



Titulació:

**GRADO EN INGENIERÍA ELÉCTRICA**

Alumno (nombre y apellidos):

**ELIAS GIL DE AVALLE MUÑOZ**

Enunciado TFG / TFM:

**DISEÑO, CONSTRUCCIÓN Y ENSAYO DE UN EQUIPO  
DE REGULACIÓN DE VELOCIDAD PARA  
APLICACIONES ESCÉNICAS**

Director/a del TFG / TFM:

**RICARD HORTA BERNÚS**

Convocatoria de entrega del TFG / TFM:

**30/06/2020**



## Índice de contenido

	Página
1 INTRODUCCIÓN.....	10
1.1 Resumen .....	10
1.2 Abstract .....	10
1.3 Declaración de honor .....	11
1.4 Objeto.....	12
1.5 Alcance .....	12
1.6 Especificaciones básicas.....	13
1.7 Justificación.....	13
2 ESTADO DEL ARTE: MAQUINARIA ESCÉNICA O TRAMOYA .....	14
2.1 Empresas especializadas.....	16
2.1.1 PACO SPIRALIFT .....	16
2.1.2 Retom .....	18
2.1.3 Chemtrol.....	19
2.2 Escenarios equipados con maquinaria escénica .....	20
2.2.1 Teatro Real.....	20
2.2.2 Palacio de las artes .....	21
3 EQUIPOS DEL SISTEMA.....	22
3.1 Variador de frecuencia ABB ACS-580-01-R1 .....	22
3.1.1 El variador de frecuencia/velocidad .....	22
3.1.2 Descripción.....	23
3.1.3 Especificaciones iec.....	23
3.1.4 Principio de funcionamiento .....	24
3.1.5 Diagrama de conexionado de potencia .....	25
3.1.6 Macros de control y diagrama de E/S.....	26
3.1.7 Panel de control.....	29
3.1.8 Función safe torque off (sto) .....	31
3.2 Motor trifásico ABB M3AA 90LB 4 .....	31
3.2.1 El motor trifásico asíncrono .....	31
3.2.2 Principio de funcionamiento .....	32
3.2.3 Especificaciones .....	33
3.3 Motorreductor SUMITOMO CYCLO 6000.....	34
3.3.1 Descripción.....	34
3.3.2 Especificaciones .....	35

3.4	Microcontrolador ARDUINO MEGA 2560.....	36
3.4.1	Descripción.....	36
3.4.2	Especificaciones.....	37
3.4.3	Arduino software IDE.....	39
3.5	Sensores y dispositivos de control.....	40
3.5.1	Finales de carrera.....	40
3.5.2	Interruptor.....	41
3.5.3	Encoder rotatorio con pulsador.....	41
3.5.4	Sensor ultrasónico.....	44
3.5.5	Relés.....	45
3.5.6	Pantalla LCD.....	46
3.5.7	Resistencias PULL-UP & PULL-DOWN.....	47
3.5.8	Módulo de alimentación.....	49
4	SOFTWARE ARDUINO.....	50
4.1	Modo manual.....	57
4.2	Modo automático.....	59
4.2.1	Modo subir/bajar telón.....	61
4.2.2	Modo subir/bajar plataforma.....	63
4.2.3	Giro en sentido horario/antihorario de base giratoria.....	66
4.2.4	Contador de 5 vueltas en base giratoria.....	68
4.3	Cambio de valores para otros usos.....	70
4.3.1	Altura de detención de plataforma.....	70
4.3.2	Tiempo de espera final de carrera base giratoria.....	71
5	IMPRESIÓN 3D.....	72
5.1	Plataforma giratoria.....	73
5.2	Plataforma elevadora.....	74
5.3	Telón de escenario.....	75
6	MONTAJE MECÁNICO.....	76
7	MONTAJE ELÉCTRICO.....	88
8	DIAGRAMA DE GANTT.....	100
9	IMPACTO MEDIOAMBIENTAL.....	101
10	SEGURIDAD DE USO.....	101
11	CONCLUSIONES.....	102
12	BIBLIOGRAFÍA.....	104

## Índice de ilustraciones

Ilustración 1 Tramoya de Teatro Canal (Fuente: Alfredo Urdaci).....	15
Ilustración 2 Periakitas griegos. (Fuente: Wikipedia) .....	15
Ilustración 3 Sistema Spiralift (Fuente: PACO GROUP) .....	16
Ilustración 4 Sistema Spiralift en Teatro Real (Fuente: Teatro Real) .....	17
Ilustración 5 Formación del tubo Spiralift a través de sus diferentes partes .....	17
Ilustración 6 Sistema de maquinaria escénica superior (Fuente: RETOM) .....	19
Ilustración 7 Motor de enrollamiento CHT MST (Fuente: Chemtrolstage).....	19
Ilustración 8 Proyecto Auditorio y Centro de congresos de Burgos (Fuente: Youtube- PACO GROUP).....	20
Ilustración 9 Teatro Real en Madrid, Sala principal (Fuente: Teatro Real) .....	20
Ilustración 10 Palacio de las artes de Valencia (Fuente: el Mundo) .....	21
Ilustración 11 Escenario principal de Palacio de las artes de Valencia (Fuente: Dta. Ocio y turismo) .....	22
Ilustración 12 Variador ABB ACS-580-01-R1.....	23
Ilustración 13 Esquema eléctrico del variador ABB ACS-580-A1-R1 (Fuente: Manual hardware ABB) .....	24
Ilustración 14 Diagrama de conexionado de potencia (Fuente: Manual hardware ABB) .....	25
Ilustración 15 Instalación cableado (Fuente: Manual hardware ABB).....	26
Ilustración 16 A la izquierda, E/S digitales de Macro ABB estándar, a la derecha, E/S digitales de macro ABB Manual/Auto (Fuente: Manual Hardware ABB).....	27
Ilustración 17 Diagrama de conexiones digitales y analógicas del variador con Macro ABB estándar (vectorial) (Fuente: Manual Hardware ABB) .....	28
Ilustración 18 Panel de control de ABB (Fuente: Manual de Hardware ABB).....	29
Ilustración 19 Tipos de control en variador ABB ACS-580-A1-R1 (Fuente: Manual hardware ABB) .....	30
Ilustración 20 Botón de parada de emergencia .....	31
Ilustración 21 Motor trifásico abierto con rotor tipo jaula de ardilla (Fuente: CLR Blog) .....	32
Ilustración 22 Corriente de inducción y fórmula de Ley Faraday-Lenz (Fuente: Hyperphysics) .....	32
Ilustración 23 Teorema de Ferraris .....	33
Ilustración 24 Motor ABB M3AA 90LB4.....	33
Ilustración 25 Placa de características técnicas del motor .....	34
Ilustración 26 Motorreductor Cyclo 6000 desglosado (Fuente: Manual Sumitomo Drive Technologies).....	35
Ilustración 27 Reducto Cyclo 6000 Mod. CHHX-6135DBE-165/G90/A200 .....	36
Ilustración 28 Microcontrolador Arduino MEGA 2560 (Fuente: www.arduino.com) .....	37
Ilustración 29 Arduino MEGA 2560 dividido en secciones, cada una con una función diferente.....	38
Ilustración 30 Conexion USB y jack para alimentación .....	39
Ilustración 31 Programa Arduino IDE .....	40
Ilustración 32 Final de carrera AMZLAB V-155-1C25 .....	40
Ilustración 33 Interruptor tipo KCD1-8 .....	41

Ilustración 34 Composición y descripción de las partes que conforman el encoder.....	42
Ilustración 35 Encoder KY-040 de AZ Delivery.....	42
Ilustración 36 Gráficos de generación de señales de A y B en sentido horario .....	43
Ilustración 37 Gráficos de generación de señales de A y B en sentido antihorario.....	43
Ilustración 38 Sensor ultrasónico HC-SR04 .....	44
Ilustración 39 Módulo de Relé TONGLING JQC-3FF-S-Z .....	45
Ilustración 40 Circuito interno del relé (Fuente: Tecnología).....	46
Ilustración 41 LCD de 16x2 para Arduino (Fuente: desucbreArduino).....	47
Ilustración 42 Esquema de resistencia Pull-Up (Fuente: zonamaker) .....	48
Ilustración 43 Esquema de resistencia Pull-Down (Fuente:zonamaker).....	49
Ilustración 44 Módulo de fuente de alimentación ELEGOO Power MB v2.....	50
Ilustración 45 Variables utilizadas en el desarrollo del proyecto.....	51
Ilustración 46 Inicialización de entradas y salidas digitales en void setup().....	53
Ilustración 47 Función que determina la posición en la que se encuentran los encoders .....	54
Ilustración 48 Determinación de la altura a la que se encuentra la plataforma elevadora .....	54
Ilustración 49 Mensaje de error cuando Manual y Automático están en ON.....	55
Ilustración 50 Interruptor Manual y Automático desactivados y programación de escritura de texto.....	56
Ilustración 51 Mensaje de Variador de frecuencia cuando manual y automático se encuentran en OFF .....	56
Ilustración 52 Mensaje que muestra Arduino para indicar que se trabaja en Modo Manual.....	57
Ilustración 53 Mensaje por pantalla de submenú Rampa1/Rampa 2 en Modo Manual .....	58
Ilustración 54 Código de activación de Avance o Retroceso del motor .....	58
Ilustración 55 Mensaje por pantalla de submenú Avance/Retroceso en Modo Manual .....	59
Ilustración 56 Mensaje de Arduino indicando Modo Automático .....	60
Ilustración 57 Variables encargadas de guardar el estado de las diferentes funciones .....	60
Ilustración 58 Mensaje por pantalla de LCD indicando modo SUBIR TELON y modo BAJAR TELON.....	61
Ilustración 59 Telón con cierre de tipo Guillotina (Fuente: Alcázar Textiles) .....	61
Ilustración 60 Código de función subirTelon, ejecutora de la acción de subir el telón.....	62
Ilustración 61 Mensaje de Arduino de modo SUBIR PLATAFORMA Y BAJAR PLATAFORMA.....	63
Ilustración 62 Plataforma utilizada en teatro para elevar asientos (Fuente: DEXEV) ...	64
Ilustración 63 Código de función subirPlataforma, ejecutora de la acción de subir plataforma.....	65
Ilustración 64 Mensaje de Arduino indicando BASE EN SENTIDO GIRATORIO Y BASE EN SENTIDO ANTIHORARIO .....	66
Ilustración 65 Base giratoria para escenarios (Fuente: Pascualin Estructures) .....	66
Ilustración 66 Código que determina las acciones que llevará a cabo la función giraHorario .....	67
Ilustración 67 Mensaje de Arduino indicando modo BASE GIRATORIA 5 GIROS.....	68

Il·lustració 68 Còdigo que determina las acciones que llevarà a cabo la funció giraCinco .....	70
Il·lustració 69 Variable DISTANCIA que detendrá la plataforma a 11cm del suelo .....	70
Il·lustració 70 Relació de transformació especificada por el fabricante para sensor ultrasónico.....	71
Il·lustració 71 Variable de tiempo para funció giraCinco.....	71
Il·lustració 72 Impresora 3D con la que se treballaràn las estructuras .....	72
Il·lustració 73 PLA negro, material empleado para la creació de las estructuras.....	73
Il·lustració 74 Plataforma giratoria montada.....	73
Il·lustració 75 Plataforma giratoria abierta donde se observa el rodamiento ABEC-6.	73
Il·lustració 76 Plataforma giratoria mecanizada .....	74
Il·lustració 77 Diferentes posiciones de la plataforma consiguiendo diferentes alturas .....	74
Il·lustració 78 Plataforma elevadora mecanizada.....	75
Il·lustració 79 Construcción de telón ficticio .....	75
Il·lustració 80 Estructura de telón con finales de carrera mecanizado.....	76
Il·lustració 81 Caja donde se alojaràn todos los componentes ABB de 220x170x80mm .....	76
Il·lustració 82 Taladro común y broca para madera de 10 mm.....	77
Il·lustració 83 Banana hembra de 4 mm.....	77
Il·lustració 84 Mediciones con metro para determinar ubicación de las bananas hembra .....	78
Il·lustració 85 Taladrando los agujeros pertinentes para las bananas hembra .....	78
Il·lustració 86 Limando asperezas post taladrado .....	79
Il·lustració 87 18 Bananas hembras instaladas.....	79
Il·lustració 88 Tornillos de 3x25mm y tuerca de métrica 3.....	80
Il·lustració 89 Fijador de tornillos anaeróbico CEYS.....	80
Il·lustració 90 Pareja de relés fijada a la caja por la parte posterior.....	81
Il·lustració 91 Vista posterior de la caja con los 2 primeros relés instalados.....	81
Il·lustració 92 Relés completamente instalados.....	81
Il·lustració 93 Instalación de LCD en parte frontal .....	82
Il·lustració 94 LCD instalada con tornillería de métrica 3 .....	82
Il·lustració 95 Realización del agujero para el interruptor con broca de 18mm de madera .....	83
Il·lustració 96 Lijado general del contorno del agujero para el interruptor .....	83
Il·lustració 97 Interruptores de manual y automático instalados .....	84
Il·lustració 98 Agujeros para colocación de encoders .....	84
Il·lustració 99 Encoders instalados y sujetos con tuerca propia .....	85
Il·lustració 100 Encoders con botón de impresión 3D colocado.....	85
Il·lustració 101 LEDs de diferentes colores instalados.....	86
Il·lustració 102 Potenciometro de 10 k para la retroiluminación de la pantalla .....	86
Il·lustració 103 Placa base Arduino instalada en caja ABB.....	87
Il·lustració 104 Parte trasera de la caja, 3 tornillos para placa Arduino.....	87
Il·lustració 105 Conexión para fuente de alimentación de Arduino.....	88
Il·lustració 106 Cableado utilizado a lo largo de la instalación eléctrica, marca TUOFENG.....	88
Il·lustració 107 Tubos termo retráctiles pre cortados de EVENTRONIC.....	89

Il·lustració 108 Pines de connexió soldados en cables.....	89
Il·lustració 109 Encoder con cables soldados .....	90
Il·lustració 110 Encoder de modo automàtic y manual soldados.....	90
Il·lustració 111 Encoder con termo retràctil instalado .....	90
Il·lustració 112 Pines de pantalla LCD soldados.....	91
Il·lustració 113 Pines de pantalla protegidos con tubo termo retràctil .....	91
Il·lustració 114 Estañado de resistencia de 220 ohm .....	92
Il·lustració 115 Interruptor conectado con sus respectivos conectores FASTON y tubo termo retràctil de aislamiento. ....	92
Il·lustració 116 Parte frontal de la caja finalizada .....	93
Il·lustració 117 Cableado de bananas hembra.....	93
Il·lustració 118 Creación de 4 cables para los negativos de todos los finales de carrera .....	94
Il·lustració 119 Finalización de cableado de bananas hembras .....	94
Il·lustració 120 Instalación y cableado de relés.....	95
Il·lustració 121 Cableado interno de los 5 relés al completo .....	95
Il·lustració 122 Embarrado de polo negativo conectado a GND de Arduino .....	96
Il·lustració 123 Embarrado de polo positivo conectado a 5V de Arduino .....	96
Il·lustració 124 Encintado con cinta aislante del embarrado negativo o GND .....	96
Il·lustració 125 LED's instalados en parte frontal .....	97
Il·lustració 126 Caja de componentes ABB finalizada .....	97
Il·lustració 127 Conexiones de entradas y salidas digitales y fuente externa de 24 V <sub>DC</sub> hacia los relés.....	98
Il·lustració 128 Parte trasera de la caja de componentes con los visos de fijación vistos .....	98
Il·lustració 129 Diagrama conceptual de la instalación .....	99

## Índice de tablas

Tabla 1 Explicación de elementos marcados en Il·lustració 2 .....	24
Tabla 2 Explicación de elementos marcados en Il·lustració 3 .....	25
Tabla 3 Entradas y salidas digitales del variador a utilizar en el proyecto .....	28
Tabla 4 Interpretación del LED del Panel de Control .....	30
Tabla 5 Características técnicas motor ABB M3AA 90LB A.....	34
Tabla 6 Identificación de pines digitales requeridos.....	53
Tabla 7 Funciones de los interruptores una vez accionados.....	55
Tabla 8 Funciones de cada uno de los submenús de Modo Manual.....	57
Tabla 9 Sentencia del modo SUBIR TELON.....	62
Tabla 10 Sentencia del modo BAJAR TELON.....	63
Tabla 11 Sentencia determinada para subirPlataforma .....	64
Tabla 12 Sentencia de la función bajarPlataforma .....	65
Tabla 13 Sentencia determinada para giraHorario .....	67
Tabla 14 Sentencia de la función giraAntihorario .....	68
Tabla 15 Sentencia de la función giraCinco.....	69



# 1 INTRODUCCIÓ

## 1.1 Resumen

El presente proyecto consiste en crear un dispositivo "Plug-and-Play" con un variador de frecuencia, una placa Arduino que ejercerá de autómatas y un motor trifásico con reductor para su consiguiente aplicación en las artes escénicas. Este dispositivo ha sido una petición de proyecto por parte de la ESTAE, para que los alumnos puedan familiarizarse con los motores trifásicos, los variadores y su automatización en los procesos artísticos. Este artilugio servirá para poder practicar y conocer el funcionamiento de dicho dispositivo.

En este proyecto se tratan dos posibles formas de uso del variador:

- *En modo Manual:* en el que se encuentran las características principales como paro/marcha o avance/retroceso.
- *En modo Automático:* en donde se activará subir el telón del escenario y bajarlo, subir y bajar una plataforma elevadora de un escenario y el giro en sentido horario o antihorario de una plataforma giratoria de escenario.

Pueden derivarse una serie de problemas que plantean los escenarios en la aplicación de esta maquinaria. El principal de estos es la cantidad de cableado disperso y dispositivos de control que se encuentran en el proceso de creación del espectáculo, de ello se deriva la intención de montaje de este dispositivo.

Es decir, poder utilizarlo en cualquier escenario dado su fácil transporte y su bajo peso. Por otra parte, se persigue hacer asequible algo que por su complejidad suele quedar fuera del alcance de la mayoría de estudiantes. Ampliando el uso de dichos equipos a aquellas personas con poco o nulo conocimiento en la materia.

**Palabras clave:** *variador de frecuencia, Arduino, motor trifásico, maquinaria escénica*

## 1.2 Abstract

The present project consists on creating a "Plug-and-Play" device with a frequency converter, an Arduino board that will act as an PLC and a three-phase motor with a reducer for performing arts applications. This device has been a project request by ESTAE, so that students can become familiar with three-phase motors, drives and their automation of artistic processes. This contraption is used to practice and learn how it works.

In this project we deal with two possible ways of using the converter:

- In Manual mode, in which we find the main characteristics such as stop/start or forward/reverse.
- In automatic mode, where we can raise the stage curtain and lower it, raise and lower a stage lifting platform and the clockwise or counterclockwise rotation of a stage rotating platform.

The main problem in a stage is the amount of dispersed wiring and control devices that we can find to create the show, that is why the intention of mounting this device is to be able to use it in any stage and easy to transport given its low structural weight and complexity. On the other hand, the easy understanding of complex electronic equipment for students is a point that has been taken into account.

*Key words: frequency converter, Arduino, three-phase motor, stage machinery*

### 1.3 Declaración de honor

I declare that,

the work in this Degree Thesis is completely my own work, no part of this Degree Thesis is taken from other people's work without giving them credit, all references have been clearly cited.

I'm authorised to make use of the company's related information I'm providing in this document.

I understand that an infringement of this declaration leaves me subject to the foreseen disciplinary actions by *The Universitat Politècnica de Catalunya - BarcelonaTECH*.

Elías Gil de Avalue Muñoz

Student Name

Signature

04/05/2020

Date

Title of the Thesis : **DISEÑO, CONSTRUCCIÓN Y ENSAYO DE UN EQUIPO DE REGULACIÓN DE VELOCIDAD PARA APLICACIONES ESCÉNICAS**

## 1.4 Objeto

El objetivo de este proyecto es crear un sistema que pueda controlar un motor trifásico con reductor -o sin él- a través de un variador de frecuencia junto a un Arduino, el cual actúa como PLC, para poder crear diferentes situaciones escénicas a partir de un uso sencillo (fácilmente realizable para alumnos de la escuela ESTAE<sup>1</sup>).

El sistema deberá ser de fácil transporte, así como manejable y sencillo de conectar en cualquier lugar. Este contará con una serie de ventajas a nivel conexión para que cualquier usuario pueda conectarlo para hacer uso de él de forma segura y efectiva.

El conjunto de componentes está formado por una LCD, para poder interactuar entre el usuario y el dispositivo, 2 interruptores de selección de modo y 2 encoders para poder desplazarse a través de los menús preconfigurados en Arduino. A su vez, la caja contará con conexiones trifásicas para poder conectar tanto la red al variador de frecuencia, como el motor al mismo.

Las aplicaciones de motores eléctricos más comunes en artes escénicas son:

- Subir y bajar telón del escenario.
- Elevar y descender la plataforma de tijera<sup>2</sup>
- Girar en sentido horario o antihorario una plataforma giratoria

El sistema deberá cumplir con todos los aspectos necesarios para un funcionamiento óptimo.

## 1.5 Alcance

De este proyecto se obtiene:

- Manual de uso del sistema Arduino-Variador de frecuencia-Motor.
- Manual de instalación y montaje del sistema comentado anteriormente
- Guía y descripción de las funciones del sistema.
- Modelaje de la caja de transporte para todos los equipos que lo componen en 2D y en 3D.
- Planos del conjunto.

---

<sup>1</sup> Escuela superior de técnicas de las Artes del Espectáculo

<sup>2</sup> Son un tipo de plataforma elevadora de personal utilizada frecuentemente en aplicaciones de construcción y mantenimiento, aunque también son utilizadas en escenarios.

