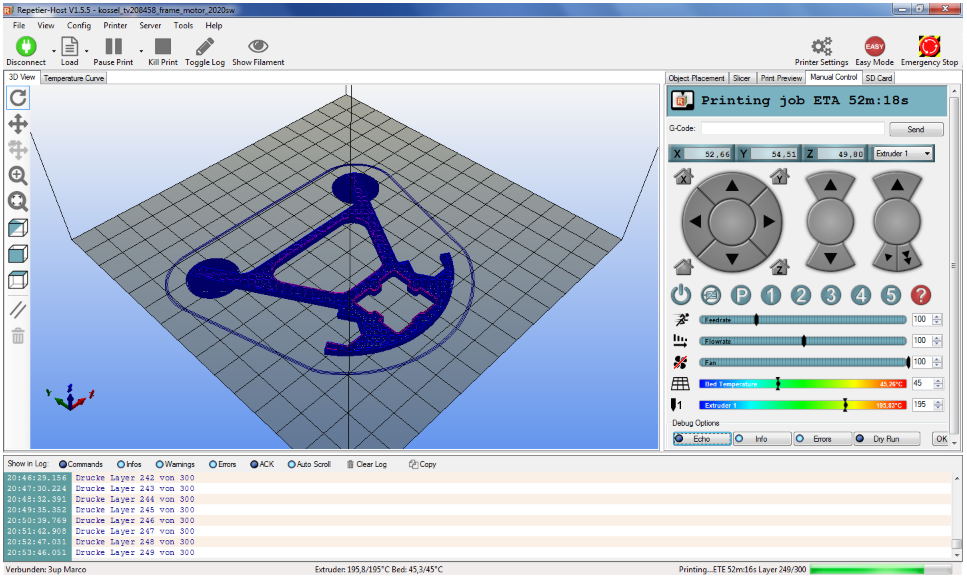


Ilustración 11. Interfaz de Repetier Host

Repetier Host es compatible con los firmwares de la mayoría de las impresoras 3D, nos permite controlar variables como la temperatura de la cama, velocidad de impresión y la cantidad de relleno. Además, esta herramienta gratuita está disponible en sistemas operativos Windows, Linux y Mac [9].

Para llevar a cabo la impresión de una pieza se han seguido los siguientes pasos:

1. **Posicionamiento del objeto**: Se importa uno o varios objetos en 3D en formato .stl y se coloca en la posición que más favorezca la impresión. Por ejemplo, para garantizar una buena impresión colocaremos la zona más plana como base y la zona que tenga más huecos de forma que queden paralelos al suelo. Aparte de rotarlos, escalarlos y girarlos, también es posible duplicarlos e imprimirlos simultáneamente. En el caso de los huecos, Repetier genera una estructura interna provisional para evitar que se deformen y posteriormente podremos retirarla.
2. **Slice**: Se secciona la pieza en “rebanadas” en función del relleno seleccionado. En las piezas que no soporten demasiadas tensiones podremos reducir el relleno para aumentar la velocidad de impresión
3. **Vista previa**: Se comprueban si los parámetros seleccionados para la impresión son los correctos (temperatura de la cama, temperatura del extrusor, relleno, etc.)
4. **Impresión**: Es el paso final, una vez comprobado todo se procede a imprimir la pieza objetivo.

### Componentes impresos

El hecho de diseñar algunos de los componentes en vez de comprarlos facilitó la adaptabilidad de las piezas a las características de la estructura y del prototipo.

Sin embargo, algunas piezas conllevaron un tiempo de desarrollo superior al esperado debido a problemas de impresión, mala elección de las tolerancias entre piezas ensambladas, y optimizaciones de diseño.

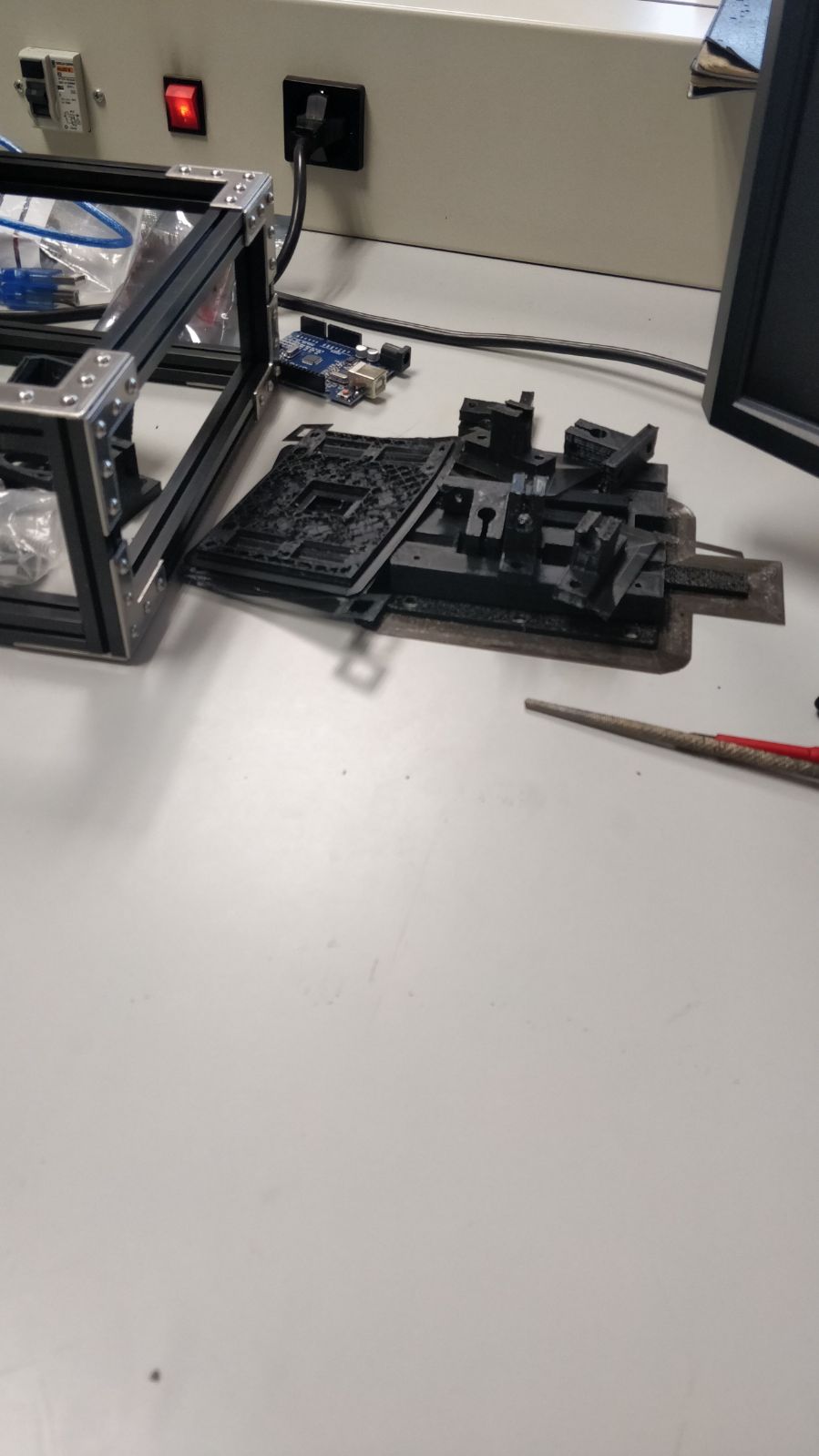


Ilustración 12. Piezas impresas en 3D fallidas

Todas las piezas se encuentran subidas en el repositorio de GitHub [7], en *coil-winder/freecad-components*.

1. **Soporte principal y carcasas**

Esta pieza es la piedra angular del sistema de bobinado, actúa como deslizadera durante en el movimiento horizontal. Se dedicó un tiempo de diseño acorde con su importancia en el sistema de bobinado, pero tuvo que modificarse según se fueron imprimiendo distintas versiones del diseño por las razones que se explican a continuación:

1. En un primer momento se pensó en el soporte principal como **dos piezas idénticas y simétricas** las cuales se moverían a lo largo de los ejes con el movimiento de la correa.
2. Se tuvieron que reducir sus dimensiones para optimizar el material utilizado y reducir el tiempo de impresión. Para ello, se reestructuró el componente simplificándolo lo máximo posible **a un soporte principal y dos carcasas** que contendrían los ejes e irían situados en la parte inferior.
3. Una vez fabricada, cuando se intentó ensamblar con otros componentes de la estructura se encontraron **problemas con las tolerancias utilizadas** y las piezas no terminaban de encajar.
4. Además, debido a la larga duración de la impresión de la pieza tuvo que repetirse en numerosas ocasiones, ya que **quedaba incompleta por diversos problemas con la impresora** utilizada.

Tras cinco impresiones la pieza se fabricó satisfactoriamente y el diseño final es el que se muestra en la *Ilustración 14,* visto desde la parte inferior (imagen de la izquierda), y desde la parte superior (imagen de la derecha, posición real en que va montada en la estructura).

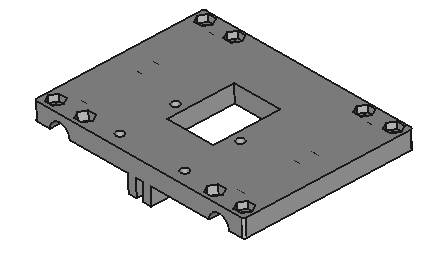
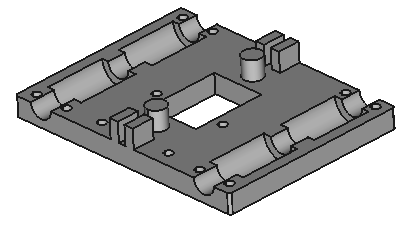


Ilustración 13. Soporte principal

Como se puede apreciar en la imagen, en los extremos del soporte se encajan dos rodamientos que permiten el deslizamiento del soporte a lo largo de los ejes horizontales. En la parte central se puede observar un cilindro y dos prismas rectangulares en los cuales se encaja la correa que proporciona el movimiento horizontal.