- Presupuesto económico
- Creación del código fuente para el correcto funcionamiento de Arduino.
- Prueba y puesta en marcha del equipo
- Recreación a pequeña escala de los diferentes escenarios propuestos para el uso de este conjunto en 3D para posteriormente imprimirlos con impresora 3D.
- Montaje del equipo propuesto y guía paso a paso de montaje desde 0.
- Montaje de sensores en las recreaciones a pequeña escala de los diferentes escenarios.
- Explicación y estudio de los componentes utilizados.

## 1.6 Especificaciones básicas

El sistema deberá cumplir, por lo menos, con las siguientes especificaciones básicas:

- Utilización intuitiva para los alumnos en prácticas.
- Facilidad en el transporte del conjunto.
- Sistema de comandos intuitivo.
- Manejabilidad y adaptación óptima a cada espacio de trabajo.
- Montaje en destino sencillo.
- Producción del sistema efectivo para futuras recreaciones.
- Coste reducido de creación respecto a sistema tradicional con Autómata.
- Conexiones lógicas asequibles entre caja (o conjunto) y exterior.
- Mantenimiento mínimo y fácil

## 1.7 Justificación

Hoy en día los lugares donde se crean artes escénicas precisan de automatismos<sup>3</sup> para poder crear el control de los diferentes elementos que componen el escenario. En este caso, para artes escénicas, son necesarios motores capaces de levantar grandes plataformas o subir telones pesados.

Todo ello consigue controlar su velocidad y su aceleración a través de un variador de frecuencia. ¿Por qué se necesitan? Muy sencillo. Por ejemplo, para que una plataforma se eleve lentamente cuando tenemos la intención de subir un elemento frágil, o quizás se requiere una retracción rápida (para poder crear una escena de shock).

Mas allá de esto, supongamos que necesitamos monitorizar todos estos elementos desde un centro de control, de forma ordenada. ¿Cómo podemos controlar todos estos

---

<sup>3</sup> Equipos tales como finales de carrera, sensores de posición o de láser, etc.

componentes de forma automática? Con sensores ultrasónicos, finales de carrera y autómatas. El sistema Arduino permite hacer un seguimiento de todo lo comentado con anterioridad.

Ello supone un gran reto dado que, un autómata, normalmente, tiene un coste de alrededor de los 2.000€ y una placa Arduino de unos 13€. Es decir, se estaría reduciendo el coste 150 veces. De este modo se consigue abaratar costes de una forma sustancial manteniendo los resultados que se obtendrían con un autómata.

Crear un espacio de control, sencillo e intuitivo para cualquier persona es necesario en el ámbito de las artes escénicas, por ello, se crea un modelo flexible en cuanto a portabilidad y conectividad, reduciendo las tareas y los tiempos necesarios para el montaje de estos escenarios.

## 2 ESTADO DEL ARTE: MAQUINARIA ESCÉNICA O TRAMOYA

El mundo del teatro ha ido renovándose a lo largo de los últimos años. Esto corre en paralelo con la llegada de las TIC<sup>4</sup> ha habido todo un proceso de modernización de instalaciones y de escenografía para poder satisfacer las renovadas exigencias en las artes escénicas. La infraestructura empleada en dichas artes normalmente está compuesta por una serie de estructuras mecánicas con variaciones de movilidad y motores eléctricos que hacen que el movimiento de cualquier objeto sea facilitado.

Las artes escénicas no solamente comprenden el teatro, sino que esta maquinaria escénica también puede ser aplicada en eventos de todo tipo tales como: Danza, zarzuela, musicales, circo, música en vivo, eventos como pasarelas, convenciones, congresos, mítines e incluso exhibiciones como ferias, exposiciones y museos entre otros.

La tramoya es el conjunto de máquinas e instrumentos con los que se trabaja durante una representación escénica para crear ambientes, climas y cambios en el decorado. Esto ha ido evolucionando con el paso de los años ya que, por ejemplo, en la antigua Grecia o la antigua Roma, ya se utilizaban diferentes maquinarias en los teatros por tal de representar los diferentes escenarios. Véase el caso del *Deus ex machina*<sup>5</sup>.

---

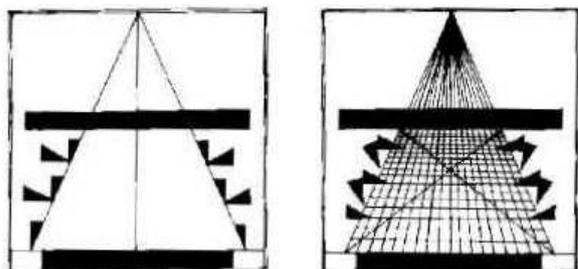
<sup>4</sup> Tecnologías de la información y la comunicación

<sup>5</sup> Deus ex machina. Se origina en el teatro griego y romano, cuando una grúa (*machina*) o cualquier otro medio mecánico introducía desde fuera del escenario a un actor que interpretaba a una deidad (*deus*) para resolver una situación o dar un giro a la trama.



*Il·lustració 1 Tramoya de Teatro Canal (Fuente: Alfredo Urdaci)*

En la antigua Grecia, uno de los recursos más utilizados eran las “periakitas”, que consisten en pequeñas pirámides que giraban sobre un eje e intercambiaban los decorados o también algunos primitivos sistemas de grúa para poder hacer descender o ascender a los dioses. En aquellos tiempos, puesto que no tenían motores, utilizaban los medios que tenían a su alcance, es por eso que también contaban con unos sistemas de plataforma de madera móviles con el fin de transportar los decorados.



*Il·lustració 2 Periakitas griegos. (Fuente: Wikipedia)*

Actualmente, en un mundo mucho más avanzado, existen sistemas mecánicos, eléctricos y electrónicos que combinados proporcionan mayores ventajas de creación. Hay muchas empresas dedicadas en este sector, pero hay algunas que están especializadas en producción de sistemas de elevación de plataformas o de apertura y cierre del telón. Estas empresas son:

- RETOM
- CHEMTROL
- STRONG

Existe una empresa que, pese a no estar especializada en este sector, su tecnología SPIRALIFT es puntera en sistemas de elevación de plataformas. Esta es PACO SPIRALIFT de Canadá.



Ilustración 3 Sistema Spiralift (Fuente: PACO GROUP)

## 2.1 Empresas especializadas

### 2.1.1 PACO SPIRALIFT

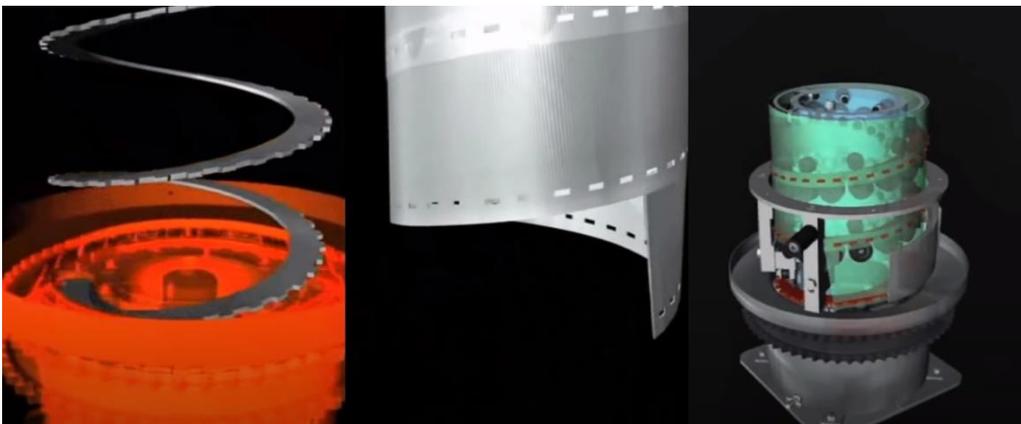
Es una empresa canadiense especializada en el sistema SPIRALIFT (patentado por ellos). Esta se dedica a innovar y crear soluciones pensadas para cada situación concreta para la manipulación de material, así como sistemas de elevación hechos a medida. En un principio la actividad de PACO GROUP se concentraba en sistemas de minería y manejo de "concreto"<sup>6</sup>, pero a través de la adquisición de experiencia en estos sectores, formó 3 submarcas, entre las que esta GALA SYSTEMS, dedicada única y exclusivamente a aplicaciones teatrales/escénicas.

<sup>6</sup> Hormigón, material empleado en la construcción, formado esencialmente por aglomerante.



*Il·lustració 4 Sistema Spiralift en Teatre Real (Fuente: Teatre Real)*

Este sistema de elevación es el más compacto del mundo. Se trata de dos roscas de metal. En una de ellas va el tubo de forma descompuesta y en la otra el sistema que se encarga de fijar el tubo descompuesto.



*Il·lustració 5 Formació del tubu Spiralift a través de sus diferentes partes*

Este sistema permite transformar una sala en cualquier otra diferente y así adaptarla a las múltiples funciones que se requieran. Entre sus características destacan los tiempos de conversión o elevación que normalmente no superan los 10-30 minutos, la amalgama de configuraciones disponibles y la opción de crear gradas a través de la elevación de butacas.

Las ventajas de este sistema respecto al convencional son:

- Altura muy reducida necesaria para su montaje
- Gran capacidad de carga
- Gran rendimiento mecánico del sistema
- Posibilidad de velocidad de elevación alta
- Sistemas de seguridad integrados
- Mantenimiento reducido ya que no requiere de lubricantes

### 2.1.2 Retom

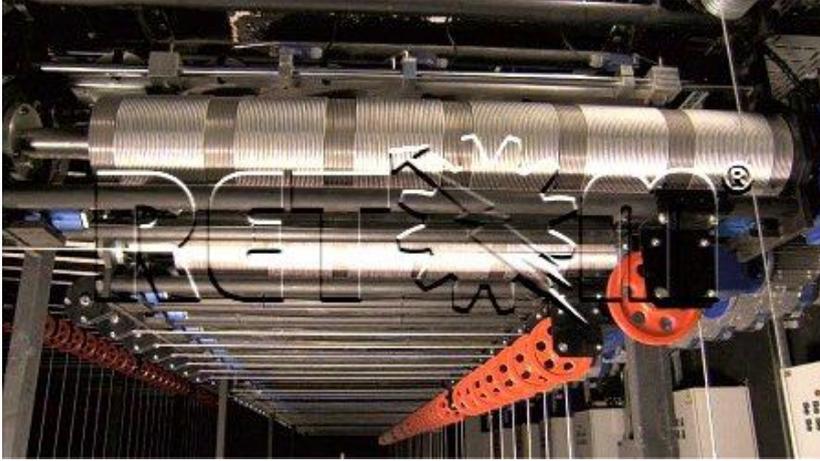
Es una empresa española con 35 años de antigüedad, de Ciudad Real. Disponen de una amplia gama de equipamiento para artes escénicas. Entre ellos podemos destacar:

- Tejidos escénicos y telones cortafuegos.
- Maquinaria escénica superior:
  - Cortes motorizados de velocidad variable.
- Maquinaria escénica inferior:
  - Plataformas Spiralift RSM<sup>7</sup> (instalador autorizado).
- Escenarios modulares.
  - Plataformas desmontables.
  - Escenarios desmontables.
  - Sistemas de tarimas.
  - Stands.
  - Pasarelas desmontables.
- Control de motores con sistema SIEMAQ<sup>8</sup>.
- Gradas móviles.

---

<sup>7</sup> Plataformas de orquesta con sistema de elevación a base de Spiralifts

<sup>8</sup> Sistema de control escénico inalámbrico



*Il·lustració 6 Sistema de maquinaria escènica superior (Fuente: RETOM)*

### 2.1.3 Chemtrol

Empresa española con 25 años de trayectoria. Está dedicada a ofrecer equipamiento técnico para teatros, auditorios, palacios de congresos, parques temáticos, entre otros. Su entrada en el mercado vino marcada por varios acuerdos comerciales con empresas británicas y americanas y por el proyecto de gran importancia, la reforma del "Gran teatro del Liceu".

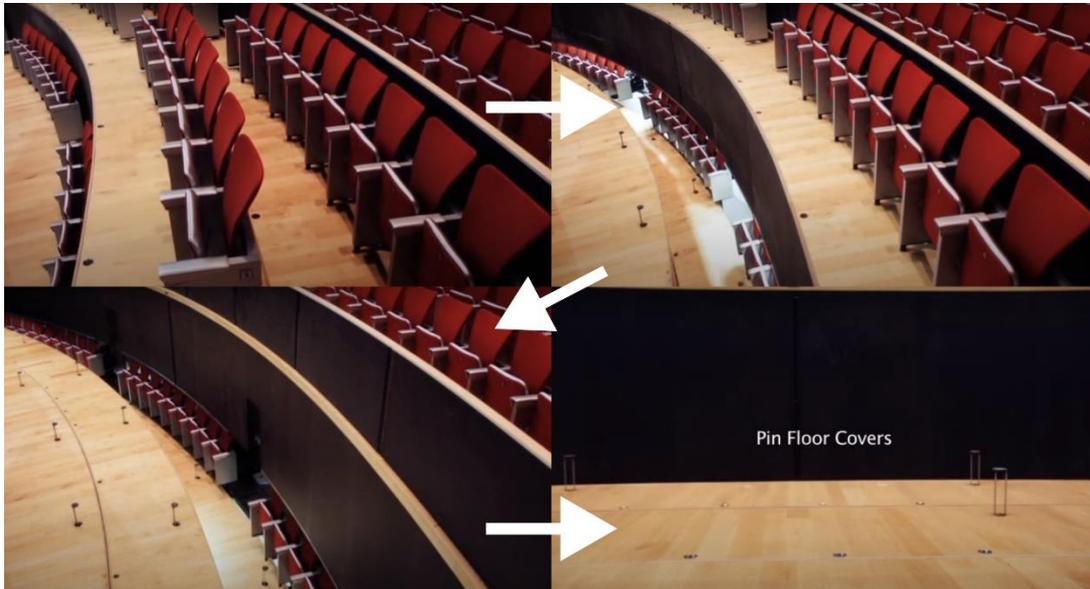
Esta empresa tiene una línea de productos propios pensados especialmente para maquinaria escénica y sistemas de control.



*Il·lustració 7 Motor de enrollamiento CHT MST (Fuente: Chemtrolstage)*

Esta empresa, igual que la anteriormente comentada RETOM, dispone de sistemas para maquinaria escénica inferior y superior, equipamiento textil y control de motores, entre otros.

En el 2012, crearon junto a PACO GROUP un sistema de movimiento automático de butacas en el nuevo Auditorio y Centro de congresos de Burgos, en el que se muestra la gran complejidad y precisión del sistema.

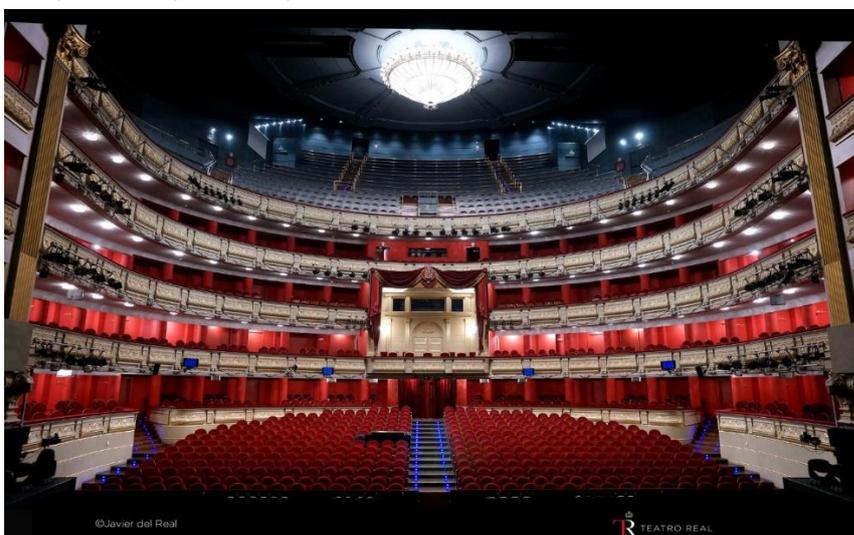


*Ilustración 8 Proyecto Auditorio y Centro de congresos de Burgos (Fuente: Youtube-PACO GROUP)*

## 2.2 Escenarios equipados con maquinaria escénica

### 2.2.1 Teatro Real

Situado en la capital de España, en Madrid, es el teatro más avanzado a nivel de maquinaria escénica de todo el país. Cuenta con un total de 22 plantas con diferentes niveles abatibles y convertibles para poder recrear una situación u otra. Estos pisos van desde la planta -8 hasta la 14. En los pisos inferiores es donde se encuentra toda la maquinaria que incorpora el teatro.



*Ilustración 9 Teatro Real en Madrid, Sala principal (Fuente: Teatro Real)*

Entre sus equipos podemos destacar:

- Cuatro plataformas con movimiento vertical e inclinable
- Cuatro plataformas con movimiento horizontal
- Cinco plataformas con movimiento vertical
- Nueve plataformas secundarias con movimiento vertical
- Dos plataformas para tráiler
- Una plataforma monta decorados
- Tres plataformas con movimiento vertical en el foso de orquesta

En la última actualización realizada por la empresa CHEMTROL, se llevó a cabo la instalación de testers motorizados, conjuntos motorizados de eje continuo, sistema de control de la iluminación lateral y red digital DMX<sup>9</sup> de control de iluminación escénica.

El sistema de motorización de velocidad variable de los techos de la cámara de conciertos, incluyendo el sistema de control computarizado de tiros de cable y cadena, también se llevaron a cabo pasados unos meses.

### 2.2.2 Palacio de las artes

Lanzado en 2005, utiliza tecnología de punta y utiliza imágenes correspondientes a espacios vanguardistas y contemporáneos. Este gran proyecto de Santiago Calatrava se perfeccionó con un buen equipo para la escenografía. Está ubicado en Valencia.



*Ilustración 10 Palacio de las artes de Valencia (Fuente: el Mundo)*

---

<sup>9</sup> Digital Multiplex, protocolo de comunicación de luminiotecnica.

Está equipado con:

- 1 telón cortafuegos
- 111 barras motorizadas
- 2 controladores de motores centralizados
- 100 motores puntuales
- 1 cámara de conciertos modular
- 1 telón de embocadura y bambilón
- 1 cámara negra
- 36 enrolladores
- 25 plataformas de escenario
- 5 plataformas de foso para Orquesta



*Ilustración 11 Escenario principal de Palacio de las artes de Valencia (Fuente: Dta. Ocio y turismo)*

### 3 EQUIPOS DEL SISTEMA

#### 3.1 Variador de frecuencia ABB ACS-580-01-R1

##### 3.1.1 El variador de frecuencia/velocidad

Un variador de velocidad es un dispositivo electrónico que permite alimentar al motor con una tensión y frecuencia variable, modificando las magnitudes fijas de tensión y frecuencia de red, regulando así su velocidad.

Se utiliza cuando las necesidades del funcionamiento del motor precisan de dominio del par y la velocidad, regulación sin golpes mecánicos, movimientos complejos, etc.

Estos dispositivos permiten conseguir considerables ahorros de energía, menor ruido de funcionamiento del motor, mejor control, mayor duración de vida útil del motor, etc.

### 3.1.2 Descripción

Es el actuador principal de este proyecto. El ABB ACS-580-01-R1 es un convertidor de frecuencia para motores trifásicos. El rango de potencia que abarca es desde 0,75 kW hasta 5,5 kW (1 CV hasta 7,37 CV).



Il·lustració 12 Variador ABB ACS-580-01-R1

Este variador dispone de algunas características importantes que se utilizarán como:

- Entradas y salidas digitales
- Función de inversión de giro
- Tiempos de rampas de aceleración y deceleración
- Ajustes de V/Hz de par variable
- Optimizador de energía
- Control PID
- Eliminación de vibraciones mecánicas
- 

### 3.1.3 Especificaciones iec

- Tensión de alimentación: trifásica,  $U_N = 380 \text{ a } 480 \text{ V}$ , +10%/-15%
- Frecuencia de alimentación: 50/60 Hz  $\pm 5\%$
- Rango de corriente de salida nominal: trifásica, 2,6 A – 12,6 A
- Rango de corriente de salida máxima: trifásica, 3,2 A – 14,1 A
- Margen de control de frecuencia: de 0 hasta  $\pm 500$  Hz
- Tipo de control del motor: Escalar y vectorial

- Tipo de control de par: incremento de par < 10ms
- Tipo de control de velocidad:
  - Precisión estática: 20 % del deslizamiento del motor nominal
  - Precisión dinámica: 1% segundos con escalón de par del 100%

### 3.1.4 Principio de funcionamiento

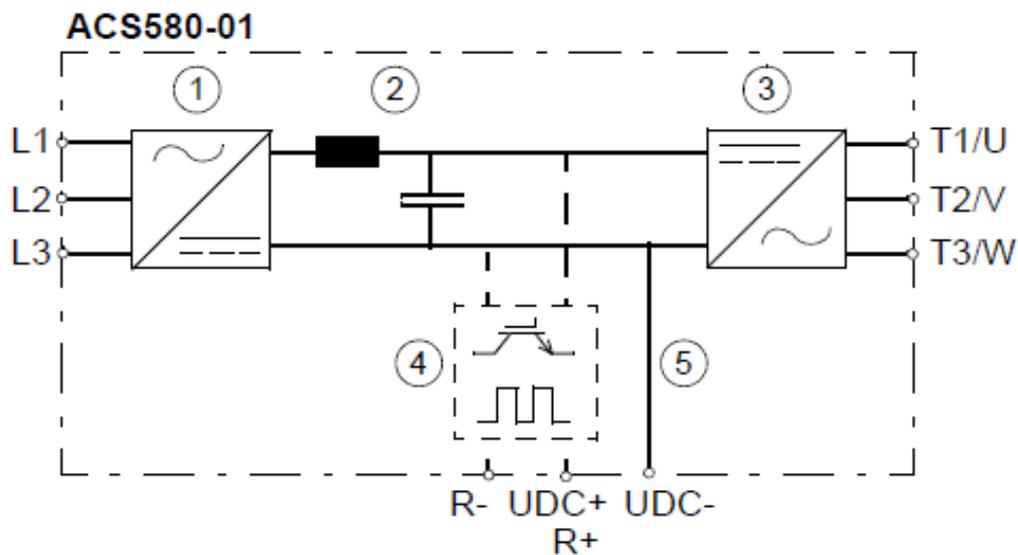
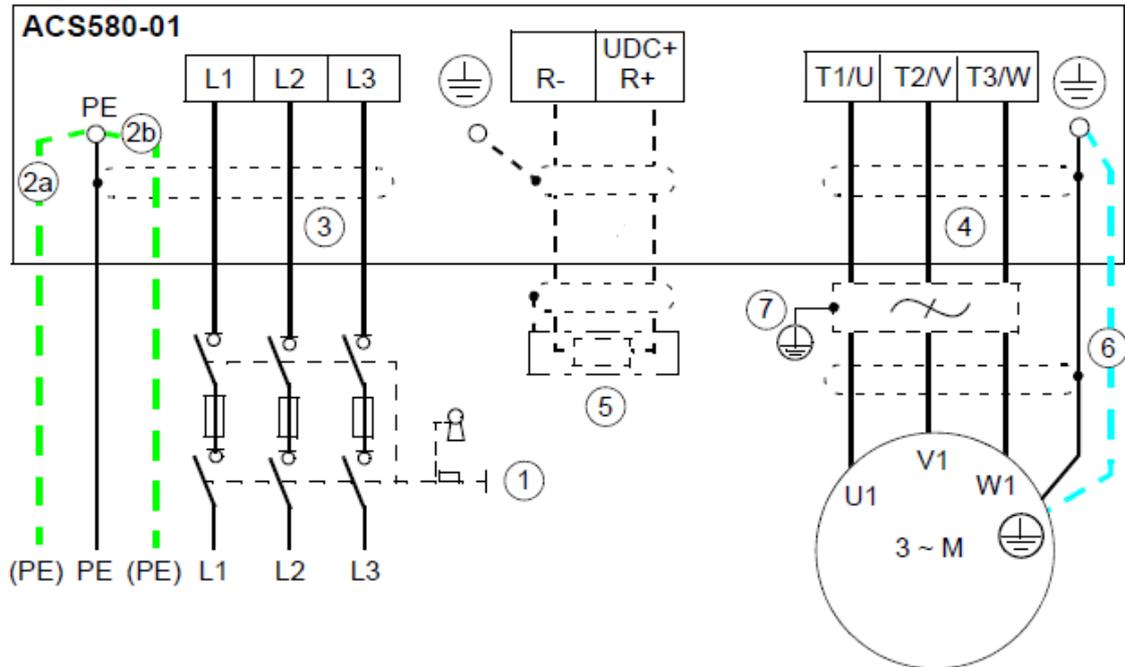


Ilustración 13 Esquema eléctrico del variador ABB ACS-580-A1-R1 (Fuente: Manual hardware ABB)

1	Rectificador. Convierte tensión e intensidad CA en tensión e intensidad CC
2	Bus de CC. Es el circuito eléctrico que tenemos entre rectificador e inversor
3	Inversor. Convierte la corriente y tensión CC en intensidad y tensión CA
4	Chopper de frenado integrado (R-, R+). El exceso de energía que queda en el circuito intermedio de corriente continua, lo reconduce hacia la resistencia de frenado cuando este sea requerido. Este se activa cuando la tensión del bus de CC sobrepasa el límite superior. Si la tensión sobrepasa el límite máximo, se debe principalmente a que el motor se encuentra en un momento de deceleración.
5	Solo utilizado en los modelos R5-R9

Tabla 1 Explicación de elementos marcados en Ilustración 2

### 3.1.5 Diagrama de conexionado de potencia

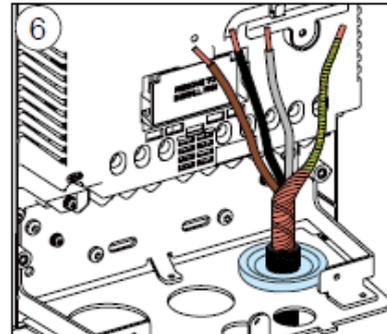


Il·lustración 14 Diagrama de conexionado de potencia (Fuente: Manual hardware ABB)

1	Instalación de dispositivo de desconexión de entrada con acción manual entre la fuente de desconexión de CA y el convertidor de frecuencia. <b>Deberá ser de tipo bloqueable para poder realizar trabajos de mantenimiento.</b>
2	<ul style="list-style-type: none"> <li>• 2a: Cable PE con conexión a tierra separada</li> <li>• 2b: Cable con un conductor de protección a tierra separado</li> </ul>
3	Conexión de fuente de alimentación CA trifásica hacia el variador, conexión a tierra 360° si se utiliza conductor apantallado
4	Conexión trifásica desde el variador de frecuencia hacia el filtro du/dt. Conexión a tierra 360° requerida
5	Resistencia de frenado interna, conexión a tierra 360° si se utiliza conductor apantallado
6	Conexión trifásica que va desde el filtro du/dt hasta el motor trifásico
7	Filtro du/dt recomendado opcional

Tabla 2 Explicación de elementos marcados en Ilustración 3

Para una correcta instal·laci3, todos los cables a utilizar deber3n pasar por el pasacables incluido, de esta forma se asegura su hermeticidad.



Il·lustraci3 15 Instalaci3 cableado  
(Fuente: Manual hardware ABB)

### 3.1.6 Macros de control y diagrama de E/S

Nuestro variador est3 compuesto por 6 entradas digitales. Estas entradas digitales, seg3n la macro que se seleccione, tendr3n una funci3n u otra.

¿Qu3 es una macro? Son conjuntos de valores de par3metros por defecto adecuados para una cierta configuraci3n de control. Simplemente ABB ha creado 10 formas de controlar el variador a trav3s de sus salidas y entradas digitales. Cada macro ha sido creada para un tipo de aplicaci3n diferente:

- Macro ABB est3ndar: esta macro es la que viene instalada de serie en el variador y que se deber3 cambiar si queremos utilizar cualquiera de las otras. Tiene control escalar y ofrece un uso general.
- **Macro ABB est3ndar (vectorial):** esta es la macro que se utilizar3 para este proyecto ya que tiene control de tipo vectorial. Igual que la ABB est3ndar, pero con control vectorial.
- Macro 3 hilos: creada para controlar el variador a trav3s de pulsadores
- Macro Alterna: compuesta por 2 se3ales, donde una de ellas activa el motor en direcci3n avance y la otra en direcci3n retroceso.
- Macro Potenci3metro del motor: dise3ada para ajustar la velocidad con 2 pulsadores.
- Macro Manual/Auto: a utilizar si hay necesidad de cambio entre dos dispositivos de control externo.
- Macro Manual/PID: controla el convertidor con un regulador PID de proceso incorporado.
- Macro PID: cuando la referencia proviene de la entrada anal3gica y el convertidor est3 controlado por el PID, deberemos utilizar esta macro.

- Macro Panel PID: utilización de esta macro si el punto de ajuste se define con el panel de control y el variador está siendo controlado por un PID.
- Macro PFC: Permite controlar múltiples bombas y ventiladores mediante las salidas de relé del convertidor.

10	+24V	Salida de tensión auxiliar +24 V CC, máx. 250 mA	10	+24V	Salida de tensión auxiliar +24 V CC, máx. 250 mA
11	DGND	Común de la salida de tensión auxiliar	11	DGND	Común de la salida de tensión auxiliar
12	DCOM	Común de todas las señales digitales	12	DCOM	Común de todas las señales digitales
13	DI1	Paro (0) / Marcha (1)	13	DI1	Paro (0) / Marcha (1) (Manual)
14	DI2	Avance (0) / Retroceso (1)	14	DI2	Avance (0) / Retroceso (1) (Manual)
15	DI3	Selección de velocidad <sup>(1)</sup>	15	DI3	Control manual (0) / Control automático (1)
16	DI4	Selección de velocidad <sup>(1)</sup>	16	DI4	Permiso de marcha; si es 0, el convertidor para
17	DI5	Conjunto de rampa 1 (0) / Conjunto de rampa 2 (1) <sup>(2)</sup>	17	DI5	Avance (0) / Retroceso (1) (Auto)
18	DI6	No configurado	18	DI6	Paro (0) / Marcha (1) (Auto)

Ilustración 16 A la izquierda, E/S digitales de Macro ABB estándar, a la derecha, E/S digitales de macro ABB Manual/Auto (Fuente: Manual Hardware ABB)

En la anterior imagen, se observa como las diferentes E/S, cambian por completo, ofreciendo diferentes posibilidades con un mismo sistema de entradas digitales.

Por conveniencia y dado que el control vectorial para la aplicación de este proyecto es mas adecuado, se ha elegido la macro ABB estándar vectorial.

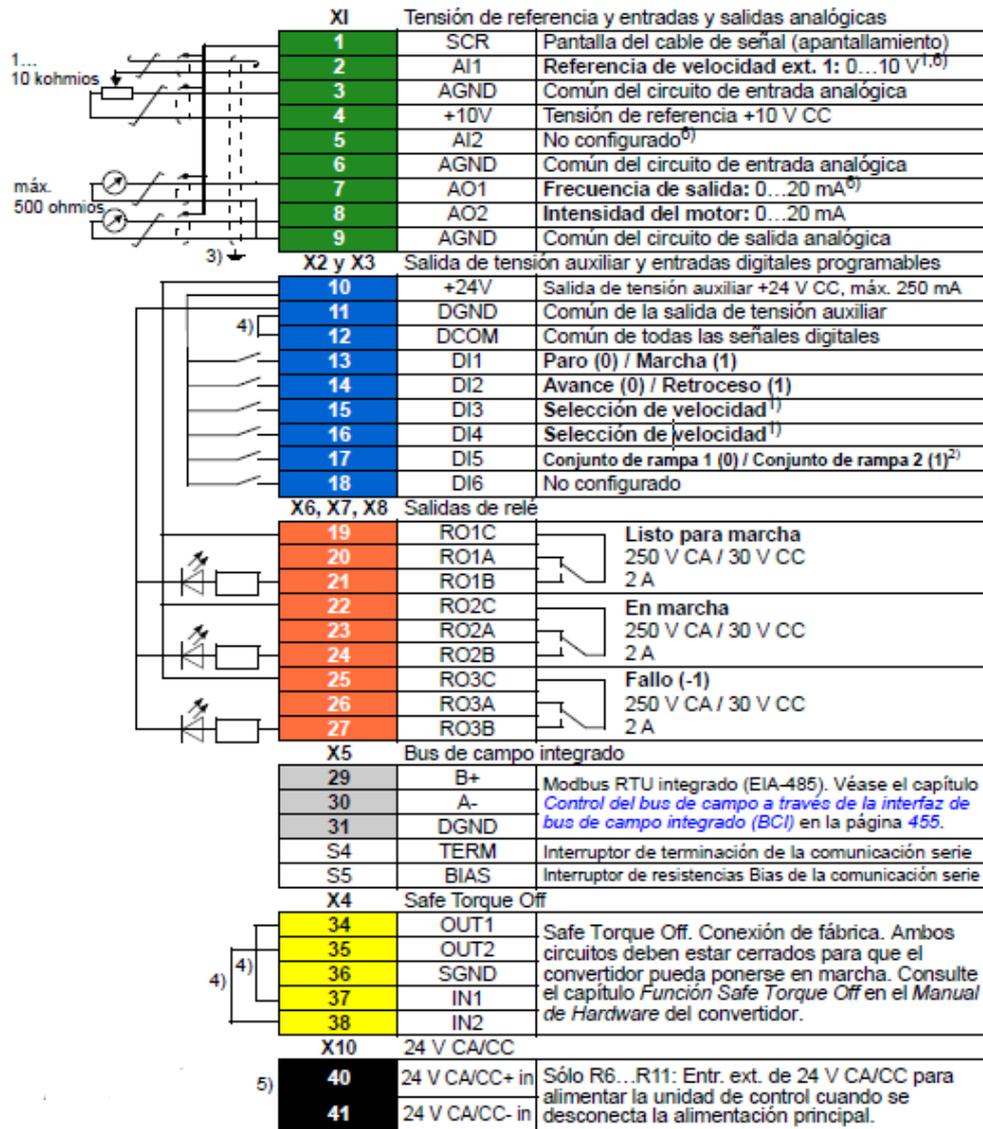


Ilustración 17 Diagrama de conexiones digitales y analógicas del variador con Macro ABB estándar (vectorial) (Fuente: Manual Hardware ABB)

Las señales de entrada y salida que se van a utilizar en este proyecto son las siguientes:

13	Paro (0) / Marcha (1)
14	Avance (0) / Retroceso (1)
15	Selección de frecuencia / Velocidad constante 1 (1)
16	Selección de frecuencia / Velocidad constante 2 (1)
15+16	Selección de frecuencia / Velocidad constante 3. Entradas 15 (1) y 16 (1)
17	Selección de rampa 1 (0) / Selección de rampa 2 (1)
20/21	Salida con indicación de que el variador está LISTO
23/24	Salida con indicación de que el variador está EN MARCHA
26/27	Salida con indicación de que el variador está EN FALLO

Tabla 3 Entradas y salidas digitales del variador a utilizar en el proyecto

### 3.1.7 Panel de control

Para poder configurar todos los parámetros del variador como, por ejemplo, el idioma o las características del motor que se van a utilizar, es necesario poder entrar los datos desde la consola central.

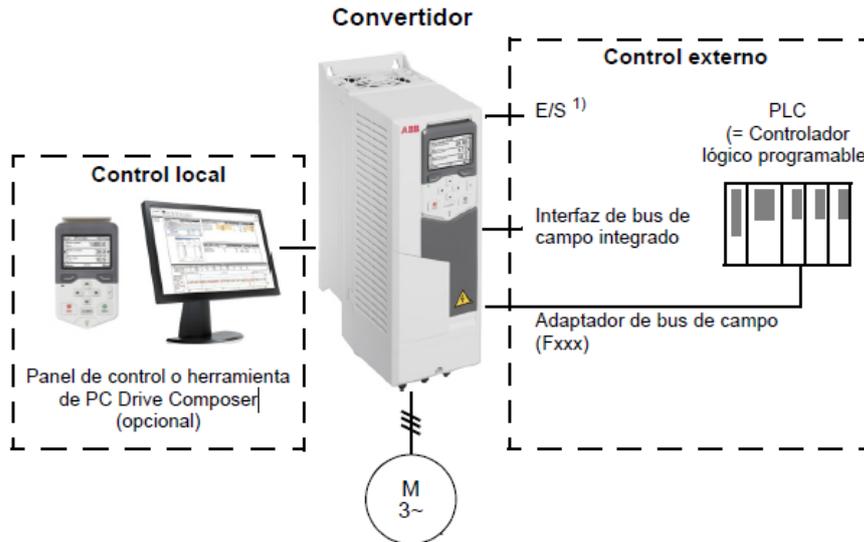
A continuación, se describen las secciones del teclado que definen las funciones de esta:

1. Clip - Presionar para extraer
2. Pantalla – Muestra los menús e información que se le solicite
3. RJ-45 – Este va en la parte trasera del panel y hace de comunicador entre panel de control y variador
4. Botón atrás – Ir hacia atrás en los menús
5. Botón OK – Seleccionar ajustes u opciones
6. Led de estado – Indica el estado del variador en rojo o verde según si tiene fallos o no
7. Flechas – Permiten moverse por los menús
8. Start/Stop – Empezar o finalizar operación en modo de control local
9. Botón Loc/Rem – Selección de control local o externo



*Ilustración 18 Panel de control de ABB  
(Fuente: Manual de Hardware ABB)*

Para este proyecto se precisa de control externo, ya que será el propio Arduino el que le mandará las señales digitales indicando las operaciones que debe llevar a cabo el variador.



Il·lustració 19 Tipus de control en variador ABB ACS-580-A1-R1 (Fuente: Manual hardware ABB)

A continuació, se adjunta una taula en la que se indiquen els diferents estats que indica el led ubicat en el panel de control del variador:

Led del panel de control, ubicat en el marge esquerre				
Led apagat	Led encendit i sense parpadear	Led parpadellant/destellejant		
Panel sense alimentació	Verde	El convertidor funciona amb normalitat. La connexió entre el convertidor i el panel de control ha fallat o se ha perdut, o el panel i el convertidor són incompatibles. Comprova la pantalla del panel de control.	Verde	<u>Parpadellant:</u> Advertència activa en el convertidor. <u>Destellejant:</u> Se transfieren dades entre la ferramenta de PC i el convertidor a través de la connexió USB del panel de control.
	Rojo	Comprova la pantalla per veure de d'on procedeix el fallat. • Fallat actiu en el convertidor. Restaure el fallat. • Fallat actiu en un altre convertidor del bus de panel. Seleccione el convertidor en qüestió i comprova	Rojo	Fallat actiu en el convertidor. Per restaurar el fallat, desconnecte i connecte de nou l'alimentació del convertidor.

Tabla 4 Interpretación del LED del Panel de Control

### 3.1.8 Función safe torque off (sto)

La función Safe Torque Off (STO) se utiliza, por ejemplo, para construir circuitos de seguridad o supervisión que paren el variador de frecuencia en una situación de peligro. Cuando se activa, la función STO inhabilita la tensión de control de los semiconductores de potencia de la etapa de salida del convertidor, lo que hace que el convertidor no pueda generar el par suficiente para mover el motor. Si está el motor en marcha, y se activa STO, el motor se para por sí solo.



*Ilustración 20 Botón de parada de emergencia*

En este caso se implementará una parada de emergencia con un botón con posibilidad de enclavamiento para que, en cualquier caso, se pueda detener el motor enseguida.

Este deberá cumplir con la normativa ISO 13850, que sirve para prevenir situaciones de riesgo

## 3.2 Motor trifásico ABB M3AA 90LB 4

### 3.2.1 El motor trifásico asíncrono

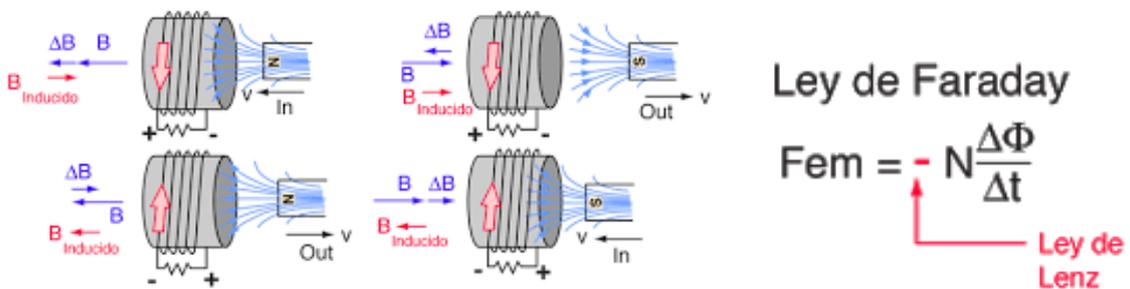
Un motor trifásico asíncrono es una máquina eléctrica alimentada en corriente alterna que puede ser utilizada tanto como generador como de motor. Está compuesto por un estátor y un rotor, entre otros, donde el primero alimenta al segundo para poder crear el giro.

El rotor puede ser de dos tipos, o bien de jaula de ardilla o de rotor bobinado. El estátor en cambio está formado por las bobinas inductoras.



Il·lustració 21 Motor trifàsic obert amb rotor tipus jaula de ardilla (Fuente: CLR Blog)

Su funcionamiento se puede entender a partir de la aplicación de la Ley de Faraday-Lenz y la de la fuerza de Lorentz, donde el campo magnético creado en el estator induce una tensión en el rotor.

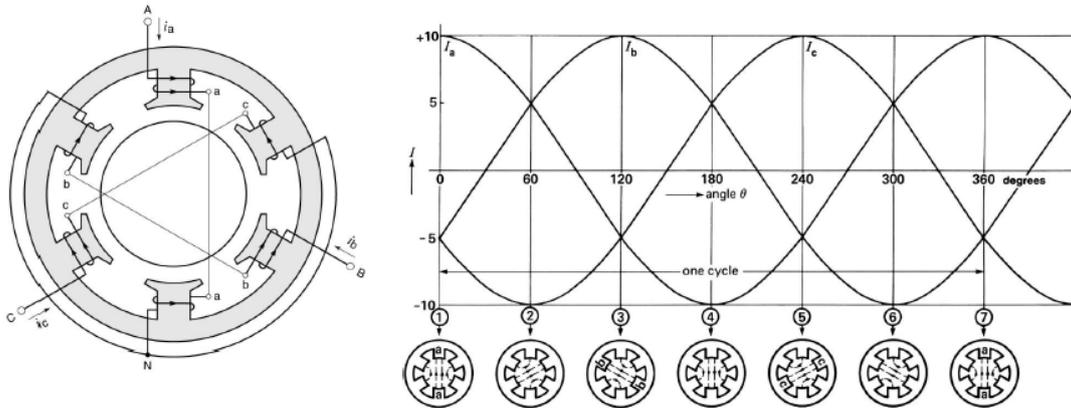


Il·lustració 22 Corriente de inducción y fórmula de Ley Faraday-Lenz (Fuente: Hyperphysics)

### 3.2.2 Principio de funcionamiento

Según el Teorema de Ferraris, cuando por las bobinas del estátor circulan corrientes trifásicas, se induce un campo magnético giratorio que envuelve el rotor. Este campo magnético variable va a inducir una tensión en el rotor según la ley de Faraday-Lenz. El campo magnético giratorio creado por el estátor, gracias a la corriente trifásica que pasa por los bobinados, atraviesa el rotor (de jaula de ardilla o bobinado), con lo cual se genera una fuerza electromotriz de inducción.