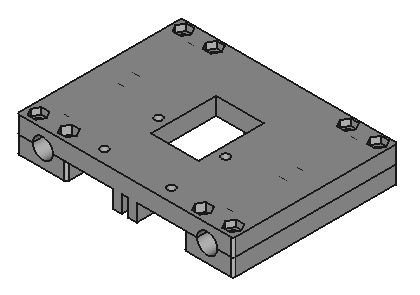
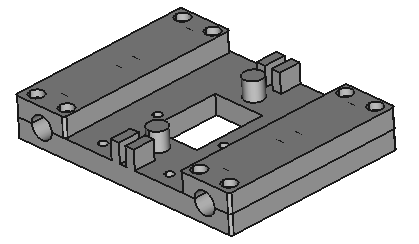


Ilustración 14. Conjunto del soporte principal y las carcasas

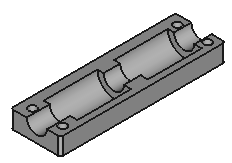
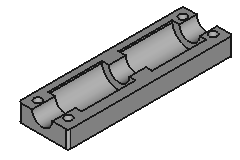


Ilustración 15. Carcasas del soporte principal

1. **Soportes de los motores**

Los motores ***Winder*** y ***Slider*** van sujetos a la estructura y al soporte principal gracias a la pieza mostrada en la *Ilustración 20*, diseñados también por Felipe Machado *[8].*

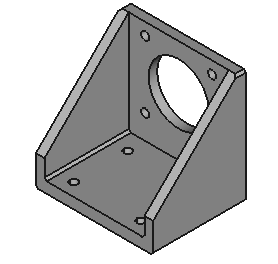


Ilustración 16. Soporte de los motores

El motor ***Slider*** va anclado a la estructura en posición vertical, y el motor ***Winder*** va sujeto al soporte principal sobre cuatro módulos alzadores en posición horizontal *(Ilustración 21)*.

1. **Módulos alzadores**

Se diseñaron cuatro módulos alzadores cuya función es evitar que la bobina roce con la correa o con el soporte principal y que sea posible el devanado de bobinas de mayores diámetros.

El hecho de ensamblar estos módulos a la estructura puede producir un problema de vibraciones durante el movimiento, por eso es muy importante que los tornillos que los sujetan estén bien ajustados.

Si en un futuro se quisiesen devanar bobinas de mayores dimensiones tan solo habría que aumentar el espesor de esta pieza para levantar el motor ***Winder***.

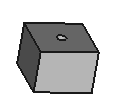


Ilustración 17. Módulo alzador

El ensamblaje de todos los componentes que van sobre el soporte principal ***(A, B, C)*** se muestra en la *Ilustración 23*:

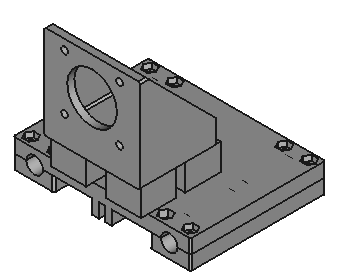


Ilustración 18. Conjunto del soporte principal con carcasas, soporte motor y módulos alzadores

1. **Soportes de los ejes**

Para la sujeción de los ejes a la estructura, se han utilizado cuatro soportes idénticos al que se muestra en la *Ilustración 17:*

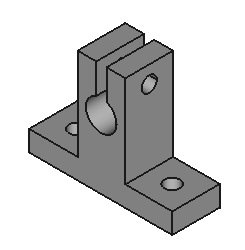


Ilustración 19. Soporte del eje

1. **Tensionador de la correa**

Para asegurar el tensionado de la correa, se imprimió la pieza que se muestra en la *Ilustración 18.* Se compone de una estructura externa que va anclada a los perfiles de aluminio, y en su interior se encaja una pieza cuya posición se puede modificar apretando un tornillo y una polea loca alrededor de la cual se coloca la correa. Gracias a este componente podemos variar la tensión de la correa simplemente apretando o aflojando un tornillo. Esta pieza ha sido diseñada por Felipe Machado, está disponible en repositorio de GitHub. [8]

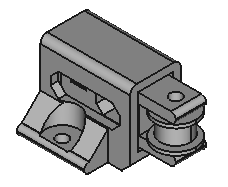


Ilustración 20. Tensionador de la correa

1. **Soporte del final de carrera**

Para la sujeción del final de carrera a uno de los ejes se ha diseñado la pieza que se muestra en la *Ilustración 19.*

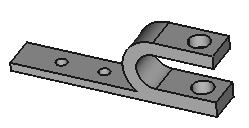


Ilustración 21. Soporte del final de carrera

1. **Tensionador del hilo**

La funcionalidad principal de esta pieza es mantener el hilo tenso durante el bobinado.

Adicionalmente, esta pieza proporciona una gran estabilidad durante el devanado, ya que al llevar directamente el hilo desde la bobina proveedora, esta se movía y hacía que el hilo no se posicionase correctamente.

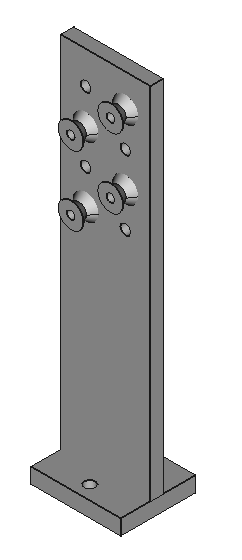


Ilustración 22. Tensionador del hilo

La pieza está formada por un pilar principal con orificios y unos semiconos en los que se enrolla el hilo. Se puede cambiar la posición de los semiconos a otros orificios, ya que los semiconos se van sujetos a la estructura con tornillos.

El hilo, que viene de la bobina proveedora, se enrolla en zigzag alrededor de los semiconos y de ahí va a la bobina objetivo.

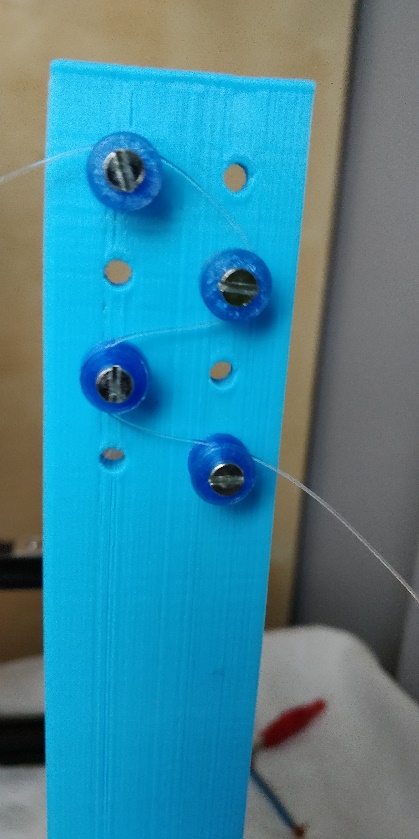


Ilustración 23. Colocación del hilo en zigzag

La adición de esta pieza al prototipo supuso un gran cambio en la calidad del bobinado, garantizando una buena colocación del hilo.

1. **Soporte del carrete**

Esta pieza ensambla el carrete de la bobina con el cilindro del motor. Está diseñado para el carrete de una bobina tipo **ETD59/31/22**, esta pieza tendría que ser modificada en función del carrete utilizado.

Esta pieza se ajusta al motor introduciendo un tornillo por el orificio superior y una tuerca en la apertura cuadrada de la parte frontal.

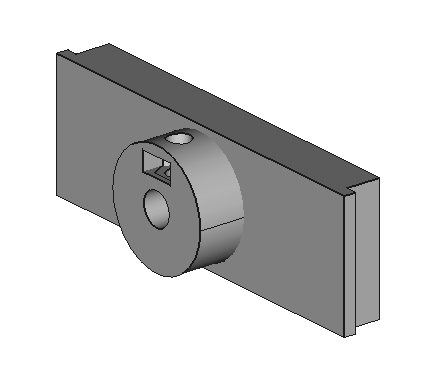


Ilustración 24. Soporte del carrete

1. **Soporte de la bobina proveedora**

Esta pieza soporta la bobina proveedora, la que nos proporciona el hilo. Este diseño es reutilizable para distintos diámetros de bobina, ya que no tiene que ir encajada en la pieza, simplemente gira libremente alrededor del cilindro central cuando tira del hilo.