Gracias al campo giratorio originado y las corrientes creadas en el rotor, se consigue originar una fuerza electrodinámica sobre dichos conductores del rotor y así crear el movimiento giratorio del propio rotor.



Il·lustración 23 Teorema de Ferraris

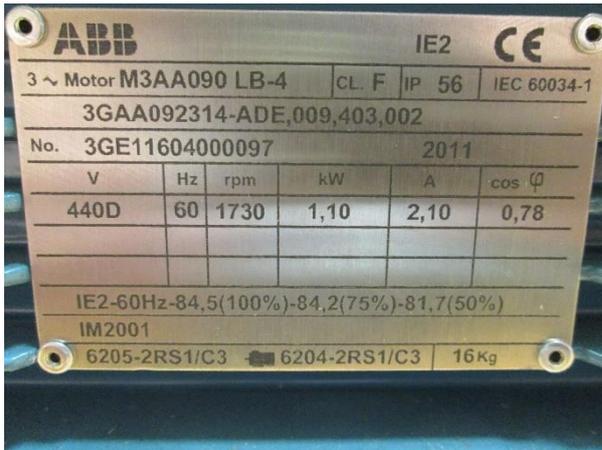
### 3.2.3 Especificaciones

Este motor es el segundo actuador más importante del proyecto. Él se encargará de subir y bajar el telón, elevar y descender la plataforma elevadora y hacer girar la plataforma giratorio tanto en sentido horario como en sentido antihorario.



Il·lustración 24 Motor ABB M3AA 90LB4

En la placa de características que viene integrada en el propio motor, se encuentran las siguientes características técnicas del motor:



Il·lustració 25 Placa de característiques tècniques del motor

La qual, se acompanya de una tabla aún més completo con información relevante como la corriente en vacío o la velocidad a par mínimo:

No.	Definición	Dato	Unidad
1	Producto	<i>TEFC, 3-phase, squirrel cage induction motor</i>	
2	Código producto	<i>3GAA 092 520-BSJ</i>	
3	Tipo/Carcasa	<i>M3AA 90LB 4</i>	
4	Posición de montaje	<i>IM3001, B5(flange)</i>	
5	Potencia nominal $P_N$	<i>1,1</i>	kW
6	Factor de servicio	<i>1</i>	
7	Servicio	<i>S1(IEC) 100%</i>	
8	Tensión nominal $U_N$	<i>400</i>	VY
9	Frecuencia nominal $f_N$	<i>50</i>	Hz
10	Velocidad nominal $n_N$	<i>1437</i>	r/min
11	Intensidad nominal $I_N$	<i>2,4</i>	A
12	Intensidad de vacío	<i>1,5</i>	A
13	Intensidad de arranque	<i>6,9</i>	A
14	Par a plena carga $T_N$	<i>7,3</i>	Nm
15	Par de arranque $T_s/T_N$	<i>3,3</i>	
16	Par máximo $T_{max}/T_N$	<i>3,8</i>	
17	Par mínimo $T_{min}/T_N$	<i>3,2</i>	
18	Velocidad a par mínimo	<i>225</i>	r/min

Tabla 5 Características técnicas motor ABB M3AA 90LB A

### 3.3 Motorreductor SUMITOMO CYCLO 6000

#### 3.3.1 Descripción

Un motorreductor es un sistema de engranajes que consiguen que un motor eléctrico reduzca o aumente su velocidad. El principal objetivo de este es reducir la velocidad del motor eléctrico sin que el propio mecanismo se resienta. De esta forma, junto con el

variador de frecuencia se consigue solventar las diferentes velocidades necesarias sin que el motor pierda su eficiencia y rigidez.

Otra de las funciones esenciales de este motorreductor será aumentar el torque para poder mover las diferentes cargas que estan presentes en los escenarios. La plataforma elevadora o la plataforma giratoria, precisan de este.

Entonces, ¿Cuál es la principal diferencia entre un variador de frecuencia y un reductor? El variador de frecuencia es capaz de mantener el par a la vez que reduce la velocidad, en cambio el reductor, a medida que bajamos la velocidad aumenta el par.

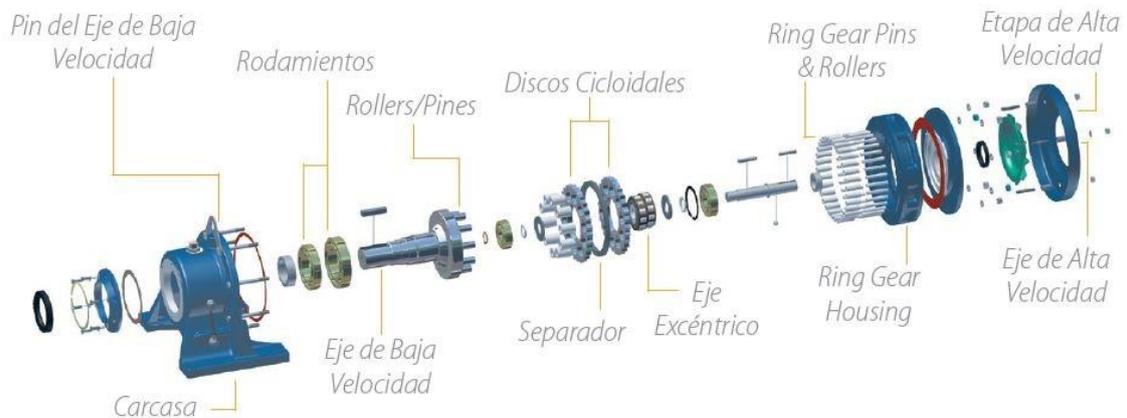


Ilustración 26 Motorreductor Cyclo 6000 desglosado (Fuente: Manual Sumitomo Drive Technologies)

### 3.3.2 Especificaciones

Las especificaciones que reporta el fabricante acerca del reductor son las siguientes:

- Par en la salida del eje calculado: 978 Nm
- RPM de entrada tarada: 1.450 rpm
- RPM de salida tarada: 8,79 rpm
- Potencia tarada del reductor: 0961 kW
- Par en la salida del eje: 940 Nm
- Capacidad de carga del reductor: 13.500 N



*Il·lustració 27 Reducto Cyclo 6000 Mod. CHHX-6135DBE-165/G90/A200*

### 3.4 Microcontrolador ARDUINO MEGA 2560

#### 3.4.1 Descripció

Arduino es una placa microcontroladora de código abierto. Esto significa que tiene una multitud de funcionalidades que permiten su aplicación práctica y la posibilidad de ser programado a través de cualquier plataforma<sup>10</sup>.

El código utilizado en Arduino es C++, un lenguaje de programación que facilita la interacción entre software y objetos físicos.

En un principio, la idea era utilizar un Autómata de ABB, pero a causa de la pandemia Covid-19, tuvimos a bien la decisión de cambiarlo por un Arduino, ya que el PLC se encuentra en la universidad y no ha sido posible acceder a él. Esto ha tenido una consecuencia positiva y otra negativa, pues las horas de dedicación necesarias para poder programar Arduino son mucho más elevadas que las horas requeridas para programar el autómata de ABB. La parte positiva es la ventaja económica, puesto que, el autómata ABB cuesta alrededor de 2.000€ y la placa Arduino cuesta apenas 13€.

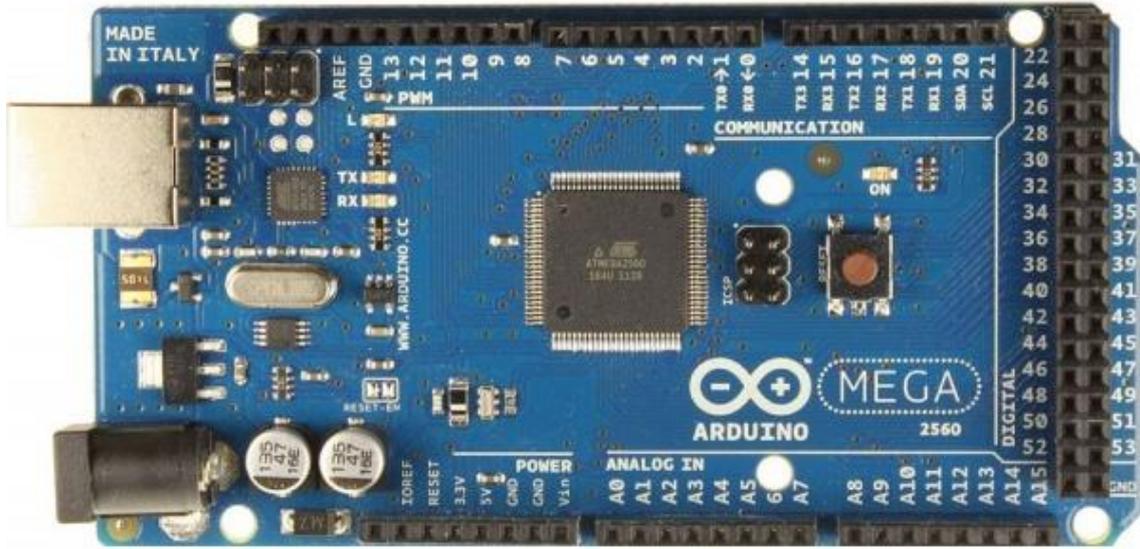
Para el exitoso desarrollo de este proyecto, la sustitución del autómata ha sido decisivo.

He debido adaptarme al lenguaje propio del código C++, con todo lo que ello conlleva. He empleado más horas, ya que he tenido que realizar un arduo proceso de aprendizaje para poder programar con él, atendiendo al IDE de Arduino<sup>11</sup>.

---

<sup>10</sup> Toda aquella programación basada en C++, posterior inclusión en IDE de Arduino.

<sup>11</sup> Programa capaz de interpretar C++. En el punto 3.4.3 se detalla su naturaleza y funcionamiento.



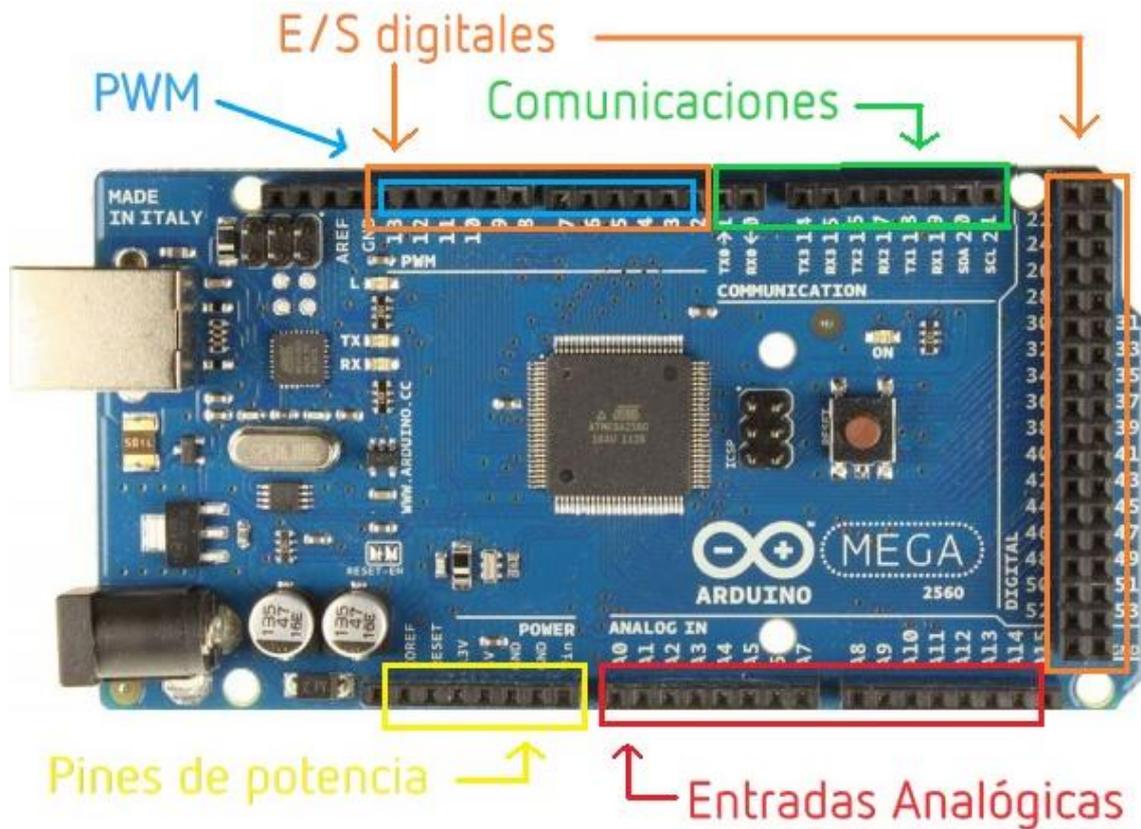
Il·lustració 28 Microcontrolador Arduino MEGA 2560 (Fuente: [www.arduino.com](http://www.arduino.com))

El variador de frecuencia dispone de 6 entradas digitales que se deberán activar o desactivar (mandando un 0 o un 1) dependiendo de la función u opción que se vaya a activar. Para poder activarlas se necesita una señal de +24V en CC, y aquí está el principal problema que nos plantea la utilización de Arduino ya que este es capaz de generar señales de +5V en CC. La solución es implementar un sistema de relés que se activarán a 5 V en CC con una fuente de 24 V de CC que la proporciona el propio variador.

### 3.4.2 Especificaciones

Arduino tiene varios modelos de microcontroladores, el más famoso es el Arduino UNO. El problema de este microprocesador es que únicamente cuenta con 11 entradas/salidas digitales, y este proyecto requiere de 25 entradas y salidas digitales. Es por este motivo que se ha escogido el modelo MEGA 2560.

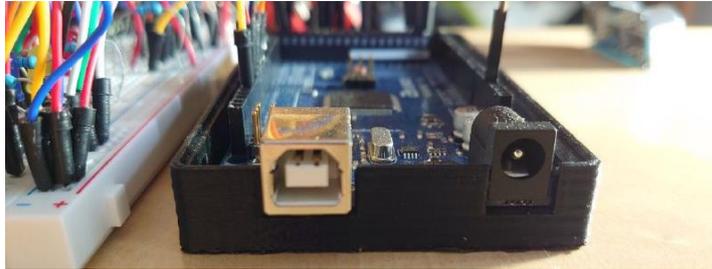
A continuación, se explica la función de los diferentes pines y puntos de conexión de los que dispone el microcontrolador:



Il·lustració 29 Arduino MEGA 2560 dividida en seccions, cada una amb una funció diferent

- E/S Digitales: El microcontrolador dispone de 44 entradas y salidas digitales operativas. Estos pines pueden recibir o transmitir señales de 5V o 0V. La corriente máxima que pueden aportar cada uno de los pines es de 40mA. Además, estos pines llevan incorporada una resistencia de entre 20-50 kΩ, que por defecto viene desconectada y que se conecta a través de IDE.
- E/S Digitales PWM: El propio microcontrolador trae incorporado un sistema PWM ("Pulse Wave Modulation" o Modulación de Onda por Pulsos) de 8 bits de resolución (valores del 0 al 255).
- Comunicaciones: Esta serie de pines, para este proyecto no van a ser utilizados. Son usados para recibir (RX) o transmitir (TX) datos a través del puerto TTL.
- Entradas Analógicas: La placa Arduino Mega dispone de 16 entradas analógicas con una resolución de 10 bits (1024 valores). Al igual que las comunicaciones, estos pines no se utilizarán.
- Módulo de potencia:
  - VIN: La Mega 2560 dispone de 3 formas de alimentación. La más utilizada comúnmente es la entrada USB que dispone la propia placa y por la cual se transfiere el software desde el PC hasta el Arduino. Por

otro lado, esta VIN por donde se alimenta el microcontrolador a través del pin, y por último una conexión de 2,1 mm.



*Ilustración 30 Conexión USB y jack para alimentación*

- 5V: Es la fuente de voltaje que proporciona la placa para poder alimentar a los diferentes componentes que se usan en el proyecto, como por ejemplo los sensores. Esta señal es estabilizada por el controlador y proviene de Vin, USB o la entrada de 2,1 mm.
- 3,3V: Es otra fuente que proporciona el controlador. Esta sirve para poder aumentar la intensidad máxima que nos permite la salida de 5V (40mA) hasta los 50mA. Esta es generada por un regulador interno.
- GND: Es el pin que se utiliza como toma de tierra o negativo.

Como especificaciones principales del Arduino Mega 2560:

- Microprocesador: ATmega2560
- Memoria de programa: 256 Kb
- Memoria SRAM de 8 Kb para datos y variables del programa
- Memoria EEPROM para datos y variables no volátiles
- Velocidad del reloj (o velocidad en la que se ejecuta el Loop) de 16 MHz

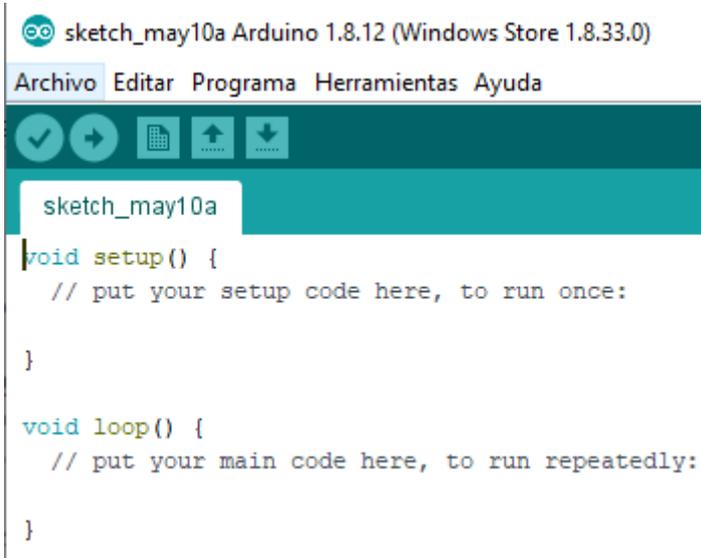
Este microcontrolador está protegido por un multifusible inicializable el cual, protege la conexión entre el PC y la conexión USB contra cortocircuitos y sobretensiones. Si se detectan más de 500 mA, el fusible automáticamente corta la conexión hasta que el cortocircuito o la sobretensión desaparece.

### 3.4.3 Arduino software IDE

IDE (Integrated Development Environment), es el programa que se utilizará para poder programar nuestro controlador. En el se escribirá el código necesario para el proyecto y una vez finalizado se trasladará hacia el Arduino Mega 2560.

Está basado en código C++ y es muy intuitivo. Se diferencia en 2 apartados:

- Setup: Donde se cargan todos los elementos que se requieran utilizar y definir las entradas y salidas.
- Loop: Es el bucle que se repite continuamente, realizando las acciones.



```
sketch_may10a Arduino 1.8.12 (Windows Store 1.8.33.0)
Archivo Editar Programa Herramientas Ayuda
sketch_may10a
void setup() {
  // put your setup code here, to run once:
}
void loop() {
  // put your main code here, to run repeatedly:
}
```

Ilustración 31 Programa Arduino IDE

## 3.5 Sensores y dispositivos de control

### 3.5.1 Finales de carrera

Para este proyecto se han seleccionado unos finales de carrera con rueda en la pletina. Esto facilitará la pulsación que realice ya sea por parte del telón como de las plataformas.

Para este caso se necesitará 4 ud. de estos:



Ilustración 32 Final de carrera AMZLAB V-155-1C25

El final de carrera está compuesto por 3 bornas, un pulsador y una palanca mecánica de accionamiento. Las 3 bornas son NO (Normalmente abierto), NC (Normalmente

cerrado) y COM (Común). Para nuestro proyecto se utilizará la borna NO y COM, porque se pretende que cuando se realice la pulsación, mande la señal de 5V.

Se utilizarán para:

- Detección del telón en posición abajo
- Detección del telón en posición arriba
- Detección de plataforma elevadora plegada
- Detección de vueltas de la plataforma giratoria

### 3.5.2 Interruptor

Para poder seleccionar el menú en el que se quiere entrar, si bien modo manual o modo automático, se instalarán 2 interruptores con indicador de servicio (LED).

Estos dispositivos, al contrario que los finales de carrera, solo darán la señal de 5V cuando sean presionados.



Ilustración 33 Interruptor tipo KCD1-8

Estos interruptores cuentan con 3 bornes:

- +: Este es el borne al que se conectará 5V.
- A: Es el automatismo que a su vez alimenta el led incorporado dentro el cual indica si el interruptor está en servicio o no
- GND: Es el encargado de cerrar el circuito para que pueda funcionar el LED. Si este no se conecta, el led no se encenderá, pero continuará funcionando como interruptor.

### 3.5.3 Encoder rotatorio con pulsador

Se trata de un dispositivo rotatorio que, en resumen, nos indicará cual es el sentido de su eje a través de los bornes CLK y DT.