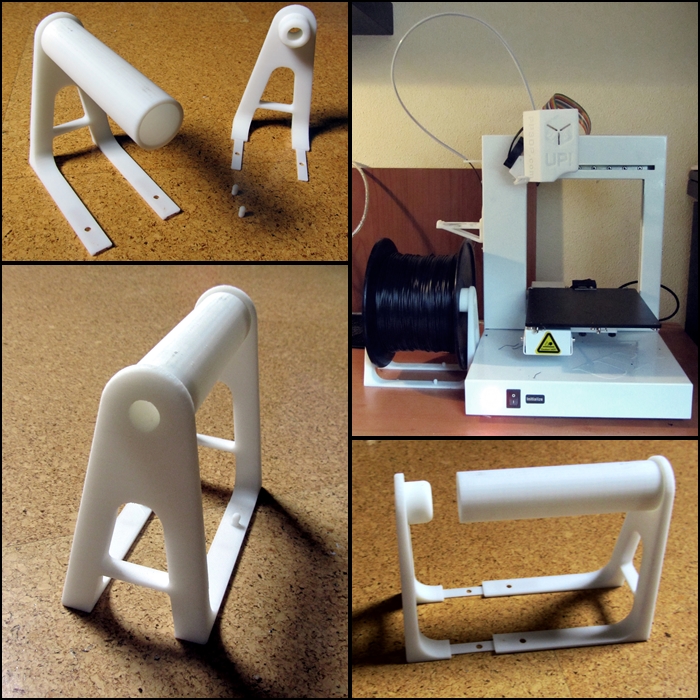


Ilustración 25. Soporte de la bobina proveedora

El soporte se compone de dos piezas laterales y un cilindro central que las une:

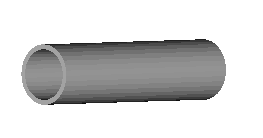
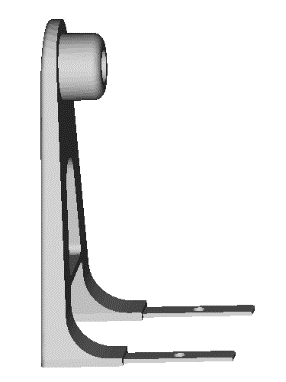
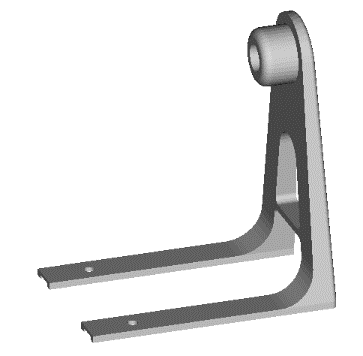


Ilustración 26. Subcomponentes soporte bobina

### Carrete de la bobina objetivo

El carrete que se va a utilizar en este proyecto es el de una bobina tipo **ETD59/31/22** que será utilizada en un transformador en uno de los proyectos del Área de Tecnología Electrónica. Las especificaciones completas sobre este modelo de carrete se encuentran disponibles en [19].



Ilustración 27. Carrete de la bobina objetivo

## Sistema electrónico de control

El sistema de control está formado por dos microcontroladores que se comunican por el protocolo I2C y actúan como *master – slave*. En uno de ellos está programada la interfaz de usuario donde se introducen los parámetros iniciales (*master*) y en el otro están programados los movimientos para el devanado de la bobina (*slave*). En este caso los microcontroladores seleccionados han sido dos placas Arduino UNO.

En los siguientes apartados se explica en detalle la metodología que se ha seguido para desarrollar este sistema y cada uno de los actores que forman parte de él.

### Arduino UNO

Durante el establecimiento de las especificaciones, se estudiaron distintas alternativas teniendo en cuenta características como la capacidad procesamiento, la cantidad de periféricos necesarios, facilidad de programación, los módulos de expansión compatibles y el coste. La solución que mejor se adaptaba a estos requisitos fue un microcontrolador, en concreto, una placa Arduino.

Una vez tomada esta decisión, se decidió qué placa Arduino utilizar entre la variedad de placas que ofrece este fabricante. Valorando todas las alternativas en cuanto a número de pines de entrada y salida y compatibilidad con otros componentes hardware, **la elección final fue una placa Arduino UNO**.

En la parte final del proyecto, para que la bobinadora fuese completamente autónoma, se decidió añadir una pantalla LCD con botones compatible con Arduino UNO para que el usuario pudiese introducir los parámetros iniciales de la bobina. Por la falta de pines en la primera placa Arduino y la facilidad de conexión en una nueva placa, **se decidió añadir otra placa Arduino UNO**.

Arduino UNO es una placa de código abierto basada en el microcontrolador ATmega328P de Microchip. Hay una gran comunidad trabajando con esta plataforma, lo que permite un fácil acceso a una gran variedad de documentación en caso de necesidad.

Su entorno de programación es multiplataforma, se puede ejecutar en sistemas operativos Windows, Mac OS y Linux y su programación está basada en C++, un lenguaje de fácil comprensión que permite escribir el código con un gran nivel de abstracción.

Una ventaja de esta placa con respecto a la de otros fabricantes es que se han diseñado una gran cantidad de componentes hardware compatibles con ella, como la placa de expansión CNC Shield para el control de los motores paso a paso y sus drivers, lo que facilitará la programación y la compatibilidad entre dispositivos.

Sus principales características son [11]:

* Microchip ATMega328P
* Puede ser alimentado por el cable USB (a través del ordenador, por ejemplo) o por una batería externa entre 7 y 20 voltios a través del puerto Jack.
* Tiene disponibles 14 pines de entradas/salidas digitales, 6 de los cuales pueden ser utilizados como salida PWM.
* Tiene 6 pines de entradas analógicas

**SCL**



**PINES DIGITALES**

**PINES ANALÓGICOS**

**PUERTO USB**

**PUERTO JACK ALIMENTACIÓN DC 7 – 12 V**

**ENTRADAS/SALIDAS VOLTAJE**

**BOTÓN RESET**

**MICROCHIP ATMEGA328P**

Ilustración 28. Esquema placa Arduino UNO

**LED INDICADOR**

**LED ESTADO TX/RX SERIAL**

**SOCKET ICSP**

**SDA**

#### Software Arduino IDE

Arduino IDE (Integrated Development Enviroment) es el software utilizado para la escribir y cargar el código la placa Arduino UNO escrito en el lenguaje C++.

Este software multiplataforma está escrito en Java y puede ser utilizado con cualquier placa Arduino. El IDE proporciona una gran variedad de bibliotecas, que facilitan la implementación de funciones, y algunos proyectos de ejemplo *(Ilustración 26).*

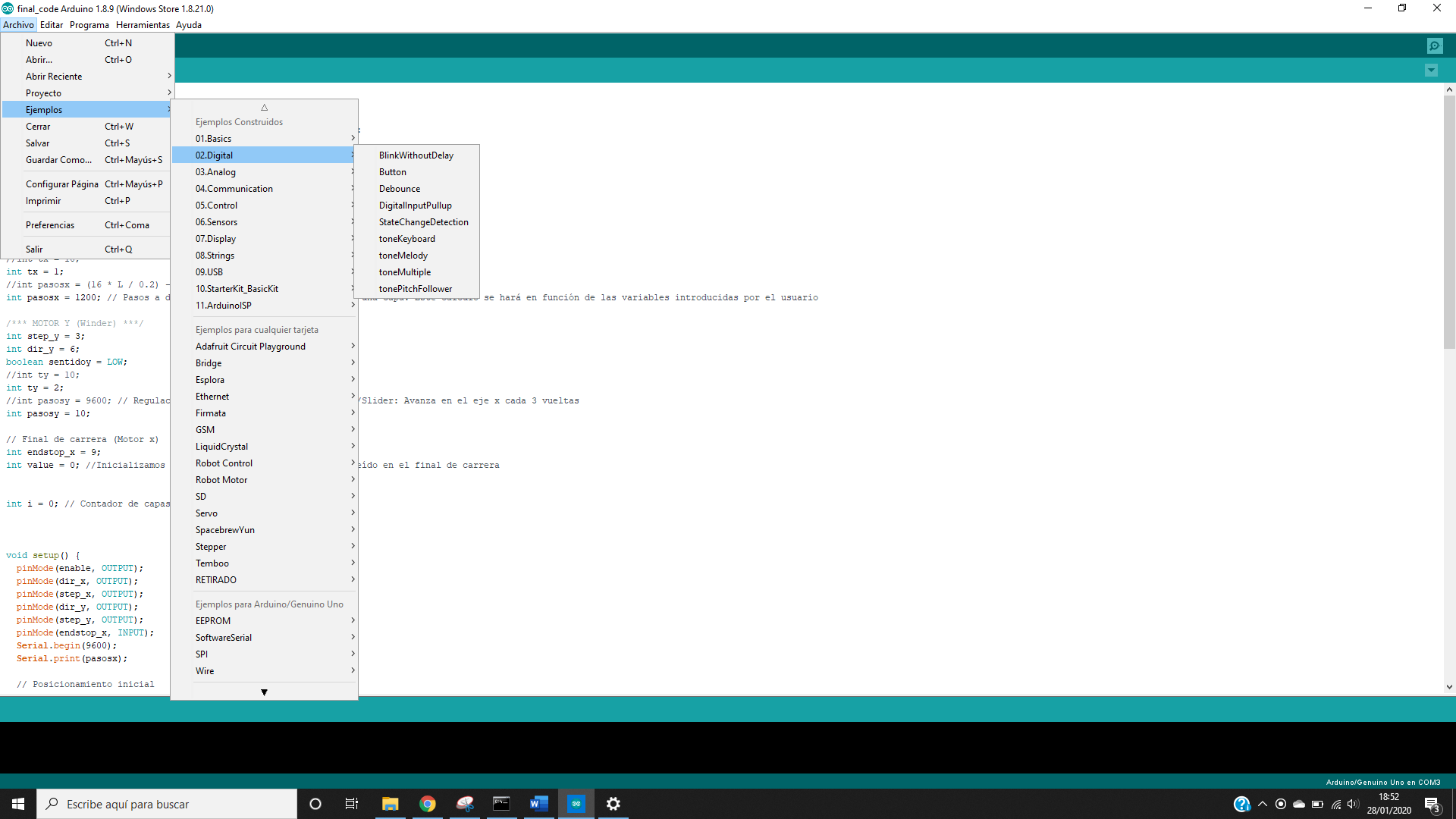


Ilustración 29. Proyectos de ejemplo

Para poder cargar el código en la placa es necesario seleccionar en *Herramientas* el tipo de placa que se va a utilizar y el puerto USB del ordenador al que está conectado *(Ilustración 26).*

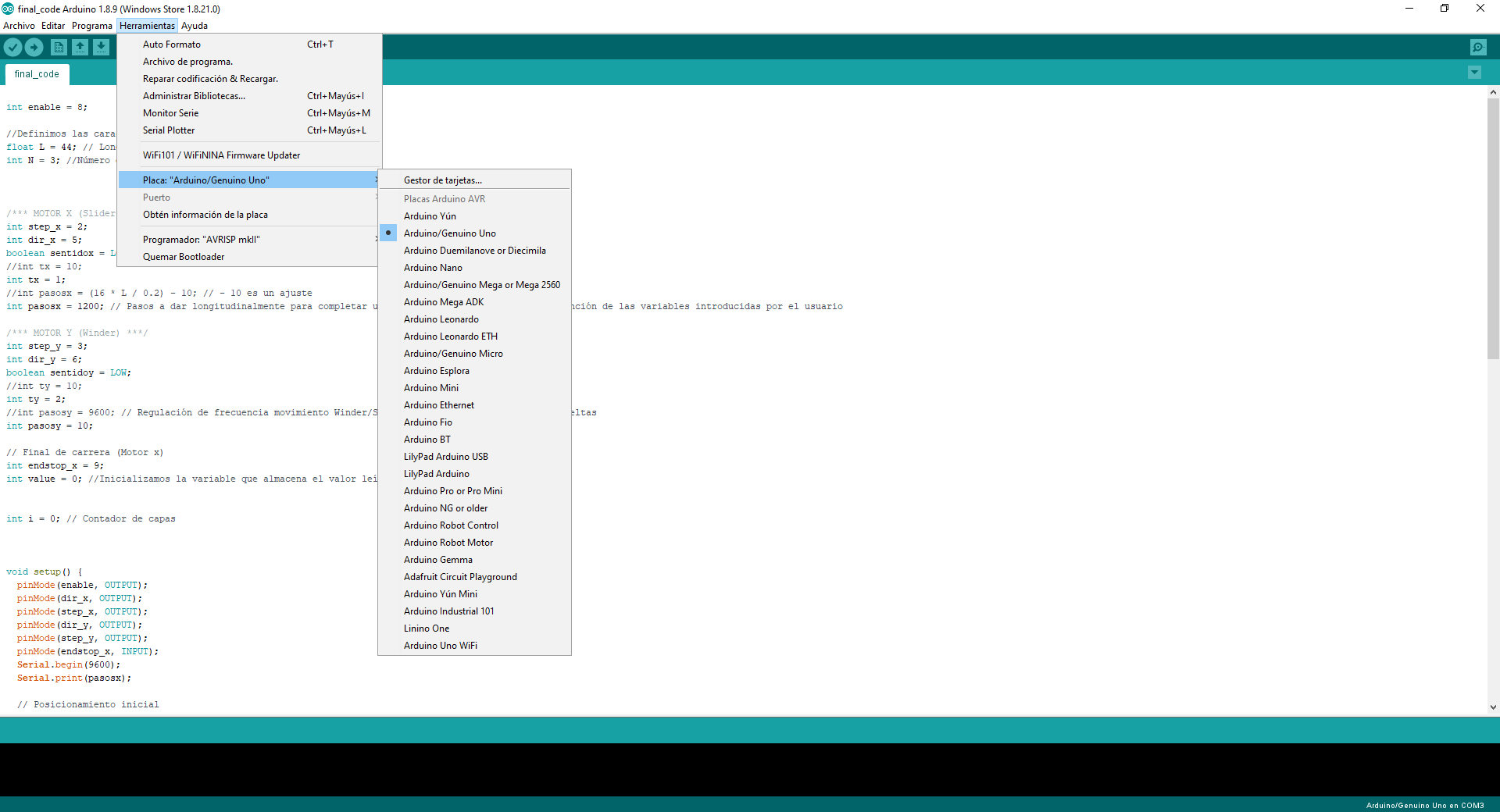


Ilustración 30. Seleccionar modelo de placa

**DESARROLLO DEL CÓDIGO**

**FUNCIONES BÁSICAS**

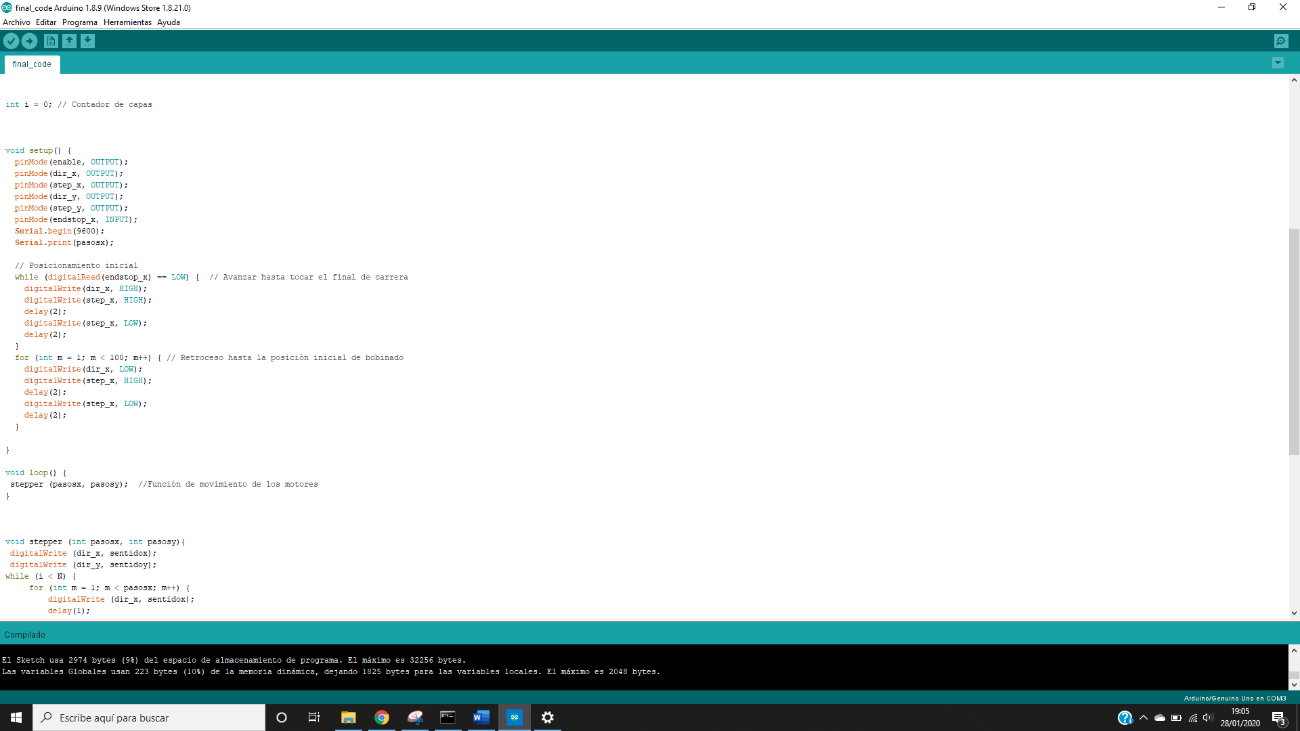


Ilustración 31. Esquema IDE

**CONSOLA**

Como se puede apreciar en la *Ilustración 27*, el IDE cuenta con una ventana principal donde se escribe el código, una consola en la parte inferior y una barra superior con botones con las siguientes funciones básicas:

* *Verificar*: Comprobar que no hay errores de sintaxis en el código
* *Cargar*: Transferir el programa a la placa Arduino
* *Nuevo*: Crear un sketch nuevo
* *Abrir*: Abrir un sketch existente
* *Guardar*: Guardar el sketch con el nombre deseado

**ABRIR**

**NUEVO**

**CARGAR**



Ilustración 32. Funciones botones

**VERIFICAR**

**GUARDAR**

La estructura general del código de Arduino se divide en dos partes:

* *void setup:* Función en la que se escribe la configuración inicial. Esta función se ejecuta una sola vez en el inicio del programa.
* *void loop:* Bucle principal del código, en esta sección se escriben o se llama a las funciones que se van a ejecutar de forma infinita en el programa.
* Se pueden escribir funciones adicionales a las que llamar en el bucle *void loop.*

### Interruptor de final de carrera

Los interruptores de final de carrera son componentes electrónicos que se sitúan al final de un recorrido para enviar señales que puedan modificar el estado de un circuito.

En el caso del sistema de bobinado, se utiliza un interruptor de final de carrera para detectar que la bobina que se está devanando ha llegado a la posición límite en su movimiento horizontal y cambiar el sentido en el motor ***Slider***.

En este sistema también se utiliza el interruptor de final de carrera como referencia para colocar la bobina en la posición inicial:

* 1. En primer lugar, la bobina avanza hasta detectar el final de carrera.
  2. Cuando lo detecta retrocede una distancia igual a la longitud de la bobina para iniciar el devanado en la dirección opuesta.
  3. Durante el bobinado, cuando llega a la posición límite vuelve a cambiar la dirección de devanado para continuar con la siguiente capa.

Estos interruptores pueden ser normalmente abiertos (NA) o normalmente cerrados (NC), en función de la operación que vayan a realizar al ser accionados.

En este sistema se ha utilizado un interruptor NC, ya que en el caso de que el interruptor esté mal conectado, no funcionaría porque no le llegaría tensión a la placa Arduino.

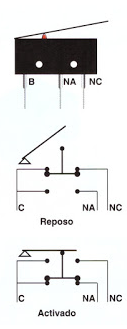


Ilustración 33. Esquema de funcionamiento del final de carrera (NC)

### Descripción del sistema

Este apartado incluye el diseño del sistema de control programado en la placa Arduino incluyendo la interfaz gráfica con el usuario.