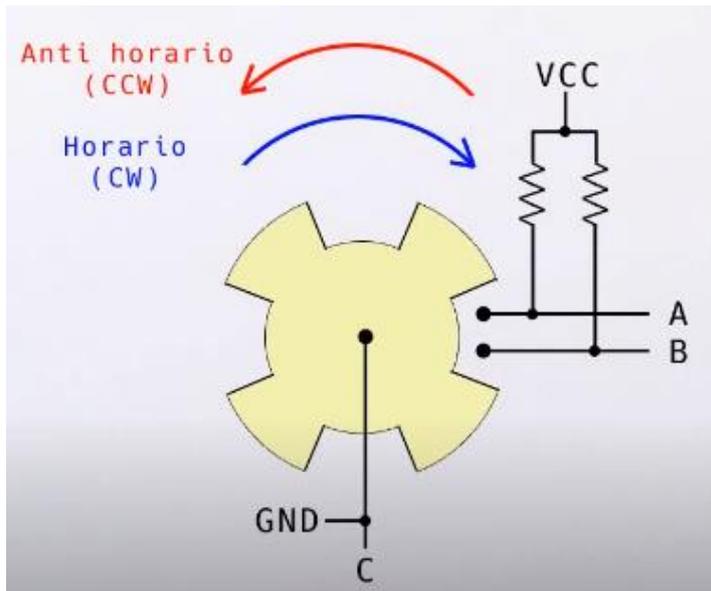


Ilustración 34 Composición y descripción de las partes que conforman el encoder

Este dispositivo tiene un eje, el cual, se puede rotar en sentido horario o antihorario sin tope, es decir, se puede girar infinitas veces. El encoder genera señales digitales que serán interpretadas por Arduino Mega 2560 a través del programa cargado y creado con IDE Arduino.

El modelo de encoder, el KY-040, viene equipado con un pulsador que se utilizará en el proyecto como selector de acciones. Este se activa presionando el eje. Para crear la señal, uno de sus extremos, está conectado a GND.



Ilustración 35 Encoder KY-040 de AZ Delivery

Para el proyecto, es muy necesario ya que, gracias a él, se podrá navegar a través de los menús automático y manual para seleccionar la acción que se quiera activar desde el variador.

Pero, ¿y cómo funciona? A y B son los denominados de señal mientras que C, es el común. C es el plato dentado que veíamos en la Ilustración 19 que realmente son 30 dientes, pero para su explicación he preferido mostrar un engranaje sencillo. De cada uno de los pines A y B, se conectan una resistencia de 10 kΩ como resistencia PullUP, que se comentará en los siguientes apartados.

Veamos su funcionamiento a través de los gráficos que generan A y B:

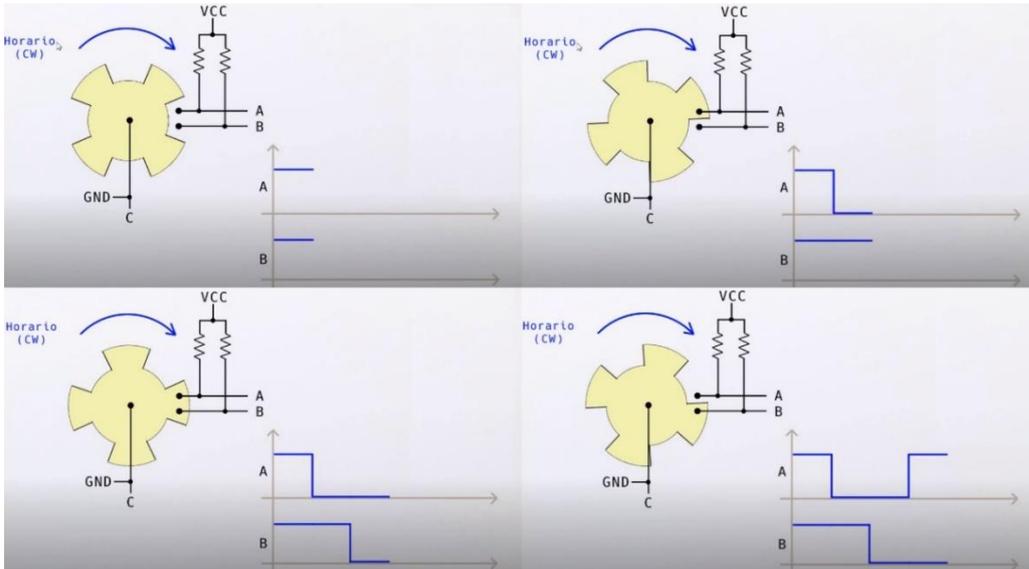


Ilustración 36 Gráficos de generación de señales de A y B en sentido horario

Cuando se gira el codificador en sentido horario, primero ocurre el pulso en el terminal A y luego sobre B.

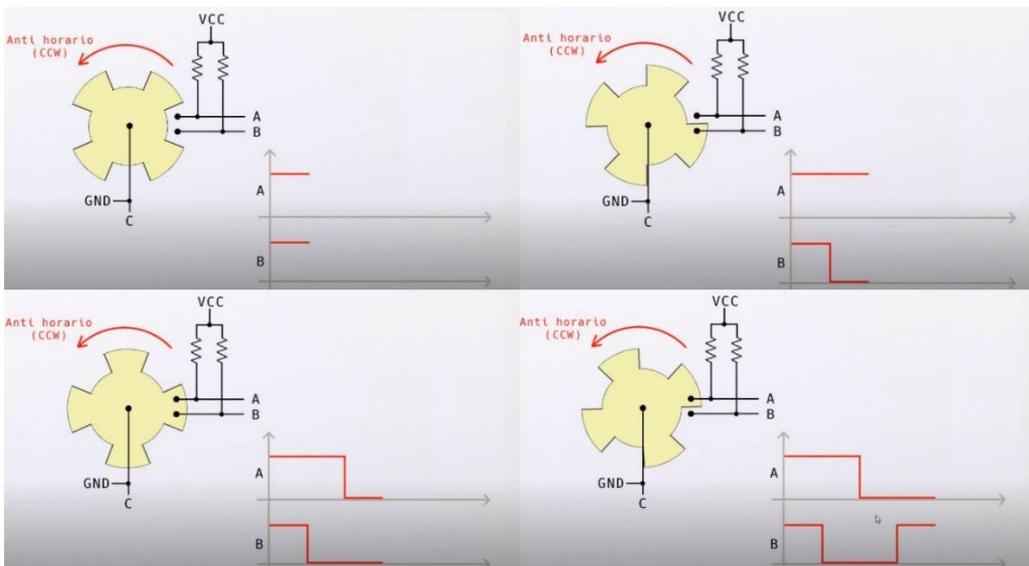


Ilustración 37 Gráficos de generación de señales de A y B en sentido antihorario

Cuando gira el codificador en sentido antihorario, primero ocurre el pulso en el terminal B y luego sobre A.

Una vez sabido esto, determinamos que cuando  $A = 0$  (LOW) y  $B = 1$  (HIGH), el sentido de giro será horario y si  $A = 0$  (LOW) y  $B = 0$  (LOW), el sentido de giro será antihorario. Todo esto luego hay que trasladarlo al código de Arduino para que sea capaz de interpretar el sentido de giro.

### 3.5.4 Sensor ultrasónico

Para poder detectar la altura a la que está la plataforma elevadora en el proyecto, se precisa un sensor ultrasónico que lo que hace es medir la altura a la que se encuentra el objeto superior a través de unos pulsos emitidos que rebotan en el objeto y los recibe un receptor que determina a través del tiempo transcurrido entre el disparo o "Trigger" los cm a los que está la plataforma.

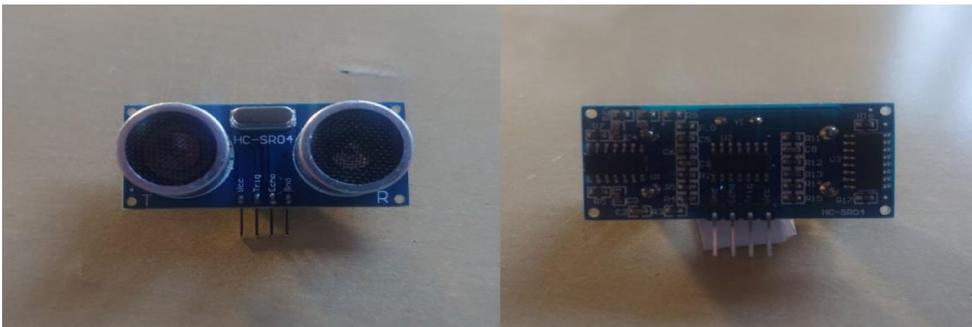


Ilustración 38 Sensor ultrasónico HC-SR04

El sensor cuenta con 4 patillas que determinarán su correcto funcionamiento:

- Vcc, que corresponde a la alimentación del sensor
- Trigg, lanzará el disparo hacia el objeto para determinar a qué distancia está del sensor
- Echo, recogerá el disparo creado por Trig y calculará el tiempo que ha transcurrido entre el disparo y la recepción.
- GND, corresponde a la toma tierra o negativo del módulo

Para que el IDE de Arduino pueda leer los datos, hay que hacer una conversión de datos recibidos para determinar la distancia en cm, ya que el propio sensor mide tiempo y se requiere conocer la distancia. La conversión que especifica el fabricante es:

$$DISTANCIA (cm) = \frac{TIEMPO}{58,2}$$

Con esta fórmula se obtiene pues, la distancia en cm que está recibiendo el sensor.

### 3.5.5 Relés

En el proyecto nada mas empezar se planteaba un problema. El microcontrolador Arduino solo es capaz de entregar 5V en continua como mucho y el variador necesita una señal de 24 V en CC para poder activar las entradas digitales de paro, marcha, velocidad 1... La solución ha sido plantear una instalación con relés, de esta forma, el sistema en ved de mandar las señales de 24 V directamente, le enviará una señal de 5V a un relé, y este dejará pasar los 24V necesarios.

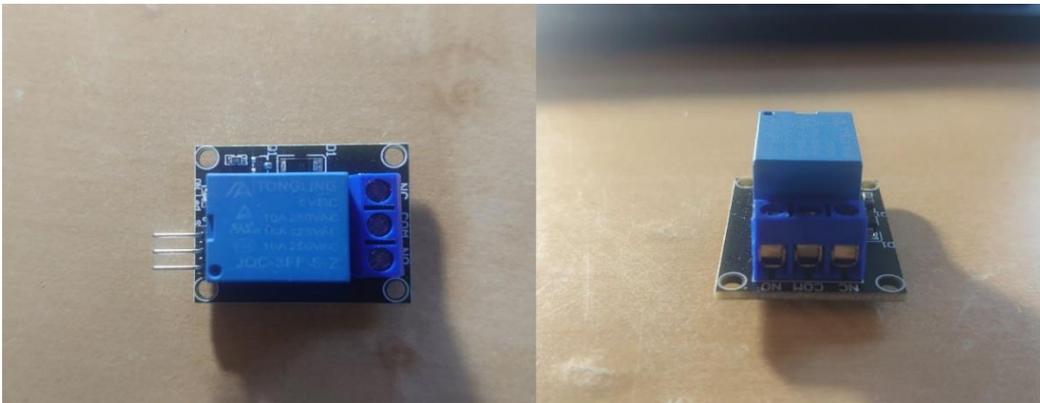


Ilustración 39 Módulo de Relé TONGLING JQC-3FF-S-Z

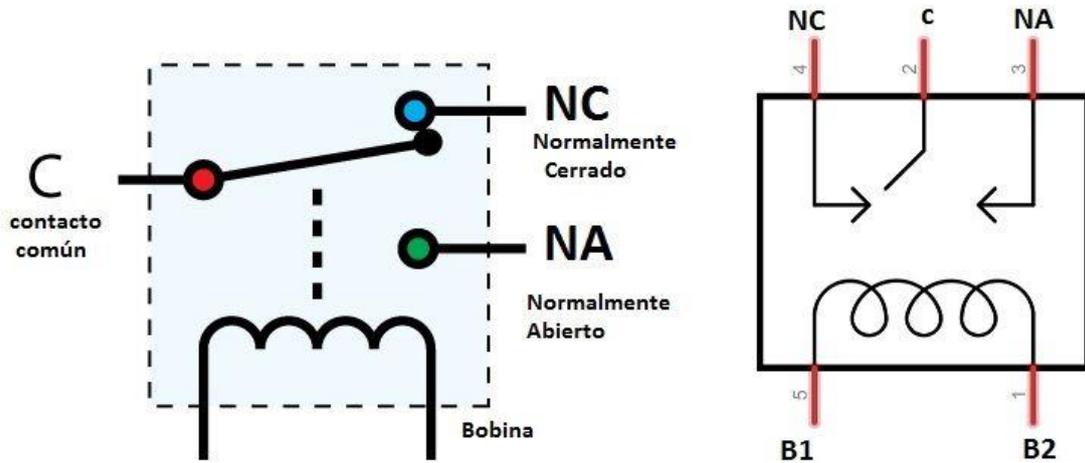
Con el relé se puede trabajar con tensiones mucho más altas que las que nos permite Arduino.

Un relé, es un dispositivo electromecánico que permite controlar cargas de tensión o corriente superiores a las que proporciona Arduino.

El relé que se utilizará dispone de un circuito NA (Normalmente abierto) y otro NC (Normalmente cerrado). En este caso solo requiere que se conecte cuando le de la señal, por lo tanto, lo conectaremos en NA.

Por otro lado, tiene 3 pines de conexión:

- Vcc, tensión de alimentación
- GND, toma tierra o negativo para cerrar el circuito
- S, sería la señal que manda el Arduino para poder activarlo



**Al meter corriente por la bobina los contactos abiertos se cierran y los cerrados se abren.**

*Ilustración 40 Circuito interno del relé (Fuente: Tecnología)*

Sus especificaciones son las siguientes:

- Corriente máxima soportada en VCA: 10 A
- Corriente máxima soportada en VCC: 10<sup>a</sup>
- Voltaje de trabajo en CA: 125 – 250 V
- Voltaje de trabajo en CC: 0 – 30 V

### 3.5.6 Pantalla LCD

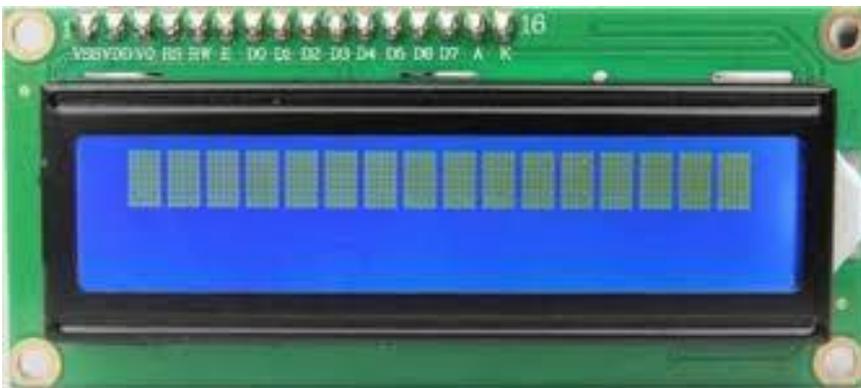
Es un dispositivo con pantalla de cristal líquido formado por dos filas de dígitos, con 16 caracteres cada fila que se suele utilizar para la comunicación entre la máquina y el hombre. En la pantalla se puede mostrar todo tipo de información sin importar qué tipo de símbolos o caracteres son.

Para el proyecto, la LCD es esencial ya que sin ella no se sabría el modo actual o que opciones escoger. Por otro lado, contar con ella es una gran ventaja ya que se ahorra por lo menos 10 pulsadores/interruptores, así como conseguir una pantalla con información acerca del estado del microcontrolador.

Las especificaciones de la LCD y su configuración es la siguiente, ya que cuenta con 16 pines para poder transmitir la información a través de la pantalla:

- PIN 1: VSS, es nuestra toma tierra o masa.
- PIN 2: VDD, es el positivo de tensión a 5V.

- PIN 3: V0, se utilizará para el contraste de la pantalla, es decir, más o menos brillo según las condiciones. A este pin se llega con una resistencia en forma de potenciómetro de hasta 10kΩ, de esta forma se consigue regular la pantalla según en las condiciones del ambiente.
- PIN 4: RS, es el selector de registro, se encargaría de las tareas de bajo nivel
- PIN 5: R/W, es para designar si la pantalla leerá los datos que se envía desde Arduino o para registrar lo que se mande a la placa. Este terminal se direcciona a masa para indicarle que lo se utiliza en modo escritura.
- PIN 6: E, indicamos el tipo de registro a utilizar para cada caso. Esto se le comunica a través de Software mediante la librería LiquidCrystal
- PIN 7 → 14: D0 → D7, es el bus de datos mediante el cual se envía la información necesaria que reproduzca por pantalla. Gracias a este módulo LCD, es posible realizar las lecturas utilizando solo 4 bits, es decir, 4 entradas.
- PIN 15: A, ánodo del led que ilumina la parte trasera de la pantalla a modo de retroiluminación. Este se conectará a una resistencia de 220 Ω para que el brillo no sea tan excesivo.
- PIN 16 K, cátodo del led que ilumina la parte trasera de la pantalla a modo de retroiluminación.



*Ilustración 41 LCD de 16x2 para Arduino (Fuente: desucbreArduino)*

### 3.5.7 Resistencias PULL-UP & PULL-DOWN

En el proyecto se incluyen pulsadores e interruptores. Los microcontroladores son muy sensibles y la no instalación de resistencias pull-up o pull-down harán que Arduino no trabaje bien y cometa errores. Por ejemplo, si no se instalasen las resistencias, se verá que, pulsando una vez, quizás se pulsa 5 veces de golpe o se pulsa sin tocar el botón, etc.

Esto es producido por el ruido eléctrico que hay dentro del circuito. Para evitar esto, se usan las configuraciones de resistencias de pull-up y pull-down.

Las resistencias Pull-Up, consisten en una resistencia entre el Vcc y la entrada del microcontrolador, por debajo de la resistencia está el pulsador conectado a tierra. Cuando no se pulsa, el controlador está recibiendo una señal que es prácticamente igual que Vcc con lo cual lo interpreta como un 1. Si pulsamos, la corriente bajará hasta tierra (por el camino con menos resistencia) y a la entrada del microcontrolador apenas llegará corriente, con lo cual Arduino interpretará un 0.

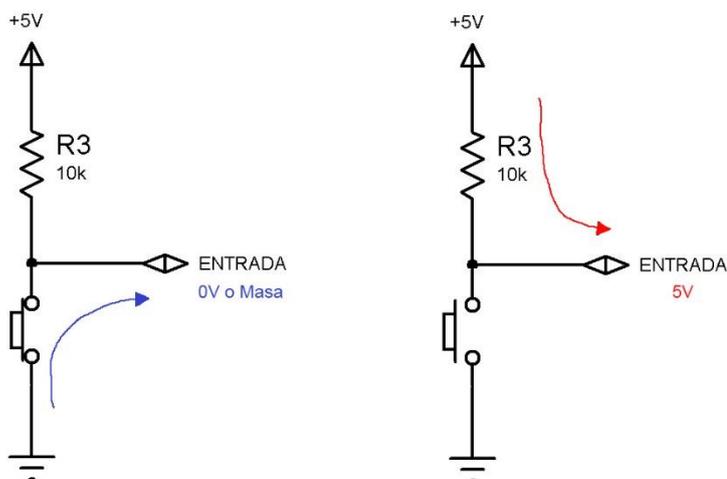
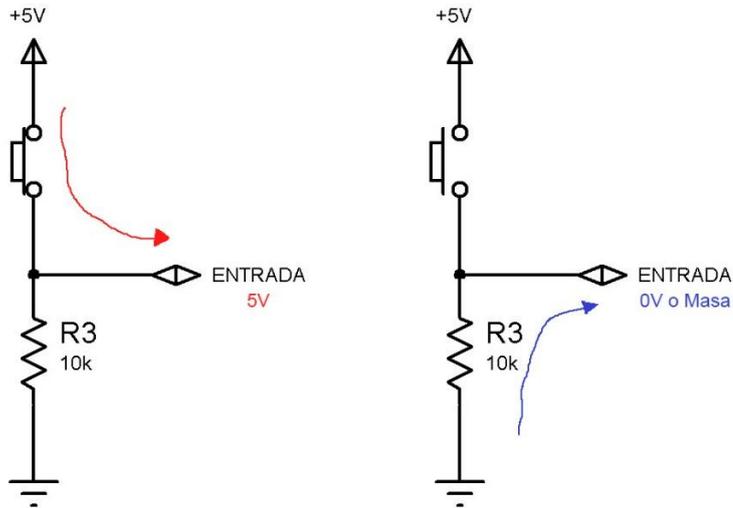


Ilustración 42 Esquema de resistencia Pull-Up (Fuente: zonamaker)

Las resistencias Pull-Down, consiste en lo contrario a pull-up, la resistencia está conectada a GND y el pulsador está conectado a VCC. En el punto medio entre la resistencia y el pulsador, se conecta la señal de Arduino. En este caso si no se pulsa nada, es prácticamente 0 o zero lo que le está llegando a la entrada de Arduino y si se pulsa, empezará a llegar corriente desde Vcc y el microcontrolador interpretará un 1 lógico.