**2**

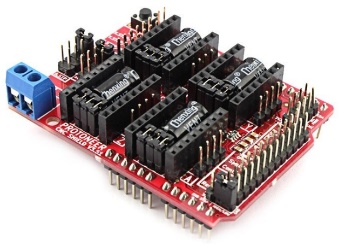
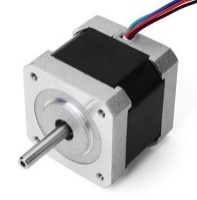
**1**

**L: Longitud bobina**

**D: Diámetro bobina**

**I2C**

***slave***



***master***

***lcd\_menus\_master.ino***

***winding\_slave.ino***

**USUARIO**

**L, D**

**L, D**

Ilustración 34. Descripción general del sistema

**3**

1. El usuario introduce los datos de longitud y diámetro de la bobina que se va a devanar. Esta información es procesada por la placa Arduino que actúa como *master*. El código utilizado en este proceso es *lcd\_menus\_master.ino*.
2. La placa *master* se comunica con la placa *slave* por el protocolo de comunicación I2C para mandarle los parámetros introducidos por el usuario.
3. La placa slave recibe la información y calcula los pasos a dar por los motores ***Winder*** y ***Slider*** en función de esos datos. El código utilizado en este proceso es *winding\_slave.ino.*

Todos los códigos desarrollados para este proyecto se encuentran disponibles en el repositorio de GitHub [7], en *coil-winder/arduino-code*.

#### LCD Keypad Shield

Se ha utilizado una pantalla LCD de 2x16 con botones integrados para la introducción de parámetros de la bobina por parte del usuario e iniciar el bobinado.

Se ha seleccionado este tipo de display por su facilidad de uso, es un módulo de expansión que se coloca encima de la placa Arduino UNO, dejando libres los pines digitales del 0 al 7, y las puertas analógicas del A1 al A5, más el ICSP.

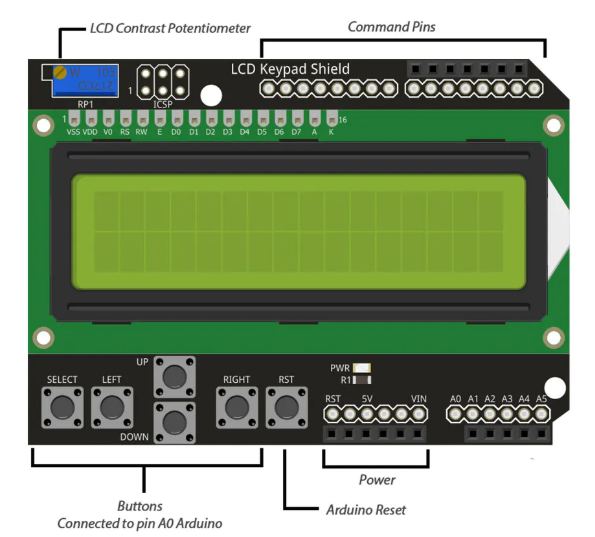


Ilustración 35. Pantalla LCD con botones

Además, se puede utilizar la librería en Arduino IDE de *LiquidCrystal* para una mayor facilidad de control de la pantalla.

La secuencia de pantallas que se mostrará al usuario será la siguiente:

**Mensaje inicial**

**Introducir longitud de la bobina objetivo**



**Introducir número de capas a devanar**

**Comenzar bobinado**

Ilustración 36. Secuencia de pantallas

En cuanto a los controles de los botones, las funcionalidades son las siguientes:

* El usuario podrá modificar la longitud de la bobina y de número de capas pulsando los botones de arriba y abajo.
* Para cambiar de menú podrá avanzar y retroceder pulsando los botones de izquierda y derecha.

Este modelo de utiliza 6 pines digitales para el control de la pantalla LCD:

Tabla 1. Pines de conexión de la pantalla LCD con botones

|  |  |
| --- | --- |
| Pin | Función |
| Analógico 0 | Control de los botones (Selección, arriba, abajo, derecha, izquierda y reset) |
| Digital 4 | DB4 |
| Digital 5 | DB5 |
| Digital 6 | DB6 |
| Digital 7 | DB7 |
| Digital 8 | RS |
| Digital 9 | Enable |
| Digital 10 | Control de la luz de fondo |

Para la comunicación por el protocolo I2C se han utilizado los pines A4 (SDA), A5 (SCL), GND y 5 V, como se explica en el siguiente apartado.

#### Bus I2C

El estándar de comunicación I2C (Inter-Integrated Circuit), también conocido como , se utiliza en la comunicación interna de dispositivos electrónicos como acelerómetros, brújulas, displays, etc.

El bus I2C sólo requiere de dos cables para su funcionamiento, uno para la señal de reloj (SCL), para mantener sincronizados a todos los dispositivos del bus, y otro para el envío de datos (SDA) [16].

Este estándar de comunicación tiene una arquitectura de tipo master-slave. El dispositivo maestro o master inicia la comunicación con los esclavos, y puede mandar o recibir datos de los esclavos. Los esclavos no pueden iniciar la comunicación, ni hablar entre ellos directamente.

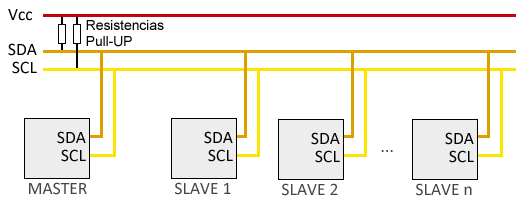


Ilustración 37. Arquitectura de tipo master-slave

Las placas Arduino disponen de soporte I2C por hardware, vinculado a ciertos pines. Los pines a los que van asociados varían dependiendo del modelo.

En la siguiente tabla se muestran los pines de conexión SCL y SDA en función del modelo de Arduino:

Tabla 2. Pines de conexión en función de la placa

|  |  |  |
| --- | --- | --- |
| Modelo | SDA | SCL |
| Arduino UNO | A4 | A5 |
| Arduino Nano | A4 | A5 |
| Mini Pro | A4 | A5 |
| Mega | 20 | 21 |

Adicionalmente, los últimos modelos de Arduino UNO disponen de dos pines independientes para utilizar SDA y SCK.

Para comunicar dos placas Arduino por el bus I2C es necesario utilizar la librería *Wire*. Algunas de las funciones básicas que nos ofrece esta librería son las siguientes:

* Wire.begin() → Inicializa el hardware del bus
* Wire.beginTransmission(address) → Comienza la transmisión
* Wire.endTransmission()→ Finaliza la transmisión
* Wire.requestFrom(address,nBytes) → Solicita un numero de bytes al esclavo en la dirección *address*
* Wire.available()→ Detecta si hay datos pendientes por ser leídos
* Wire.write()→ Envía un byte
* Wire.read()→ Recibe un byte
* Wire.onReceive(handler)→ Ejecuta la función de callback al recibir un dato
* Wire.onRequest(handler)→ Ejecuta la función de callback al solicitar un dato

En el sistema de bobinado se utiliza el estándar I2C para la comunicación de las dos placas Arduino.

* ***Arduino master*:**

Es la placa que se encarga de controlar la pantalla LCD con botones para controlar la bobinadora. Recibe los datos del usuario de longitud de la bobina y número de capas a bobinar y se los envía a la *Arduino slave*.

* ***Arduino* *slave*:**

Recibe una señal para iniciar el bobinado de la *Arduino master*, además de los parámetros introducidos por el usuario para calcular el número de pasos que tiene que dar cada motor.

En la siguiente imagen se muestra la conexión realizada entre las placas master y slave:

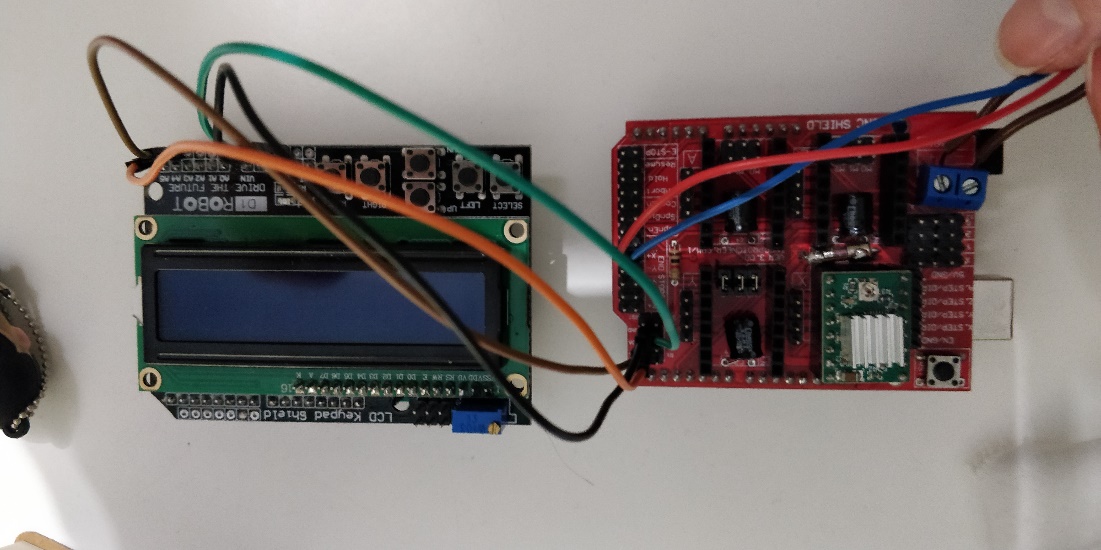
**SDA**

**GND**

**5V**

**SCL**

**A5 - SCL**



**Arduino slave**

**Arduino master**

**GND**

**5V**

**A4 - SDA**

Ilustración 38. Conexión entre placas Arduino UNO

|  |  |
| --- | --- |
| Arduino slave | Arduino master |
| SCL | A5 (SCL) |
| SDA | A4 (SDA) |
| 5V | 5V |
| GND | GND |

Tabla 3. Conexión entre las placas master - slave

La razón por la que en la placa slave se han utilizado los pines SCL y SDA directamente y en la placa master los pines equivalentes A4 y A5 es porque la accesibilidad de los pines analógicos al colocar el módulo LCD Keypad shield es mayor que la de los pines SCL y SDA.

#### Control del movimiento

Como se ha explicado en el apartado anterior, la placa Arduino slave es la encargada del control del bobinado. En el código winding\_slave.ino, disponible en el repositorio de GitHub del proyecto [7], se han programado los movimientos de los motores *Winder* y *Slider* para el bobinado.

La secuencia de movimientos para completar el bobinado es la siguiente:

1. **Posicionamiento inicial**

La bobina avanza hacia el final de carrera para tener una referencia en el espacio de la posición de la bobina. A partir de ahí retrocede hasta colocar el extremo más alejado en paralelo al tensionador del hilo.

El cálculo de los de paso que tiene que dar lo hace en función del parámetro de longitud (L) introducido por el usuario inicialmente:

El avance de la correa es de 0,2 mm en un paso, en la resolución de de paso serán 16. Para avanzar una distancia de *L* mm tendrá que dar los siguientes de paso:

1. **Bobinado**

El sentido de avance del bobinado de la primera capa siempre se hace en sentido opuesto al final de carrera.

FOTO CON FLECHA INDICANDO EL SENTIDO

Cuando el motor Slider ha avanzado una cantidad de pasos equivalente a la longitud de la bobina, comienza a bobinar en el sentido opuesto para iniciar la siguiente capa. Al terminar la capa llega a tocar el final de carrera y avanza en sentido opuesto al final de carrera.

FOTO CON FLECHA INDICANDO EL SENTIDO

Y así sucesivamente hasta completar las *N* capas.

Para ajustar el avance horizontal en función del giro, el motor Slider avanza de paso cuando el motor Winder ha completado tres vueltas. De esta forma se evita que se superpongan los hilos en una misma capa.

## Sistema electrónico de potencia

En este apartado se exponen los elementos utilizados en el sistema electrónico de potencia, como el uso de motores paso a paso, drivers y la placa de expansión CNC Shield.

Los movimientos en el sistema de bobinado vienen proporcionados por dos motores paso a paso. Para facilitar la programación de los movimientos de los motores se ha utilizado una placa de expansión CNC Shield V3.51 y unos drivers, Pololu A4988, que suministran la corriente necesaria a los motores.

La alimentación del sistema es suministrada por una fuente ATX de 12V convencional.

### Motores paso a paso

Los motores paso a paso son dispositivos electromecánicos que convierten una serie de impulsos eléctricos en desplazamientos angulares discretos o “pasos”.

La ventaja de este tipo de motores es que son precisos en el posicionamiento, son comúnmente utilizados en robots, drones, impresoras 3D, etc.

En este proyecto vamos a utilizar dos motores paso a paso bipolares NEMA 17 Wantai (*Ilustración 39*), es decir, contienen dos bobinas.

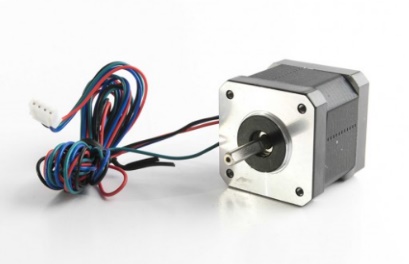


Ilustración 39. Motor paso a paso Nema 17 Wantai

De cada extremo de las bobinas sale un cable, en este caso: negro, verde, rojo y azul (*Ilustración 40*).

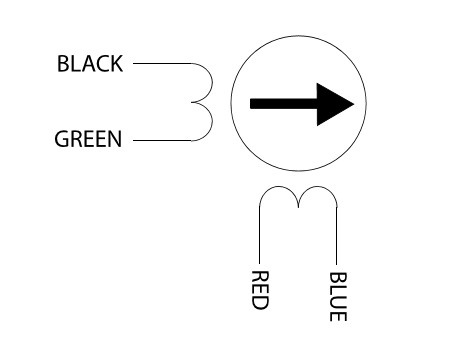


Ilustración 40. Cables de salida del motor paso a paso

Para alimentar el motor es necesario hacer un puente en H para cada bobina, conectando los cables negro y verde, y, por otro lado, rojo y azul (*Ilustración 26*).



Ilustración 41. Conexión de los cables de los motores paso a paso

El modelo de motor paso a paso seleccionado tiene una resolución de 1,8 grados por paso y, por lo tanto, tiene que dar 200 pasos para completar una vuelta.

Las especificaciones del modelo de motor paso a paso Nema Wantai 17 se encuentran en el apéndice 6.4.

### Placa CNC Shield V3.51

La placa CNC Shield V3.51 [10] es un módulo de expansión adicional a la placa Arduino UNO, que facilita el uso de los drivers de los motores paso a paso, en este caso son Pololu A4988.

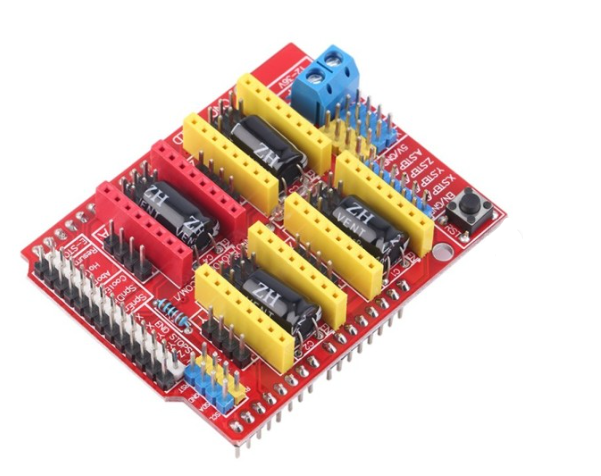


Ilustración 42. CNC Shield

La placa de expansión CNC Shield ofrece la posibilidad de controlar simultáneamente cuatro motores paso a paso. En este proyecto sólo utilizaremos dos, colocados en las posiciones X, el motor ***Slider***, e Y, el motor ***Winder*** *(Ilustración 43)*.

En la *Ilustración 44* se indican los pines de conexión de los motores en la placa CNC Shield.

**SDA**

**JUMPERS**

**FINAL DE CARRERA**

**MOTOR SLIDER**

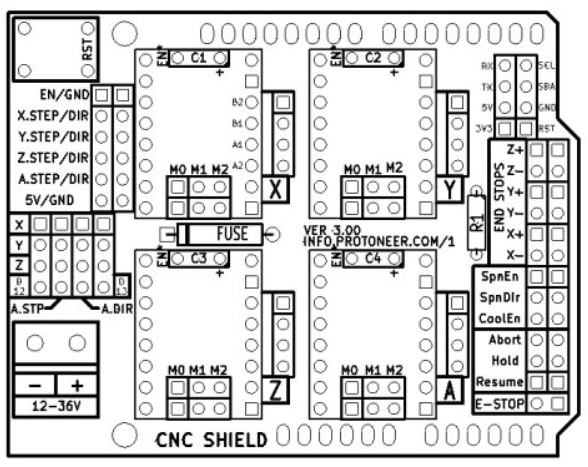


Ilustración 43. Esquema de conexiones en la placa CNC Shield

**MOTOR WINDER**

**ALIMENTACIÓN DEL MÓDULO**

**SCL**

**GND**

**5V**

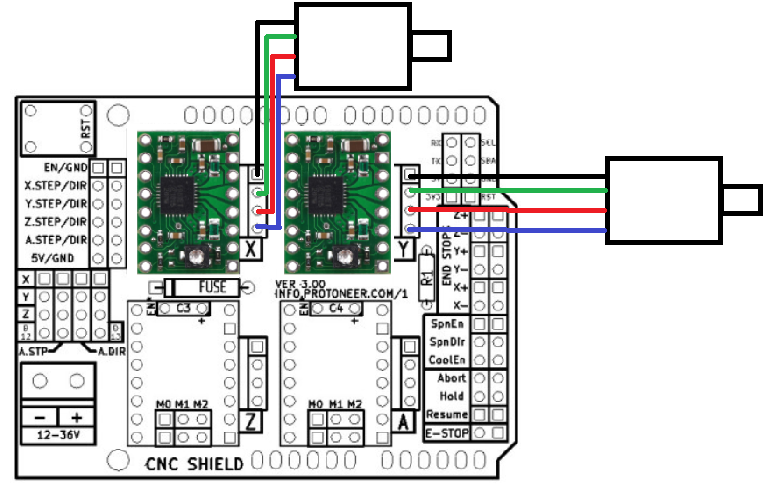


Ilustración 44. Esquema de conexión de los motores en la placa CNC Shield

La placa de expansión CNC Shield irá conectada sobre la placa Arduino UNO, los pines de la placa Arduino que controlarán las funciones del motor se muestran en la siguiente imagen:



**DIRECCIÓN SPINDLE**

**SPINDLE ENABLE**

**LÍMITE EJE Z\***

**LÍMITE EJE Y\***

**LÍMITE EJE X\***

**STEPPER ENABLE/DISABLE**

**DIRECCIÓN EJE Y**

**DIRECCIÓN EJE Z**

**DIRECCIÓN EJE X**

**PASOS EJE X**

**PASOS EJE Y**

**PASOS EJE Z**

Ilustración 45. Pines de conexión de la placa Arduino UNO

### Pololu A4988

El Pololu A4988 es una placa que permite controlar motores paso a paso bipolares y facilita una limitación de corriente ajustable, protección contra sobrecorriente y sobretemperatura, y cinco resoluciones de micro paso diferentes.