Il·lustració 43 Esquema de resistència Pull-Down (Fuente:zonamaker)

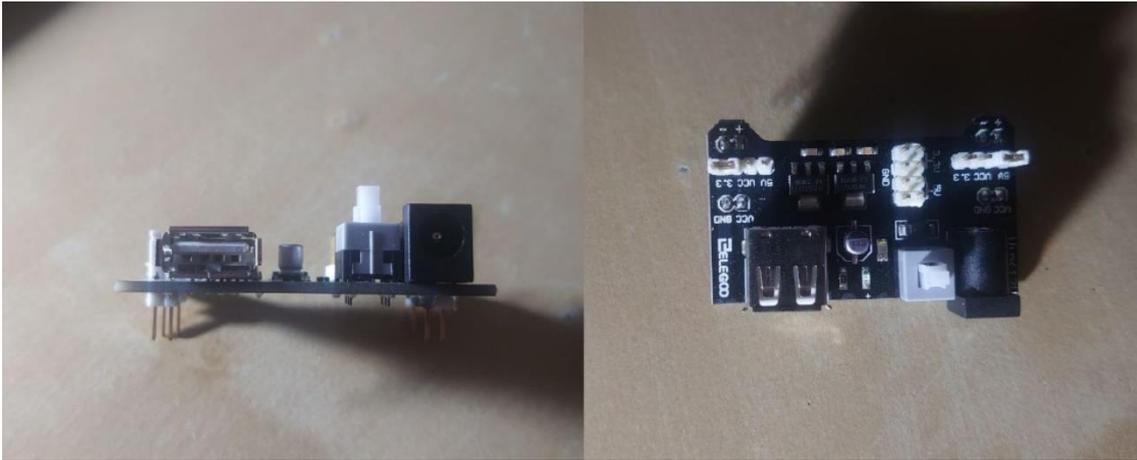
De esta forma se evitan lecturas incorrectas y se obtiene siempre una señal que va a ser siempre o 0 o 1, que es lo que necesita Arduino.

Las resistencias implementadas para todos los pulsadores, finales de carrera e interruptores son de 10 k $\Omega$ .

### 3.5.8 Módulo de alimentación

Para poder alimentar Arduino con una fuente sin variaciones, se hará a través de un módulo de fuente de alimentación específico para Arduino. Este, simplemente se alimenta a través de una fuente de entre 6,5V y 12V y tiene una salida de 3,3V a 5V.

La gran ventaja es que aporta más intensidad que la que puede aportar la propia placa Arduino. Hasta 700 mA, a comparación de Arduino que ofrece máximo 40 mA.



*Il·lustració 44 Mòdul de font de alimentació ELEGOO Power MB v2*

## 4 SOFTWARE ARDUINO

La programació del Microcontrolador Arduino Mega 2560 per a nostre projecte se ha desenvolupat a través del software IDE 1.8.12. Per a este software, el llenguatge de programació de Arduino està basat en llenguatge C++.

La programació a través de Arduino està pensada per a ser una eina informàtica per al desenvolupament de programació a través del PC i que pugui treballar amb elements electrònics de forma còmoda i ràpida.

El codi que hem creat per a este projecte conté 626 línies de codi, les quals, explicaré a continuació. El codi complet serà adjuntat en el Annex I.

Al principi del codi, se han descrit una sèrie de variables i estats que fan referència a entrades i sortides digitals, així com de variables que canvien d'estat LOW a HIGH. L'estat LOW, fa referència a que se mandi una senyal o que se mantingui en 0V tant una sortida/entrada digital del microprocessador com de les estats. En canvi, l'estat HIGH se utilitza per a enviar la senyal de 5V en CC.

Un estat és una variable que al llarg del codi font se utilitzarà per a guardar en què situació se troben els pins digitals, o fins i tot, per a crear estats virtuals que utilitzarem per a ajudar-nos en la programació.

Al llarg del projecte, i en coneixement de que la memòria del model Arduino Mega 2560 que se utilitzarà en el projecte és àmplia, se utilitzaran en la seva majoria variables de tipus int. Aquestes variables representen un conjunt d'enters de 32 bits el seu rang va

desde -2.147.483.648 hasta 2.147.483.647 en los que además de almacenar estos enteros, es capaz de comprender, sumas, restas y multiplicaciones.

```
int C = 3; // Definimos que el CLK del encoder es la entrada digital 6 (Automático)
int D = 5; // Definimos que el DT del encoder es la entrada digital 5 (Automatico)
int PULSADOR2 = 6; // Definimos que el SW del encoder es la entrada digital 3 (Automatico)
int A = 2; // Definimos que el CLK del encoder es la entrada digital 6 (Manual)
int B = 4; // Definimos que el DT del encoder es la entrada digital 5 (Manual)
int PULSADOR = 7; // Definimos que el SW del encoder es la entrada digital 3 (Manual)

int RELE1 = 8; // Definir salida de PARO/MARCHA
int RELE2 = 9; // Definir salida de AVANCE/RETROCESO
int RELE3 = 10; // Definir salida de VELOCIDAD 1
int RELE4 = 11; // Definir salida de VELOCIDAD 2
int RELE5 = 12; // Definir salida de RAMP1/RAMPA2

int FINALCARRERAUP = 22; // Definir entrada del final de carrera dispuesto abajo del telón
int FINALCARRERADOWN = 23; // Definir entrada del final de carrera dispuesto arriba del telón
int FINALCARRERAPLAT = 51; // Definir entrada del final de carrera dispuesto abajo de la pataforma
int FINALCARRERAGIRA = 50; // Definir entrada del final de carrera dispuesto en la platafoma giratoria

int ESTADOTELONUP = LOW; // Nos indica si el telon esta subiendo
int ESTADOTELONDOWN = LOW; // No indica si el telon esta bajando
int ESTADOPLATUP = LOW; // Indica que plataforma esta subiendo
int ESTADOPLATDOWN = LOW; // Indica que plataforma esta bajando
int ESTADOGIRA = LOW; // Indica que plataforma giratoria esta girando en sentido horario
int ESTADOGIRAANTI = LOW; // Indica que plataforma giratoria esta girando en sentido antihorario
int ESTADOGIRAS = LOW; // Indica que plataforma giratoria esta girando hasta que cuente 5
int CONTADORGIRO = 0; // Cuenta el n° de vueltas que lleva la plataforma giratoria
```

*Ilustración 45 Variables utilizadas en el desarrollo del proyecto*

Cuando a un int se le adjudica un número, como por ejemplo RELE1 = 8, se refiere a que la salida de la señal que active o desactive el RELE1 (paro/marcha) se enviará a través del PIN 8 incorporado en Arduino Mega 2560.

El código contiene 2 partes diferenciadas y esenciales, el voidsetup() y el voidloop(): El voidsetup() corresponde a las líneas de código que se ejecutan al inicio del programa, mientras que voidloop() contiene las líneas de código que se ejecutan en bucle de principio a fin e indefinidamente.

En voidsetup() declaramos las entradas y salidas de nuestros equipos digitales por tal de que Arduino sea capaz de leerlas, en nuestro caso hemos tenido que declarar:

Nº de pin	Tipo de entrada	Dispositivo	Cometido
53 y 52	53: OUTPUT 52: INPUT	Sensor ultrasónico	Enviar y recibir la señal ultrasónica producida por TRIG y recogida por ECO
40	INPUT	Interruptor modo Manual	Encargado de mandar la señal digital de 5V para entrar en el menú Manual

41	INPUT	Interruptor modo Automático	Encargado de mandar la señal digital de 5V para entrar en el menú Automático
3, 5 y 6	3: INPUT 5: INPUT 6: INPUT	Encoder Automático 1:	Enviar señales digitales de 5V a Arduino para definir en qué posición se encuentra. Estos están compuestos por DT, CLK y SWITCH.
2, 4 y 7	2: INPUT 4: INPUT 7: INPUT	Encoder 2: Manual	Enviar señales digitales de 5V a Arduino para definir en qué posición se encuentra. Estos están compuestos por DT, CLK y SWITCH.
8	OUTPUT	Relé 1	Es una salida digital que activa el relé para mandar una señal de 24 V al variador y activar PARO/MARCHA
9	OUTPUT	Relé 2	Es una salida digital que activa el relé para mandar una señal de 24 V al variador y activar AVANCE/RETROCESO
10	OUTPUT	Relé 3	Es una salida digital que activa el relé para mandar una señal de 24 V al variador y activar VELOCIDAD 1
11	OUTPUT	Relé 4	Es una salida digital que activa el relé para mandar una señal de 24 V al variador y activar VELOCIDAD 2
12	OUTPUT	Relé 5	Es una salida digital que activa el relé para mandar una señal de 24 V al variador y activar RAMPA 1/RAMPA 2
22	INPUT	Final de carrera Telón 1	Manda 0V o 5V para indicar si el telón está subido. (0V telón no detectado)
23	INPUT	Final de carrera Telón 2	Manda 0V o 5V para indicar si el telón está abajo. (0V telón no detectado)
50	INPUT	Final de carrera plataforma elevadora	Manda señal de 0V o 5V si la plataforma está plegada. (0V, plataforma elevada)
51	INPUT	Final de carrera plataforma giratoria	Manda señal de 0V o 5V si la plataforma acciona a su paso el final de carrera.

30, 31, 32, 33, 34 y 35	30: INPUT 31: INPUT 32: INPUT 33: INPUT 34: INPUT 35: INPUT	Pantalla LCD	Enviar los bits necesarios para poder escribir por pantalla el texto que le dictamine Arduino
-------------------------------	--	--------------	---

*Tabla 6 Identificación de pines digitales requeridos*

Para poder utilizar la LCD, dentro de IDE deberemos descargar y compilar la librería de Arduino LiquidCrystal. Esta es la encargada de encender y apagar cada píxel de la LCD para poder conformar una palabra. Dentro de voidsetup(), deberemos declarar que tipo de LCD está trabajando, en este caso, es un LCD de 16 filas y 2 columnas, por lo tanto lo iniciamos como tal.

```
void setup() { //Iniciamos la pantalla y los INPUTS y OUTPUTS
  lcd.begin(16, 2);

  pinMode(A, INPUT);
  pinMode(B, INPUT);
  pinMode(PULSADOR, INPUT_PULLUP);
  pinMode(C, INPUT);
  pinMode(D, INPUT);
  pinMode(TRIG, OUTPUT);
  pinMode(ECO, INPUT);
  pinMode(FINALCARRERAUP, INPUT);
  pinMode(FINALCARRERADOWN, INPUT);
  pinMode(FINALCARRERAPLAT, INPUT);
  pinMode(FINALCARRERAGIRA, INPUT);
  pinMode(PULSADOR2, INPUT_PULLUP);
  pinMode(RELE1, OUTPUT);
  pinMode(RELE2, OUTPUT);
  pinMode(RELE3, OUTPUT);
  pinMode(RELE4, OUTPUT);
  pinMode(RELE5, OUTPUT);
  pinMode(MANUAL, INPUT);
  pinMode(AUTO, INPUT);
}
```

*Ilustración 46 Inicialización de entradas y salidas digitales en void setup()*

En voidloop(), se llamará a las funciones creadas para que se vayan inicializando a medida que el microcontrolador avanza por las diferentes líneas de código. En él se ejecutarán los comandos y acciones que se hayan programado con anterioridad por tal de dar funcionalidades automáticas.

Cuando Arduino inicie el voidloop(), empezará por el primer comando e irá recorriendo toda la secuencia hasta el final. Una vez llegado al final, volverá al principio del comando e iniciará de nuevo la secuencia. Y de esta forma se repetirá una infinidad de veces. Este proceso se ejecuta a una velocidad de 16 MHz.

Cuando el microcontrolador se inicia, se inicia un reloj interno que lleva incorporado. Este es muy útil porque se puede utilizar en IDE a través de la función millis(), que devuelve el tiempo transcurrido desde su puesta en marcha de forma indefinida.

Al comenzar el voidloop(), hay 3 funciones que son necesarias para el correcto funcionamiento de los encoders y del sensor de posición de la plataforma elevadora:

1. Definición de posición del encoder 1 y 2 que navegará por el menú Manual o Automático. Su lógica funciona comprobando si la posición actual es diferente que la posición anterior. Si esto se cumple, la variable ANTERIOR pasará a valer lo mismo que POSICION, determinando la posición actual.

```
if (POSICION != ANTERIOR){ // Determina la posicion del encoder (Manual)
  ANTERIOR = POSICION;
}

if (POSICION2 != ANTERIOR2){ // Determina la posicion del encoder (Auto)
  ANTERIOR2 = POSICION2;
}
```

*Ilustración 47 Función que determina la posición en la que se encuentran los encoders*

2. Definición de la altura a la que se encuentra la plataforma elevadora. Para ello se indica a la variable TRIG que mande un pulso. Después se recoge este dato a través de la variable ECO y se guarda en la variable DURACION. Como este dato, el sensor ultrasónico, lo devuelve en segundos, hay que aplicar la relación de 58,2 para pasarlo a cm y se guarda en la variable DISTANCIA.

```
digitalWrite(TRIG, HIGH); // Determina la distancia del objeto al emisor de ultrasonidos
if(millis() >= TiempoAhora + periodo){
  TiempoAhora = millis();
  digitalWrite(TRIG, LOW);
}
DURACION = pulseIn (ECO, HIGH);
DISTANCIA = DURACION / 58.2;
```

*Ilustración 48 Determinación de la altura a la que se encuentra la plataforma elevadora*

Siguiendo el transcurso del loop, se inicia el proceso de determinación del estado de los interruptores que indican si queremos acceder al modo manual o al modo automático. Para hacer esta valoración, contamos con una sentencia condicional, con la que se trabajará durante todo el código. Estas sentencias determinan la acción que se deberá realizar si se cumplen los requisitos del "test" efectuado. Se trabajará con la estructura if, else if y else que en español se traduce en: si se cumplen las condiciones, haz esta acción, sino haz esta otra y sino haz esta última o no ejecutes ninguna acción.

Acciones a realizar por Arduino cuando se presionan los interruptores de entrada en ambos menús:

Posición de los interruptores	Acciones y mensajes
MANUAL: <b>OFF</b> AUTOMÁTICO: <b>OFF</b>	Mensaje en pantalla: "VARIADOR FREQ. ABB-ACS-580-R1"
MANUAL: <b>ON</b> AUTOMÁTICO: <b>OFF</b>	Mensaje en pantalla: "MODO MANUAL" y activación y entrada del modo manual. Imposibilidad de accionar ningún relé de forma manual si hay algún proceso del modo automático en ejecución.
MANUAL: <b>OFF</b> AUTOMÁTICO: <b>ON</b>	Mensaje en pantalla: "MODO AUTOMATICO" y activación y entrada del modo automático. Desactivación de cualquier relé activado en manual.
MANUAL: <b>ON</b> AUTOMÁTICO: <b>ON</b>	Mensaje en pantalla: "ERROR SELECCIONA UNA OPCION" y desactivación de todos los relés activados.

*Tabla 7 Funciones de los interruptores una vez accionados*



*Ilustración 49 Mensaje de error cuando Manual y Automático están en ON*

En el caso que se esté ejecutando una acción en modo automático y se accione cualquiera de los dos interruptores:

1. Manual OFF y Auto OFF: Se dejará finalizar la acción que se esté llevando a cabo. Por ejemplo, el telón está subiendo, pero mientras este sube, se ha accionado ambos interruptores, en este caso se dejará finalizar la subida del telón.
2. Manual ON y Auto OFF: Ocurrirá lo mismo que en el punto nº1. Hasta que el proceso en ejecución no haya finalizado, no se podrá activar ni desactivar ningún relé.
3. Manual ON y Auto ON: Se finalizará el proceso en ejecución de forma inmediata. Por ejemplo, la plataforma elevadora se está elevando hacia la altura deseada, esta frenará en seco y se quedará paralizada. Para su reinicio, se deberá devolver los elementos a sus puntos de origen.

Para la escritura en pantalla de los diferentes mensajes que se deben mostrar en cada caso, se utilizará la función `lcd.setCursor`, en la que se indica a Arduino que en la fila *x* y columna *y*, se escriba el texto correspondiente.

```
if ((digitalRead(MANUAL) == LOW) && (digitalRead(AUTO) == LOW)){  
  lcd.setCursor(0,0);  
  lcd.print(" VARIADOR FREQ. "); // Imprime por pantalla  
  lcd.setCursor(0,1);  
  lcd.print(" ABB ACS-580-R1 ");
```

*Ilustración 50 Interruptor Manual y Automático desactivados y programación de escritura de texto*

Cabe recordar que al inicio de la programación de Arduino se incluye una librería, que determina que píxel debe estar encendido en cada fila y columna para la proyección por pantalla LCD.



*Ilustración 51 Mensaje de Variador de frecuencia cuando manual y automático se encuentran en OFF*

## 4.1 Modo manual

En este modo se podrá activar o desactivar cualquier relé de forma manual. De este modo, se mandará consignas de forma independiente al variador para determinar las acciones que debe realizar el motor.



*Ilustración 52 Mensaje que muestra Arduino para indicar que se trabaja en Modo Manual*

Para poder navegar por el menú, se hará uso de un encoder, el cual, como se ha comentado anteriormente, determina la posición mediante pulsos. Este menú tiene 6 submenús:

Posición del encoder	Mensaje	Acción
0	MODO MANUAL	Sin acción
1	FUNCION PARO/MARCHA	Activará o desactivará el relé 1 asociado a paro/marcha a través del pulsador/encoder
2	FUNCION AVANCE/RETROCESO	Activará o desactivará el relé 2 asociado a avance/retroceso a través del pulsador/encoder
3	MOTOR VELOCIDAD 1 EN	Activará o desactivará el relé 3 asociado a velocidad del motor 1 a través del pulsador/encoder
4	MOTOR VELOCIDAD 2 EN	Activará o desactivará el relé 4 asociado a velocidad del motor 2 a través del pulsador/encoder
5	ARRANQUE MOTOR RAMPA 1/RAMPA 2	Activará o desactivará el relé 5 asociado a la rampa de aceleración/deceleración 1 u 2 a través del pulsador/encoder
6	PARA VELOCIDAD 3 ACTIVAR V1 Y V2	A modo informativo, indica que para activar la velocidad 3 del motor, V1 y V2 deben estar activadas

*Tabla 8 Funciones de cada uno de los submenús de Modo Manual*



*Il·lustració 53 Mensaje por pantalla de submenú Rampa1/Rampa 2 en Modo Manual*

Dado que la activación de los distintos relés se realiza mediante un pulsador, se ha trabajado con estados.

Lo que se ha buscado en este sistema es que con un mismo pulsador se pueda activar y desactivar cualquier relé para hacerlo más compacto y sencillo. De esta forma surge una problemática; el pulsador emite pulsos de 5V CC.

Para poder utilizar un pulsador para activar cada uno de los relés, se debe guardar en el flanco de bajada que genera el pulsador después de ser accionado:

```

void avanceRetroceso(){ // Activa o desactiva el AVANCE o el RETROCESO
  int estadoRele2 = digitalRead(RELE2);

  if (digitalRead(PULSADOR) == LOW && ESTADO2 != HIGH && POSICION == 2){
    digitalWrite(RELE2, !estadoRele2);
    ESTADO2 = HIGH;
    M2 = HIGH;
  }
  else if (digitalRead(PULSADOR) == HIGH && ESTADO2 != LOW && POSICION == 2){
    ESTADO2 = LOW;
  }
}
  
```

*Il·lustració 54 Còdigo de activación de Avance o Retroceso del motor*

Primeramente, se debe aclarar que el pulsador incorporado en el encoder, está continuamente en estado HIGH, ya que la patilla está en contacto permanente con GND y cierra el circuito, es por eso que se determina el flanco de bajada como condicionante de activación.

El ESTADO2 se ha declarado al principio del código con el valor de LOW. En primera instancia, Arduino lee el valor actual del RELE2 y lo almacena en una variable estadoRele2, después comienza a interpretar la sentencia condicional if.

Pongamos por ejemplo el inicio, el relé 2 se encuentra en estado LOW y nos encontramos en el submenú "FUNCION AVANCE/RETROCESO (posición 2 para el encoder). En if, se cumple la segunda y tercera condición, pero no la primera. En else if se cumple la primera y la última.

Cuando se pulsa el switch, se cumple el condicionante if; en ese momento entra en la sentencia, le manda a RELE2 al contrario de su condición actual (recordamos que el relé al principio estaba en LOW, ahora pasará a HIGH) y cambia el ESTADO2 a HIGH.

Justo acabar la sentencia de if, entrará en else if ya que se cumplen todas las condiciones y cambiará el ESTADO2 a LOW. Si volvemos a empezar, la variable estadoRele2 estará en HIGH y cuando entremos en la sentencia de nuevo, se cambiará a LOW, y así, relé 2 pasará a LOW y se desactivará.



*Ilustración 55 Mensaje por pantalla de submenú Avance/Retroceso en Modo Manual*

## 4.2 Modo automático

El modo automático de nuestro sistema con Arduino permite activar y desactivar variables de forma automática a modo de PLC. Esto es posible gracias a los diferentes sensores instalados en cada una de las situaciones que se describen a lo largo del capítulo.



*Il·lustració 56 Mensaje de Arduino indicando Modo Automático*

En este menú, se dispone de 7 modos de uso distintos que podremos aplicar en las situaciones creadas para su representación. Cada modo enviará consignas diferentes al motor para bien ponerlo en velocidad 1, 2 o 3 dependiendo de los requerimientos de cada situación o bien en avance o retroceso según se deba desplazar el objeto a controlar por el variador en una dirección u otra.

En este modo, si el variador está llevando a cabo una acción, no se podrá comenzar otra. Para ello se han creado unas variables de tipo entero en las que se guardara el estado en el que se haya la acción.

```

int ESTADOTELONUP = LOW; // Nos indica si el telon esta subiendo
int ESTADOTELONDOWN = LOW; // No indica si el telon esta bajando
int ESTADOPLATUP = LOW; // Indica que plataforma esta subiendo
int ESTADOPLATDOWN = LOW; // Indica que plataforma esta bajando
int ESTADOGIRA = LOW; // Indica que plataforma giratoria esta girando en sentido horario
int ESTADOGIRAANTI = LOW; // Indica que plataforma giratoria esta girando en sentido antihorario
int ESTADOGIRA5 = LOW; // Indica que plataforma giratoria esta girando hasta que cuente 5
int CONTADORGIRO = 0; // Cuenta el nº de vueltas que lleva la plataforma giratoria
  
```

*Il·lustració 57 Variables encargadas de guardar el estado de las diferentes funciones*

Si se está ejecutando, por ejemplo, el modo "SUBIR PLATAFORMA", la variable ESTADOPLATUP cambiará de LOW a HIGH, y cuando esta acción finalice, volverá a tener el valor LOW.