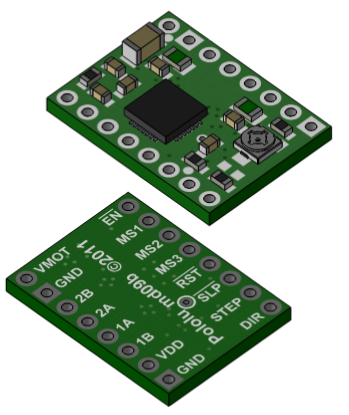


Ilustración 46. Pololu A4988

Para la alimentación de este controlador es necesario que se conecte un voltaje de suministro lógico (3 - 5,5 V) a través de los pines VDD y GND y un voltaje de suministro del motor (8 - 35 V) a través de VMOT y GND *(Ilustración 47).*

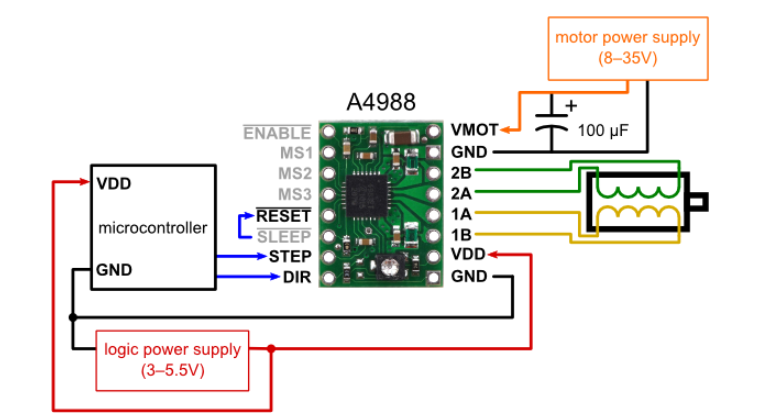


Ilustración 47. Esquema de conexiones del Pololu A4988

Para modificar la resolución de micropaso de los motores se colocarán jumpers en la CNC Shield en las posiciones MS1, MS2 y MS3. En la siguiente tabla se resumen las posiciones de los jumpers en función de la resolución deseada para el motor paso a paso:

Tabla 4 Resolución de los pasos en función de la posición de los jumpers

|  |  |  |  |
| --- | --- | --- | --- |
| MS1 | MS2 | MS3 | Resolución de micropaso |
| Bajo | Bajo | Bajo | 1 paso |
| Alto | Bajo | Bajo | paso |
| Bajo | Alto | Bajo | paso |
| Alto | Alto | Bajo | paso |
| Alto | Alto | Alto | paso |

En el sistema de bobinado **se ha seleccionado una resolución de de paso en ambos motores** para garantizar la estabilidad del bobinado y reducir las vibraciones.

Esta placa permite controlar un motor paso a paso bipolar con una corriente de salida de hasta 2 A por bobina si se le añade un disipador de calor a la placa Pololu (sin disipador sólo proporciona hasta 1A por bobina).

Antes de conectar los motores es preciso hacer un ajuste de corriente regulando el potenciómetro que lleva incorporado el Pololu a la corriente máxima por fase del motor Nema 17 *(Tabla 2)*.

Tabla 5. Tabla de especificaciones del motor Nema 17

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Modelo | Giro por paso | Voltaje nominal | Corriente nominal | Resistencia por fase | Inductancia por fase | Peso |
| 42BYGHW811 | 1, 8º | 3,1 V | 2,5 A | 1,25 ohm | 1,8 mH | 0,34 kg |

En la página web del fabricante se especifican los pasos a seguir para llevar a cabo la regulación de la corriente en función del modelo de motor paso a paso bipolar utilizado [14].

## Alimentación del sistema

Para alimentar el sistema, se utiliza una fuente de fuente de alimentación 500W ATX convencional.



Ilustración 48. Fuente de alimentación ATX

Este tipo de fuente de alimentación es fácil de obtener, pues se puede reutilizar de algún ordenador antiguo y tiene salidas a 3,3V, 5V, 12V.

El código de colores normalizado de las fuentes tipo ATX es el siguiente:

Tabla 6. Código de colores fuente ATX

|  |  |
| --- | --- |
| COLOR | Voltaje |
| Negro | GND |
| Naranja | 3,3 V |
| Rojo | 5 V |
| Amarillo | 12 V |

Conectamos la fuente ATX tanto a la placa Arduino slave como al módulo de expansión al que está ensamblado:

1. Alimentación de 12V a la placa Arduino *slave* a través de un conector Jack

El pin positivo del conector Jack se ha conectado al cable amarillo (12 V) y el negativo al cable negro (tierra).



Ilustración 50. Jack conectado a la fuente ATX

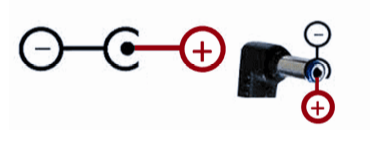


Ilustración 49. Conector Jack

1. Alimentación de 12V al módulo de expansión CNC Shield.

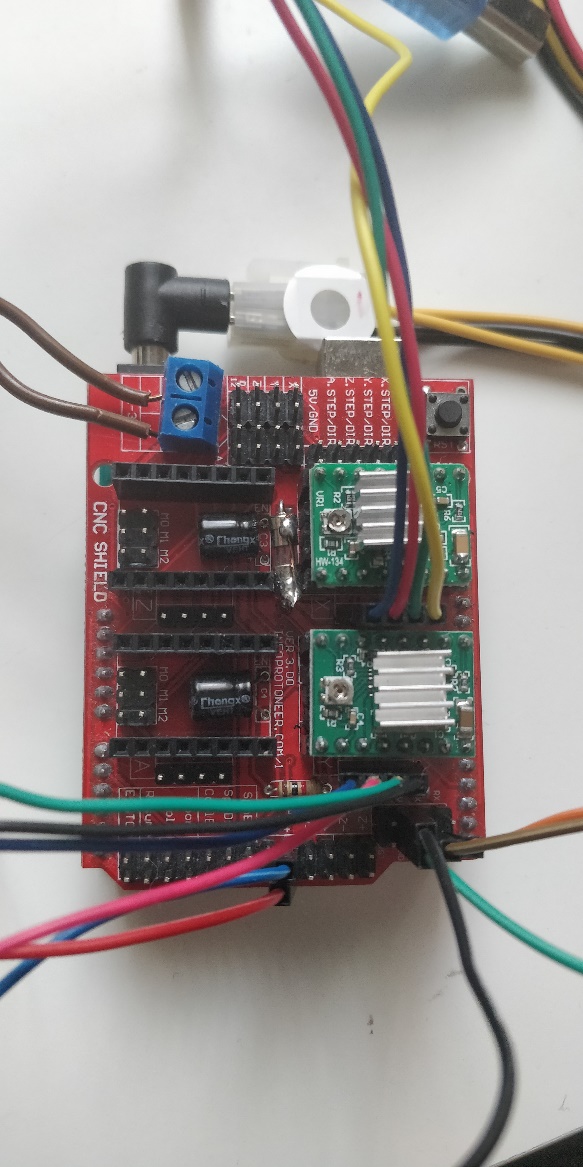
Desde la fuente ATX se han sacado dos cables, uno de 12V (amarillo) y otro de tierra (negro) y se han conectado al módulo CNC Shield como se muestra en la *Ilustración 50.*

La placa Arduino *master* es alimentada a través de la Arduino *slave*, al conectar los pines de 5V y GND, como se explica en el apartado 3.4.3.2.

**CONEXIÓN GND Y 5V CON LA PLACA ARDUINO *MASTER***

**ALIMENTACIÓN 12V PUERTO JACK DE LA PLACA ARDUINO**

Ilustración 51. Alimentación del sistema



**ALIMENTACIÓN 12V MÓDULO CNC SHIELD**

## Especificaciones del diseño

En este apartado se describen las especificaciones del sistema de bobinado automático.

Tabla 7. Especificaciones del sistema de bobinado

|  |  |
| --- | --- |
| Rango de longitudes de la bobina | 10 – 60 mm |
| Rango de diámetros de la bobina | 15 – 40 mm |
| Velocidad de bobinado |  |
| Diámetro máximo del hilo | 0,8 mm |
| Número máximo de capas |  |

El rango de diámetros de la bobina depende del carrete utilizado, la longitud máxima de la dimensión más larga del carrete medida desde el centro de la bobina no puede superar los 45 mm porque durante el giro podría interferir con la correa.

Al cambiar el modelo de carrete sería necesario rediseñar la pieza del soporte el carrete (*Ilustración 24*).

## Estudio económico

Se ha evaluado el coste del desarrollo de este proyecto, para ello se han tenido en cuenta los siguientes hechos:

* El software utilizado es open-source y no ha generado un gasto adicional
* Se ha utilizado equipamiento del laboratorio que no ha sido necesario adquirir, como la impresora 3D

Por lo tanto, en el estudio económico el gasto generado es la suma del coste de las horas de investigación dedicadas y del precio de los componentes adquiridos.

* Para la realización de este proyecto se estima que se han dedicado 120 horas de investigación, a un coste de unos 20€/h.
* El listado completo de materiales adquiridos se encuentra en el apéndice 6.2.

Tabla 8. Coste total del proyecto

|  |  |
| --- | --- |
|  | COSTE |
| Horas de investigación | 2400 € |
| Componentes | 196,26 € |
| TOTAL | **2596,26‬ €** |

Cabe destacar que si se quisiese replicar este proyecto sólo sería necesario adquirir los componentes y el gasto se reduciría a 196,26 €.