Estas variables nos aseguran que, en cualquier caso, no se activará una entrada digital del variador que produzca un fallo en el sistema y, en consecuencia, derive en un accidente grave o incluso dejar inservibles componentes del sistema.

Todos los modos se han conformado en funciones tipo void para simplificar código.

#### 4.2.1 Modo subir/bajar telón

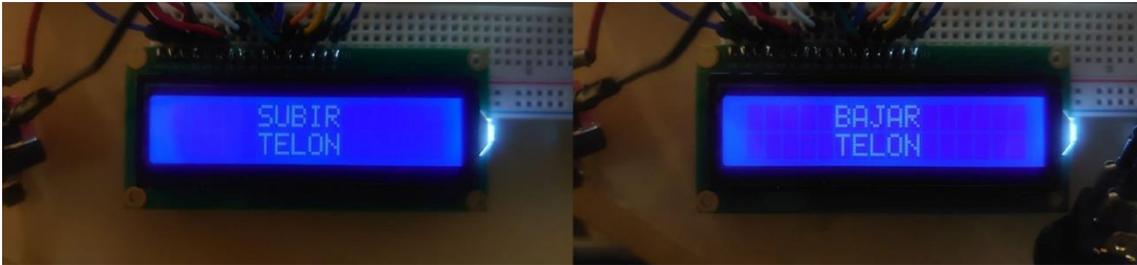


Ilustración 58 Mensaje por pantalla de LCD indicando modo SUBIR TELON y modo BAJAR TELON

En este modo, se pretenderá controlar un telón de escenario que tendrá un movimiento de forma vertical y su movimiento será de arriba abajo y viceversa. Contará con un sensor final de carrera en la parte alta del telón para detener el telón cuando llegue arriba y otro final de carrera en la parte baja del telón para detenerlo a su llegada al punto más bajo. El estilo de apertura de este telón se denomina de “Guillotina” o la “Veneciana”.



Ilustración 59 Telón con cierre de tipo Guillotina (Fuente: Alcázar Textiles)

El código que es capaz de llevar a cabo esta acción se ha creado, como todos los códigos que se utilizan a lo largo del software, con una sentencia condicionante.

Para subir el telón, las condiciones de entrada en la sentencia deben ser:

- Pulsador de encoder de modo automático en LOW, cuando se pulsa, manda un flanco negativo.
- Posición del encoder ha de ser = 2.

- El sensor final de carrera de abajo deberá estar en condición HIGH, indicando que el telón esta abajo.
- El sensor final de carrera de arriba deberá mandar la señal de LOW, es decir 0V, señalando que el telón no se encuentra recogido.
- Todos los estados que indican si cualquiera de los diferentes modos está trabajando, deberán estar en LOW, indicando que no se está ejecutando ninguna acción.

La sentencia que deberá llevar a cabo será la siguiente:

Nº de relé	Estado
1	HIGH (Motor en marcha)
2	HIGH (Sentido de giro, retroceso)
4	HIGH (Velocidad de giro 2)

Tabla 9 Sentencia del modo SUBIR TELON

La condición de desactivación será que el telón haya llegado arriba, pulsando el final de carrera instalado en la parte alta, dando un estado de HIGH. A su vez el sensor final de carrera de abajo deberá dar la señal de LOW. Todos los relés activados anteriormente pasarán a tener un valor LOW.

```
void subirTelon(){ // Funcion que nos activa y desactiva salidas para poder subir el telón

  if ((digitalRead(PULSADOR2)) == LOW && POSICION2 == 1 && (digitalRead(FINALCARRERADOWN)) == HIGH && (digitalRead(FINALCARRERAUP)) == LOW
  && ESTADOTELONDOWN == LOW && ESTADOPLATUP == LOW && ESTADOPLATDOWN == LOW && ESTADOGIRA == LOW && ESTADOGIRAANTI == LOW && ESTADOGIRA5 == LOW){
    digitalWrite(RELE1, HIGH);
    digitalWrite(RELE2, HIGH);
    digitalWrite(RELE4, HIGH);
    ESTADOTELONUP = HIGH;
  }
  else if ((digitalRead(FINALCARRERAUP)) == HIGH && (digitalRead(FINALCARRERADOWN)) == LOW && ESTADOTELONUP == HIGH){ // HAY QUE PONER 2 FINALES :
    digitalWrite(RELE1, LOW);
    digitalWrite(RELE2, LOW);
    digitalWrite(RELE4, LOW);
    ESTADOTELONUP = LOW;
  }
  else if ((digitalRead(FINALCARRERAUP)) == HIGH && (digitalRead(FINALCARRERADOWN)) == HIGH){
    digitalWrite(RELE1, LOW);
  }
}
```

Ilustración 60 Código de función subirTelon, ejecutora de la acción de subir el telón.

Para bajar el telón, las condiciones de entrada en la sentencia deben ser:

- Pulsador de encoder de modo automático en LOW, cuando se pulsa, manda un flanco negativo.
- Posición del encoder ha de ser = 3.
- El sensor final de carrera de arriba deberá estar en condición HIGH, indicando que el telón está recogido arriba.

- El sensor final de carrera de abajo deberá mandar la señal de LOW, es decir 0V, señalando que el telón no se encuentra extendido.
- Todos los estados que indican si cualquiera de los diferentes modos está trabajando, deberán estar en LOW, indicando que no se está ejecutando ninguna acción.

La sentencia que deberá llevar a cabo será la siguiente:

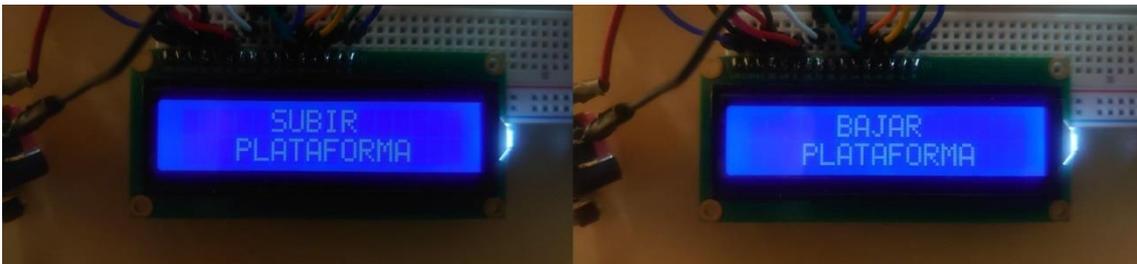
Nº de relé	Estado
1	HIGH (Motor en marcha)
4	HIGH (Velocidad de giro 2)

*Tabla 10 Sentencia del modo BAJAR TELON*

Las condiciones de desactivación que se deberán dar son; final de carrera de abajo en estado HIGH (5V) y final de carrera de arriba en LOW (0V). Los relés activados para llevar a cabo la acción, pasarán a 0V.

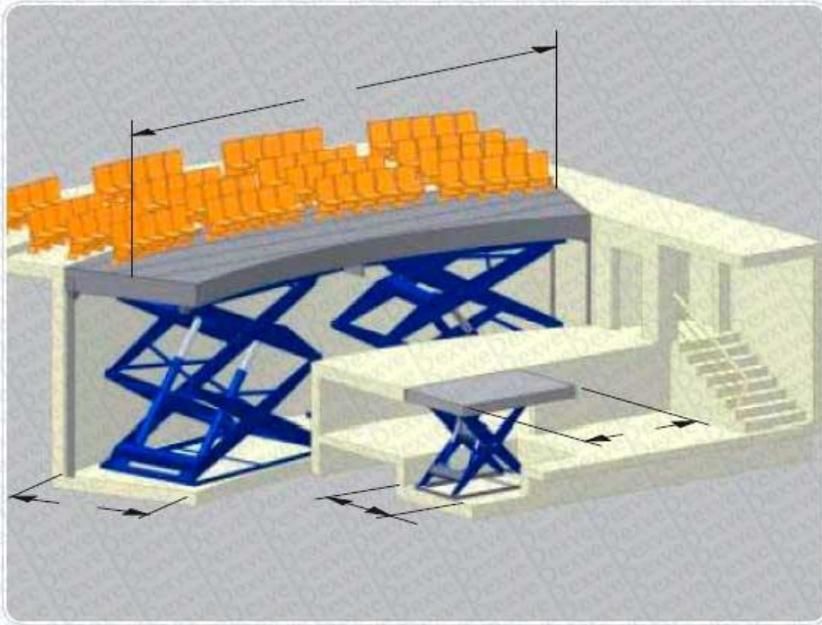
Además, estos dos sistemas, disponen de un sistema de seguridad que en el caso de que ambos finales de carrera se encuentren en HIGH, el motor se parará.

#### 4.2.2 Modo subir/bajar plataforma



*Ilustración 61 Mensaje de Arduino de modo SUBIR PLATAFORMA Y BAJAR PLATAFORMA*

La función de este modo es elevar y descender una plataforma controlada a través de un sensor ultrasónico, que nos determinará a que altura se encuentra la plataforma. Una vez haya llegado a la altura configurada por software, se detendrá. Para volver a la posición inicial, descenderá y pulsará un sensor final de carrera, que indicará que la plataforma ya se encuentra recogida. En el mercado actual se utiliza el sistema SpiraLift, el sistema de elevación más compacto y rígido adecuado para escenarios. En este caso se utilizará una plataforma de tijeras.



*Il·lustració 62 Plataforma utilitzada en teatre per elevar asientos (Fuente: DEXEV)*

A través de una sentencia condicionante se crea el código que dará funcionalidad a este modo.

Para elevar la plataforma, las condiciones de entrada en la sentencia deben ser:

- Pulsador de encoder de modo automático en LOW, cuando se pulsa, manda un flanco negativo.
- Posición del encoder ha de ser = 3.
- El sensor final de carrera colocado en el plegado de la plataforma, deberá estar en HIGH, 5V.
- Todos los estados que indican si cualquiera de los diferentes modos está trabajando, deberán estar en LOW, indicando que no se está ejecutando ninguna acción.

La sentencia que ejecutará será:

Nº de relé	Estado
1	HIGH (Motor en marcha)
2	HIGH (Sentido de giro, retroceso)
3	HIGH (Velocidad del motor: 1)
5	HIGH (Rampa de aceleración: 2)

*Tabla 11 Sentencia determinada para subir Plataforma*

La condición que desactivará los relés, será la altura a la que se encuentre la plataforma. Esta altura se ha programado a 11 cm. Es decir, cuando la altura se mayor o igual que 11, el motor se detendrá. Otro condicionante es el final de carrera de la plataforma, este deberá estar en LOW, señalando que la plataforma se encuentra elevada.

```
void subirPlataforma() { // Funcion que activa y desactiva las salidas para elevar la plataforma

    if (POSICION2 == 3 && digitalRead(FINALCARRERAPLAT) == HIGH && digitalRead(PULSADOR2) == LOW &&
        ESTADOTOLONUP == LOW && ESTADOTOLONDOWN == LOW && ESTADOPLATDOWN == LOW && ESTADOGIRA == LOW && ESTADOGIRAANTI == LOW && ESTADOGIRA5 == LOW) {
        digitalWrite(RELE1, HIGH);
        digitalWrite(RELE2, HIGH);
        digitalWrite(RELE3, HIGH);
        digitalWrite(RELE5, HIGH);
        ESTADOPLATUP = HIGH;
    }
    else if (DISTANCIA >= 11 && (digitalRead(FINALCARRERAPLAT) == LOW) && ESTADOPLATUP == HIGH) { // Aqui determinamos la altura máxima a la que c
        digitalWrite(RELE1, LOW);
        digitalWrite(RELE2, LOW);
        digitalWrite(RELE3, LOW);
        digitalWrite(RELE5, LOW);
        ESTADOPLATUP = LOW;
    }
}
```

*Ilustración 63 Código de función subirPlataforma, ejecutora de la acción de subir plataforma.*

Para bajar la plataforma, las condiciones de entrada en la sentencia deben ser:

- Pulsador de encoder de modo automático en LOW, cuando se pulsa, manda un flanco negativo.
- Posición del encoder ha de ser = 4.
- El sensor final de carrera colocado en el plegado de la plataforma, deberá estar en LOW, 0V.
- Todos los estados que indican si cualquiera de los diferentes modos está trabajando, deberán estar en LOW, indicando que no se está ejecutando ninguna acción.

Nº de relé	Estado
1	HIGH (Motor en marcha)
3	HIGH (Velocidad del motor: 1)
5	HIGH (Rampa de aceleración: 2)

*Tabla 12 Sentencia de la función bajarPlataforma*

Las condiciones de desactivación de todos los relés es el cambio de estado del final de carrera instalado en la plataforma de LOW a HIGH, esto significa que la plataforma ya está en la posición inicial, plegada.

El sensor ultrasónico tiene un rango de 40cm, en caso de necesidad de más alcance, se deberá cambiar la relación especificada por el fabricante.

#### 4.2.3 Giro en sentido horario/antihorario de base giratoria

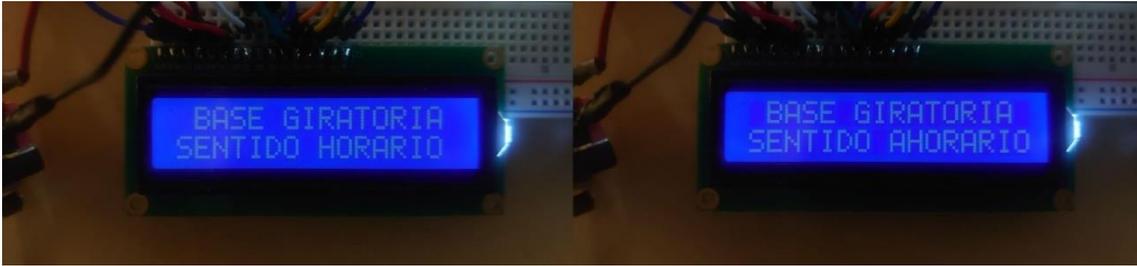


Ilustración 64 Mensaje de Arduino indicando BASE EN SENTIDO GIRATORIO Y BASE EN SENTIDO ANTIHORARIO

En este modo podremos efectuar giros en sentido horario o antihorario. La base giratoria tendrá 2 pequeñas pestañas con las que activará el sensor final de carrera instalado para detectar el desplazamiento giratorio. Se ha diseñado para que la base, de media vuelta, en vez de una vuelta completo, para poder hacer cambios de escenario simplemente girando la base.



Ilustración 65 Base giratoria para escenarios (Fuente: Pascualin Estructuras)

A través de una sentencia condicionante se crea el código que dará funcionalidad a este modo.

Para hacer rotar la base giratoria en sentido horario, las condiciones de entrada en la sentencia deben ser:

- Pulsador de encoder de modo automàtic en LOW, cuando se pulsa, manda un flanco negativo.
- Posición del encoder ha de ser = 5.
- El sensor final de carrera, que se activará a través de un pequeño resorte instalado en la base giratoria, deberá tener el estado de HIGH.
- Todos los estados que indican si cualquiera de los diferentes modos está trabajando, deberán estar en LOW, indicando que no se está ejecutando ninguna acción.

La sentencia que ejecutará será:

Nº de relé	Estado
1	HIGH (Motor en marcha)
3	HIGH (Velocidad del motor: 1)
4	HIGH (Velocidad del motor: 2)
5	HIGH (Rampa de aceleración: 2)

Tabla 13 Sentencia determinada para *giraHorario*

Como se puede observar, se activará velocidad 1 y 2 a la vez, esto, cabe recordar, que activa velocidad 3.

Se ha tenido en cuenta que desde que se pone a girar la plataforma hasta que deja de tener contacto con el final de carrera, tiene un margen de 5 seg. Es por esto que se utiliza la función `millis` para determinar los segundos transcurridos sin necesidad de utilizar un `delay` que ralentizaría el software.

La condición que desactivará los relés, será la pulsación del mismo final de carrera. Por esto mismo es importante los 5 segundos de margen, sino se activaría y desactivaría de inmediato.

```
void giraHorario() { // Activa la plataforma y la hace girar hasta que detecta el final de carrera. Añadir un pequeño resorte en la plataforma para
if (POSICION2 == 5 && digitalRead(FINALCARRERAGIRA) == HIGH && digitalRead(PULSADOR2) == LOW &&
ESTADOTELONUP == LOW && ESTADOTELONDOWN == LOW && ESTADOPLATUP == LOW && ESTADOPLATDOWN == LOW && ESTADOGIRAANTI == LOW && ESTADOGIRA5 == LOW) {
digitalWrite(RELE1, HIGH);
digitalWrite(RELE3, HIGH);
digitalWrite(RELE4, HIGH);
digitalWrite(RELE5, HIGH);
ESTADOGIRA = HIGH;
TiempoAhora2 = millis();
}
else if ((millis() >= TiempoAhora2 + periodo2) && (digitalRead(FINALCARRERAGIRA) == HIGH) && ESTADOGIRA == HIGH) { // Tiempo de seguridad para q
digitalWrite(RELE1, LOW);
digitalWrite(RELE3, LOW);
digitalWrite(RELE4, LOW);
digitalWrite(RELE5, LOW);
ESTADOGIRA = LOW;
}
}
```

Ilustración 66 Código que determina las acciones que llevará a cabo la función *giraHorario*

Para iniciar el giro en sentido antihorario, las condiciones de entrada en la sentencia deben ser:

- Pulsador de encoder de modo automático en LOW, cuando se pulsa, manda un flanco negativo.
- Posición del encoder ha de ser = 6.
- El sensor final de carrera, que se activará a través de un pequeño resorte instalado en la base giratoria, deberá tener el estado de HIGH.
- Todos los estados que indican si cualquiera de los diferentes modos está trabajando, deberán estar en LOW, indicando que no se está ejecutando ninguna acción.

Nº de relé	Estado
1:	HIGH (Motor en marcha)
2	HIGH (Sentido de giro, retroceso)
3	HIGH (Velocidad del motor: 1)
4	HIGH (Velocidad del motor: 2)
5	HIGH (Rampa de aceleración: 2)

*Tabla 14 Sentencia de la función giraAntihorario*

La condición que desactivará los relés, será la pulsación del mismo final de carrera una vez transcurrido los 5 segundos.

#### 4.2.4 Contador de 5 vueltas en base giratoria



*Ilustración 67 Mensaje de Arduino indicando modo BASE GIRATORIA 5 GIROS*

La función de este modo es dar 5 vueltas consecutivas a la base giratoria, y una vez pulsado 5 veces el sensor final de carrera instalado, se detendrá. Contaremos con una pestaña en la base que actuará como activador del final de carrera.

La base giratoria a utilizar es la misma que en el caso anterior, solo que tendrá una pestaña de activación menos dado que queremos que realice una vuelta completa en vez de media vuelta.

Se ha implementado unas líneas de código que actuarán como contador de vueltas que ha dado la base giratoria. Lo iniciamos en 0, pero Arduino, al detectar que esta pulsado suma 1 posición, es por esto que se denominará la vuelta 6 como final.

Para hacer rotar la base giratoria e iniciar el contador de vueltas, las condiciones de entrada en la sentencia deben ser:

- Pulsador de encoder de modo automático en LOW, cuando se pulsa, manda un flanco negativo.
- Posición del encoder ha de ser = 7.
- El sensor final de carrera, que se activará a través de un pequeño resorte instalado en la base giratoria, deberá tener el estado de HIGH, indicando que se encuentra en la posición inicial.
- Todos los estados que indican si cualquiera de los diferentes modos está trabajando, deberán estar en LOW, indicando que no se está ejecutando ninguna acción.

La sentencia que ejecutará será:

Nº de relé	Estado
1	HIGH (Motor en marcha)
3	HIGH (Velocidad del motor: 1)
4	HIGH (Velocidad del motor: 2)
5	HIGH (Rampa de aceleración: 2)

Tabla 15 Sentencia de la función giraCinco

Para que el contador funcione de forma óptima, se ha incluido un retraso en el conteo de vueltas ya que, al ir tan rapido el loop (procesador de 16 MHz → cada  $6,25 \times 10^{-8}$  s hace la lectura completa del código), con una simple pulsación ya habría contado las 5 vueltas.

```

void giraCinco() { // Activa la plataforma y la hace girar 5 veces en sentido horario.

  if (POSICION2 == 7 && digitalRead(FINALCARRERAGIRA) == HIGH && digitalRead(PULSADOR2) == LOW &&
  ESTADOTELONUP == LOW && ESTADOTELONDOWN == LOW && ESTADOPLATUP == LOW && ESTADOPLATDOWN == LOW && ESTADOGIRAANTI == LOW && ESTADOGIRA == LOW) {
    digitalWrite(RELE1, HIGH);
    digitalWrite(RELE3, HIGH);
    digitalWrite(RELE4, HIGH);
    digitalWrite(RELE5, HIGH);
    ESTADOGIRA5 = HIGH;
  }
  else if ((millis() >= TiempoAhora4 + periodo4) && (digitalRead(FINALCARRERAGIRA) == HIGH) && ESTADOGIRA5 == HIGH) { // Tiempo de seguridad para
    CONTADORGIRO++;
    TiempoAhora4 = millis();
  }
  else if (CONTADORGIRO == 6 && (digitalRead(FINALCARRERAGIRA) == HIGH) && ESTADOGIRA5 == HIGH) {
    digitalWrite(RELE1, LOW);
    digitalWrite(RELE3, LOW);
    digitalWrite(RELE4, LOW);
    digitalWrite(RELE5, LOW);
    ESTADOGIRA5 = LOW;
    CONTADORGIRO = 0;
  }
}
}

```

*Il·lustració 68 Codi que determina les accions que llevarà a cabo la funció giraCinco*

Las condiciones que desactivarán los relés serán que el contador haya contado 6 vueltas (empieza contando desde 1), es decir, la variable CONTADORGIRO deberá ser igual a 6. Y por último que se haya pulsado el final de carrera instalado en la base.