Este coste es **inmensamente inferior** al de una bobinadora comercial con especificaciones similares, que era de **20355€**. El detalle del presupuesto se encuentra en el apéndice 6.1.

Adicionalmente, en el repositorio de GitHub [7], en coil-winder/documentation/components hay un archivo con los enlaces de las páginas web de los proveedores donde se han adquirido los componentes.

## Líneas futuras

En este apartado se resumen las mejoras que se podrían hacer en el sistema de bobinado.

* Para devanar **bobinas de diámetros superiores** se podría modificar la altura de los módulos alzadores para evitar la interferencia de la bobina con la correa.
* En el caso de devanar **bobinas con longitudes superiores** sería necesario montar una estructura de mayores dimensiones. Las barras paralelas a los ejes se podrían sustituir por unas de una longitud superior.
* Se podría **optimizar el código** para incluir el parámetro de diámetro del hilo entre los que introduce el usuario antes de iniciar el bobinado. La inclusión de este parámetro mejoraría la precisión del bobinado para diámetros de hilo superiores a 0,7 mm.
* Incluir más posiciones en la pieza tensionadora del hilo para **modificar el ángulo de provisión del hilo** en el caso de que este parámetro estuviese afectando al devanado.
* Para **aumentar la velocidad de bobinado** se podría reemplazar el motor Winder por un modelo que proporcionase un giro más rápido. También habría que comprobar si los drivers (Pololu A4988) serían capaces de proporcionar la corriente necesaria para ese nuevo modelo de motor.
* Si se quisiesen **añadir nuevos periféricos** se podría reemplazar una de las Arduino UNO por una placa Arduino MEGA, para disponer de más pines.

# CONCLUSIONES

# BIBLIOGRAFÍA

[1]. **FRAILE MORA, JESÚS**. *Máquinas Eléctricas. 5ª Edición.* (Madrid). McGraw-Hill/Interamericana España, 2003. ISBN 84-481-3913-5

[2]. **PURCELL, EDUARD M. y MORIN, DAVID J***. Electricity and Magnetism.* Third Edition. (Massachusetts). Cambridge University Press, 2013. ISBN 987-1-107-01402

[3]. **POPOVIC, ZOYA y POPOVIC, BRANCO D.** Introductory Electromagnetics. Prentince Hall, 2007.

[4]. **Bricogeek** [En línea] [Consulta: 15 febrero 2018]. Disponible en: <https://tienda.bricogeek.com/shields-arduino/837-arduino-cnc-shield-v3.html>

[5]. **Arduino.** [En línea] [Consulta: 20 febrero 2018]. Disponible en: <http://arduino.cl/que-es-arduino/>

[6]. **Whitelegg.** [En línea] [consulta: 24 septiembre 2018]. Disponible en: <http://134.213.30.184/products/coil-winding/coil-winding-machines/bm-a-automatic-layer-winding-machine-for-small-coils>

[7]. **MARTÍN BRAVO, Marta.** coil-winder. *GitHub.* [En línea] Disponible en: <https://github.com/mmartinbr>

[8]. **MACHADO SÁNCHEZ, Felipe.** fcad-comps. *GitHub.* [En línea] Disponible en: <https://github.com/felipe-m>

[9]. **REPETIER.** Repetier Host Documentation. [En línea] [Consulta: 15 de febrero de 2018] Disponible en: <https://www.repetier.com>

[10]. **CNC Shield**. [En línea] [Consulta: 14 de febrero de 2018] Disponible en: <https://blog.protoneer.co.nz/arduino-cnc-shield/>

[11]. **Arduino UNO**. [En línea] [Consulta: 20 de marzo de 2018] Disponible en: <https://www.arduino.cc/en/Guide/ArduinoUno>

[12]. **RepRap**. Repetier Host [En línea] [Consulta: 15 de abril de 2018] Disponible en: <https://reprap.org/wiki/Repetier-Host>

[13]. **RepRap**. *NEMA 17 Stepper motor* [En línea] [Consulta: 14 de febrero de 2018] Disponible en: <https://reprap.org/wiki/NEMA_17_Stepper_motor>

[14]. **Pololu**. *A4988 Stepper Motor Driver Carrier* [En línea] [Consulta: 18 de febrero de 2018] Disponible en: <https://www.pololu.com/product/1182>

[15]. **Concurso CNICE mec**. *Tornillos sin fin* [En línea] [Consulta: 25 de febrero de 2018] Disponible en: <http://concurso.cnice.mec.es/cnice2006/material107/operadores/ope_tornillo.htm>

[16]. **Luis Llamas**. *El bus I2C en Arduino* [En línea] [Consulta: 02 de enero de 2020] Disponible en: <https://www.luisllamas.es/arduino-i2c/>

[17]. **OpenActuator**. OpenCoilWinder. *GitHub.* [En línea] [Consulta: 30 de octubre de 2017] Disponible en: <https://github.com/OpenActuator/OpenCoilWinder>

[18]. **EpicenterBryan**. DIY Coil Winder - with Arduino Mega and Marlin 3D printer firmware. *Youtube.* [En línea] [Consulta: 30 de octubre de 2017] Disponible en: <https://www.youtube.com/watch?v=H4sOnQJavTI>

[19]. **ETD59\_31\_22**. *Ferroxcube.* [En línea] [Consulta: 25 Marzo 2018] Disponible en:<https://www.ferroxcube.com/upload/media/product/file/Pr_ds/ETD59_31_22.pdf>

# APÉNDICES

## Presupuesto de una bobinadora profesional





## Lista de materiales adquiridos

Tabla 9. Lista de materiales

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Componente | Proveedor | Cantidad | Precio unitario | Precio total |
| Correa dentada GT2 - 6mm - 1m | HTA3D | 1 | 1,37 € | 1,37 € |
| Polea GT2 - 20 dientes | HTA3D | 2 | 1,10 € | 2,20 € |
| Motor paso a paso Nema 17 | HTA3D | 2 | 12,85 € | 25,70 € |
| Ejes - diámetro 8 | HTA3D | 2 | 2,65 € | 5,30 € |
| Rodamiento lineal Lm8uu | HTA3D | 4 | 0,58 € | 2,32 € |
| Placa Arduino UNO | FARNELL | 2 | 19,49 € | 38,98 € |
| Arduino CNC Shield V3.51 | 3DSMART | 1 | 8,29 € | 8,29 € |
| POLOLU A4988 | ELECTRONICA EMBAJADORES | 2 | 7,11 € | 14,22 € |
| Carcasa Arduino UNO | AMAZON | 2 | 4,30 € | 8,60 € |
| LCD Keypad Shield | AMAZON | 1 | 9,99 € | 9,99 € |
| Fuente de alimentación 500W ATX | AMAZON | 1 | 10,70 € | 10,70 € |
| Interruptor final de carrera | CETRONIC | 1 | 5,37 € | 10,74 € |
| 4 barras de 200mm | MAKERBEAM | 2 | 9,25 € | 18,50 € |
| 4 barras de 100mm | MAKERBEAM | 1 | 5,00 € | 5,00 € |
| 100 tornillos, M3, 6mm | MAKERBEAM | 1 | 6,50 € | 6,50 € |
| 250 tuercas, M3 | MAKERBEAM | 1 | 4,50 € | 4,50 € |
| 12 escuadras | MAKERBEAM | 2 | 6,95 € | 13,90 € |
| Filamento PLA | HTA3D | 0,7 | 13,50 € | 9,45 € |
| Total |  |  |  | **196,26 €** |

## Carrete de la bobina objetivo

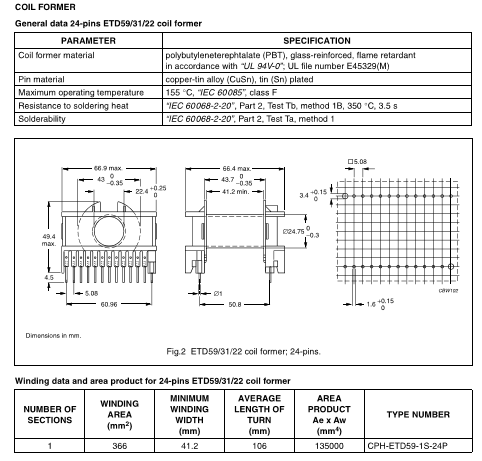


Ilustración 52. Características de la bobina

## Especificaciones del motor Nema 17

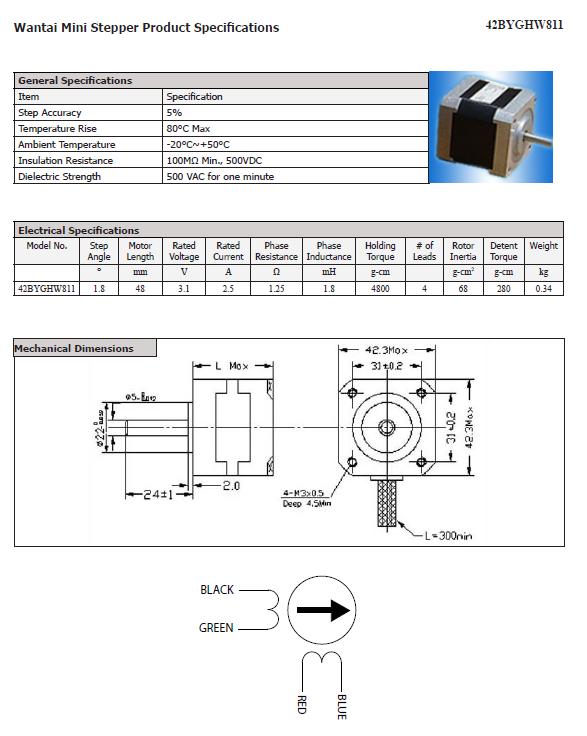


Ilustración 53. Especificaciones motor Nema 17