### 4.3 Cambio de valores para otros usos

En este apartado se especificarán los valores de código que son variables dependiendo de la situación que se presente.

#### 4.3.1 Altura de detención de plataforma

Se podrá variar la altura a la que deseamos que se detenga la plataforma elevadora cambiando el valor de la variable DISTANCIA. En este caso la altura a la que detiene son 11cm y el sensor tiene capacidad de lectura con precisión hasta los 40cm.

En la línea 332 del código IDE de Arudino se cambiará el nº 11 por el valor deseado:

```

else if (DISTANCIA >= 11 && (
  digitalWrite(RELE1, LOW);
  digitalWrite(RELE2, LOW);
  digitalWrite(RELE3, LOW);
  digitalWrite(RELE5, LOW);
  ESTADOPLATUP = LOW;
)

```

*Il·lustració 69 Variable DISTANCIA que detendrá la plataforma a 11cm del suelo*

Por ejemplo, se desea elevar la plataforma hasta los 28 cm. Debemos escribir en el condicionante de desactivación de los relés de la función subirPlataforma (línea de código 332); DISTANCIA >= 28.

De igual manera, cuando se cambie de sensor ultrasónico por uno de mayor o menor rango, se deberá cambiar la relación que indica el fabricante entre tiempo y distancia, ya que, como visto en anteriores capítulos, el sensor ultrasónico ha de transformar el tiempo que tarda en recibir la señal de vuelta en centímetros.

```

}
DURACION = pulseIn (ECO, HIGH);
DISTANCIA = DURACION / 58.2;

```

*Ilustración 70 Relación de transformación especificada por el fabricante para sensor ultrasónico*

En la línea 110 del código fuente, deberemos cambiar el valor de 58,2, por el nuevo valor que indique el fabricante del nuevo sensor ultrasónico.

#### 4.3.2 Tiempo de espera final de carrera base giratoria

A lo largo del código se aplican unos tiempos de espera para realizar las acciones posteriores por seguridad, para que no se desactiven los relés de forma involuntaria. Esto lo conseguimos con la función millis y las variables que ayudan a conocer el tiempo que ha pasado.

Si se dan las condiciones para que el tiempo necesario entre que arranca el motor y se deja del pulsar el final de carrera, sea menor o mayor, se deberá cambiar:

Línea de código	Función	Variable	Tiempo actual de espera
11	giraHorario	periodo2	3000 ms
13	giraAntihorario	periodo3	3000 ms
15	giraCinco	periodo4	5000 ms

Por ejemplo, en el caso de la función giraCinco, se ha determinado un tiempo de 5 segundos para que la base pueda activar y desactivar el final de carrera sin que el contador llegue a 6 de forma inmediata. Si en nuestro caso ponemos una velocidad superior, se podría dar el caso que, al pasar por el final de carrera, Arduino no detecte la señal de 5V porque está esperando a que pasen los 5 segundos.

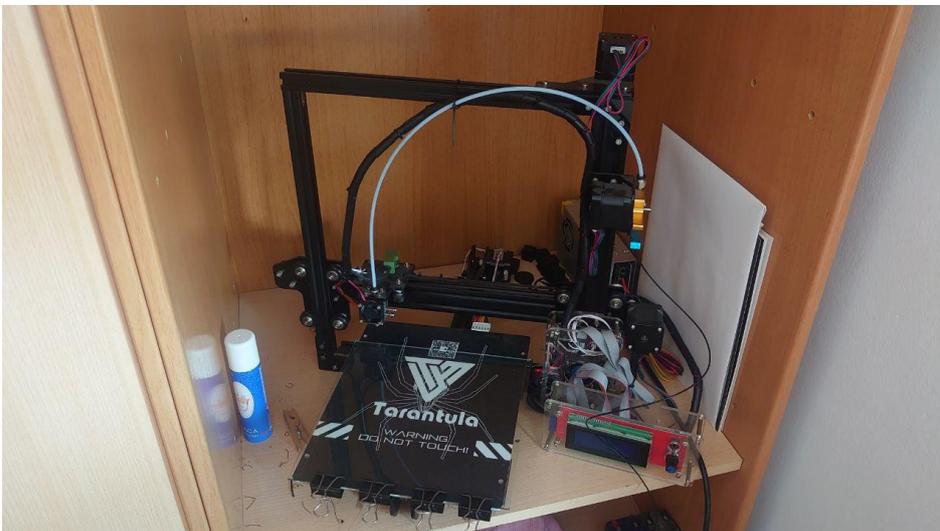
```
int periodo4 = 5000;
```

*Ilustración 71 Variable de tiempo para función giraCinco*

## 5 IMPRESIÓ 3D

Con la intenció de poder recrear al màxim les situacions per les que ha sigut pensat el projecte i per tal de poder demostrar a les estudiantes del grau el funcionament de les diferents components, se han creat diferents estructures en 3D amb una impressora 3D.

Estes estructures han sigut dissenyades primerament en SolidWorks<sup>12</sup> per posteriorment convertir-les en un arxiu StereolithographyMesh. Estos arxius debem tractar-los amb el programa CURA 3D, que crearà la fila de codificació per l'eix X, Y i Z de la impressora.



*Il·lustració 72 Impressora 3D amb la que es treballaran les estructures*

Per a la impressió de les diferents estructures s'utilitzarà un tipus de material que prové del blat de moro, el PLA. A diferència del ABS (més consistent i resistent), necessita de menys energia per derretir-lo i conleva menys problemes tècnics.

---

<sup>12</sup> Software CAD per a modelat mecànic en 2D i 3D



*Il·lustració 73 PLA negre, material empleado para la creación de las estructuras*

Como dato, se han utilizado 375g de PLA y se ha imprimido todo en 37 h.

### 5.1 Plataforma giratoria

Con la plataforma giratoria se consigue recrear el escenario requerido. Para su diseño contaremos con una base fija y una tapa móvil. Para poder hacer la base móvil, se hará uso de unos rodamientos ABEC-6.

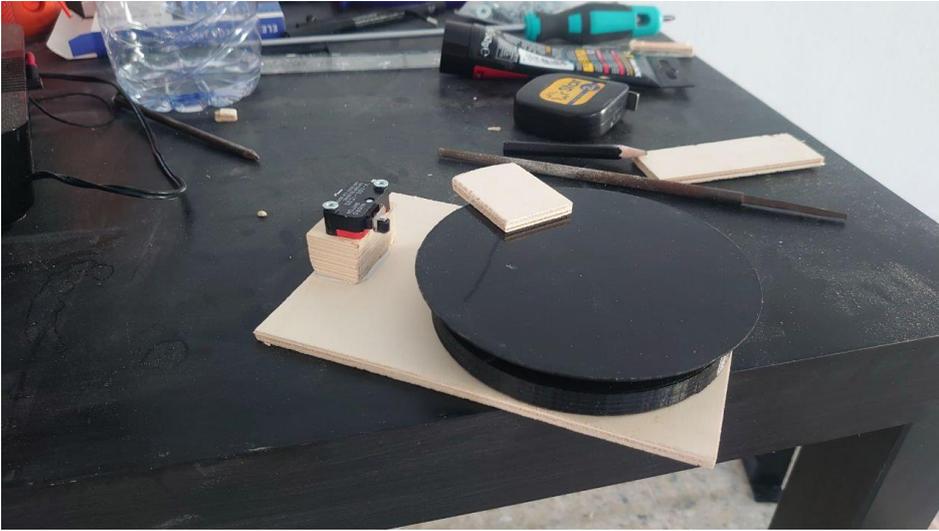


*Il·lustració 74 Plataforma giratoria montada.*



*Il·lustració 75 Plataforma giratoria abierta donde se observa el rodamiento ABEC-6*

Y, por último, se mecaniza y se cablea junto al final de carrera para poder mandar las señales digitales a Arduino.



*Ilustración 76 Plataforma giratoria mecanizada*

## 5.2 Plataforma elevadora

El modelo de plataforma elevadora más utilizado en la industria de las artes escénicas, como ya se vio en el primer capítulo, es el Spiralift. Para la recreación de una plataforma elevadora se ha utilizado un sistema más antiguo, sistema de tijera.

Este consta de un vito y una rosca que conseguirán elevar la plataforma hasta máximo 14 cm. Es por este motivo que no se ha puesto más altura a reconocer a Arduino, pues se ha dejado un margen de seguridad.



*Ilustración 77 Diferentes posiciones de la plataforma consiguiendo diferentes alturas*

Finalmente, se añade el sensor ultrasónico que detectará la altura de la estructura y el sensor final de carrera para determinar que ya esta plegado.



*Ilustración 78 Plataforma elevadora mecanizada*

### 5.3 Telón de escenario

A modo ilustrativo, se ha diseñado un telón con plegado desde el punto más alto. Consiste de una guía por la que transcurrirá un clavo adaptado al cual, se le ha añadido peso para poder activar el sensor final de carrera del punto más bajo.



*Ilustración 79 Construcción de telón ficticio*



*Il·lustració 80 Estructura de telón con finales de carrera mecanizado*

Finalmente se conectan los finales de carrera, uno en la parte superior y otro en la parte inferior. Juntos determinarán la posición del telón (abierto o cerrado).

## 6 MONTAJE MECÁNICO

En este epígrafe se explicará en detalle cómo realizar la mecanización de la caja contenedora de todo el material necesario para el correcto funcionamiento de Arduino.

Se comenzará por la caja, en este caso es una caja de ABB de medidas 220x170x80 mm con tapa transparente para poder ver el contenido interior.



*Il·lustració 81 Caja donde se alojarán todos los componentes ABB de 220x170x80mm*

Inicialmente, se incorporará 18 hembras para conector de banana que se dispondrán sobre la caja de PVC en la parte alta para una mayor comodidad en el cableado externo. Para poder realizar los agujeros, utilizaremos un taladro, un metro, un lápiz o rotulador y una broca de 10 especial para madera.



*Ilustración 82 Taladro común y broca para madera de 10 mm*



*Ilustración 83 Banana hembra de 4 mm*

Primero se mide con un metro y se determina la ubicación de las bananas. Para este proyecto se han hecho 6 grupos de bananas:

- Primer grupo: Entrada y salida de la tensión de 24 V que proviene del convertidor de frecuencia. 6 bananas.
- Segundo grupo: Sensor ultrasónico proveniente de la plataforma elevadora. 4 bananas.

- Tercer grupo: Final de carrera proveniente del telón, la parte de arriba.
- Cuarto grupo: Final de carrera proveniente del telón, la parte de abajo.
- Quinto grupo: Final de carrera proveniente de la plataforma elevadora.
- Sexto grupo: Final de carrera proveniente de la plataforma giratoria.



*Ilustración 84 Mediciones con metro para determinar ubicación de las bananas hembra*

Una vez ubicado las bananas y marcadas con lápiz o rotulador, se agujereará cada orificio con la broca de 10 mm para madera. Una vez taladrado cada uno de los agujeros, se repasará con una lima redonda para retirar cualquier aspereza que haya podido quedar presente. Por último, se colocarán las bananas hembras y se deberá collar las tuercas que vienen incluidas en ellas mismas.



*Ilustración 85 Taladrando los agujeros pertinentes para las bananas hembra*



*Il·lustració 86 Limando asperezas post taladrado*



*Il·lustració 87 18 Bananas hembras instal·lades*

Una vez instaladas las bananas hembras en la caja, se procede a montar los relés necesarios para el paso de tensión del convertidor de frecuencia. Para esta operación se necesita una broca para madera de 3mm, tornillería de 3 x 25 mm (diámetro x longitud), tuercas de métrica 3 y pegamento especial para fijar tornillos CEYS.



*Il·lustració 88 Tornillos de 3x25mm y tuerca de métrica 3*



*Il·lustració 89 Fijador de tornillos anaeróbico CEYS*

Los propios relés traen consigo unos agujeros para poder fijarlos, se presenta sobre la caja, se marca con un rotulador y se pasa la broca de 3 mm para madera. A continuación, se introduce el tornillo y se ajusta la tuerca. Una vez fijado, se le suministra una pequeña cantidad de fijatornillos CEYS para que no se aflojen (en caso de necesidad de reparación es extraíble fácilmente).



*Il·lustració 90 Pareja de relés fijada a la caja por la parte posterior.*



*Il·lustració 91 Vista posterior de la caja con los 2 primeros relés instalados*



*Il·lustració 92 Relés completamente instalados*

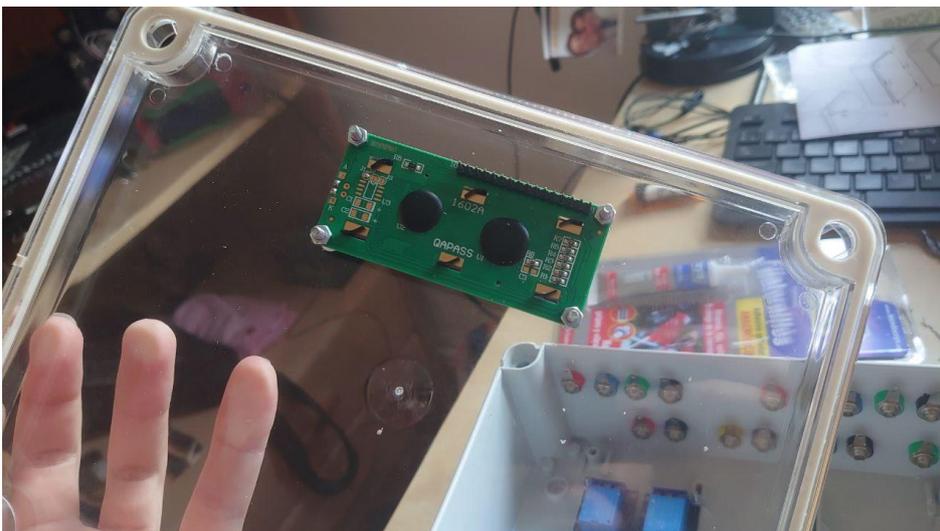
Para poder realizar las conexiones de una forma correcta, se ha dejado un espacio entre el relé ubicado en la parte superior y el relé inferior.

Posteriormente, se instala la pantalla LCD en el frontal de la caja para obtener una buena visualización de esta. Se utilizarán las mismas herramientas y componentes que en la instalación de los relés.



*Ilustración 93 Instalación de LCD en parte frontal*

Se deberá tener cuidado cuando se apriete la tuerca pues hace torsión la placa de circuitos de la LCD con la tapa transparente frontal.

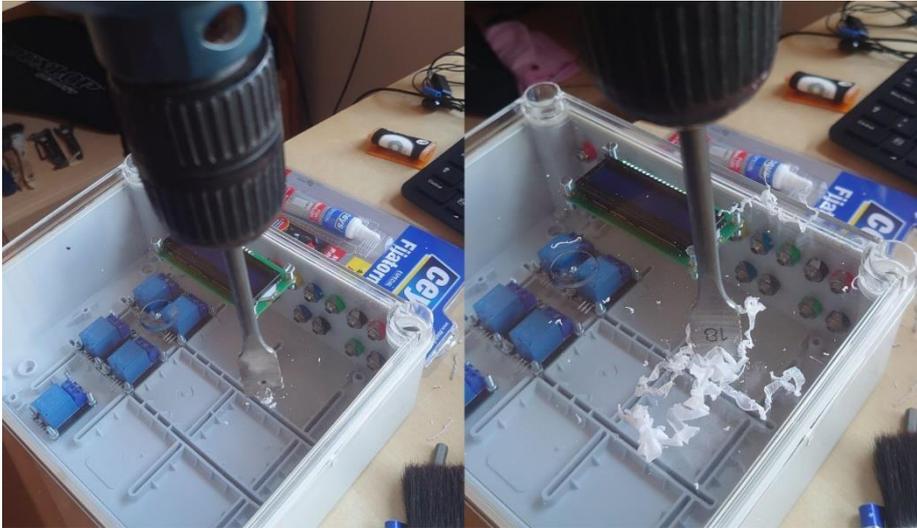


*Ilustración 94 LCD instalada con tornillería de métrica 3*

La pantalla LCD quedará así protegida de cualquier inclemencia que se nos pueda presentar o cualquier golpe o accidente.

En el mismo frontal de la caja, se instala los interruptores que darán paso al menú manual o automático. Estos interruptores tienen un diámetro interno de 18 mm y se necesita una broca especial para madera de 18 mm junto al taladro.

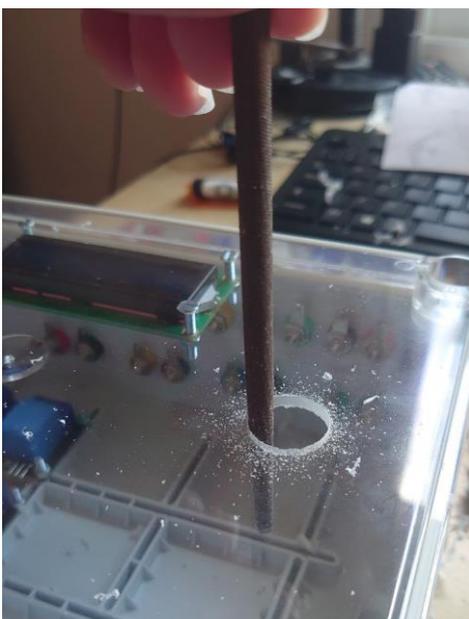
Como en los casos anteriores, tomaremos medidas con el metro para ubicar los interruptores en el frontal de forma que los componentes internos puedan ser vistos.



*Ilustración 95 Realización del agujero para el interruptor con broca de 18mm de madera*

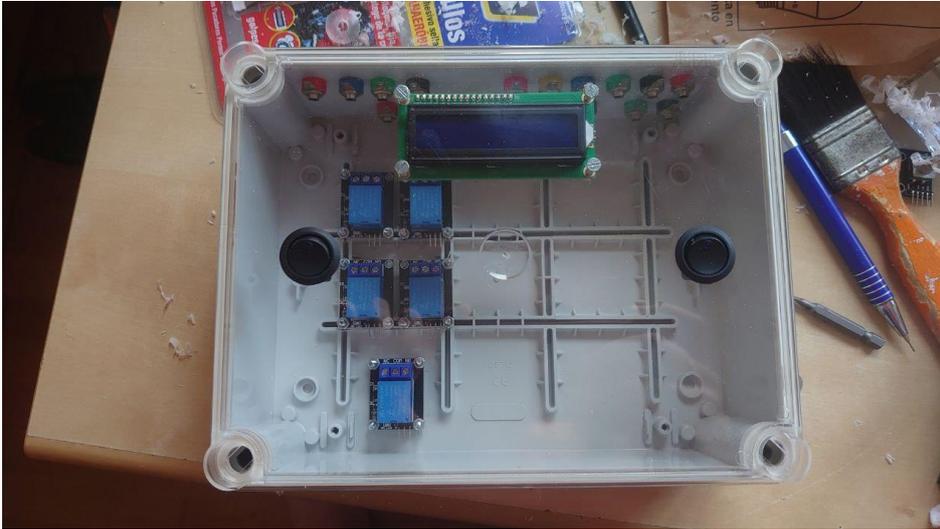
Una vez realizado el agujero pasaremos la lima para poder corregir las asperezas que se hayan podido producir en el proceso.

Este agujero se deberá realizar dos veces, tanto para el interruptor del modo manual como el del modo automático.



*Ilustración 96 Lijado general del contorno del agujero para el interruptor*

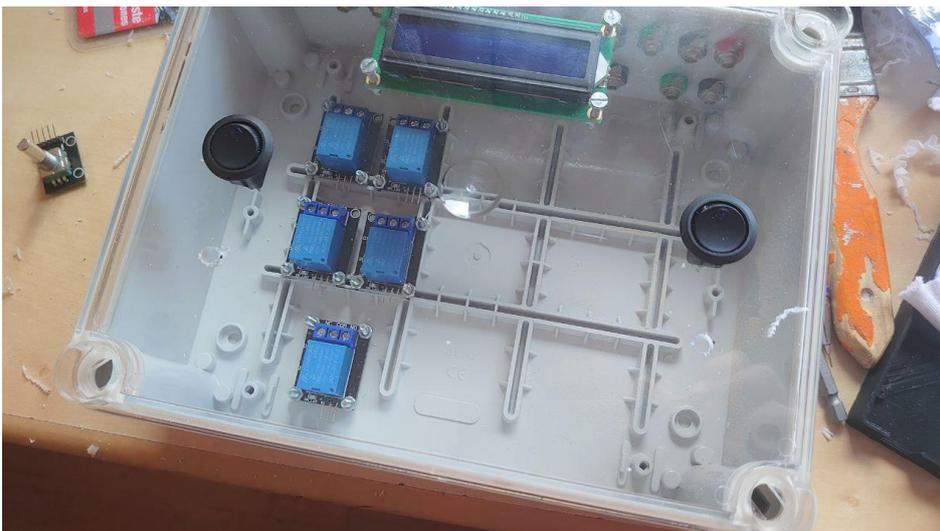
El interruptor trae unas pestañas de presión instaladas, de esta forma simplemente se colocará el interruptor en el agujero de 18 mm realizado y se ejerce presión hacia adentro para colocarlo.



*Ilustración 97 Interruptores de manual y automático instalados*

Posteriormente, se instala los encoders que permitirán navegar al usuario por los diferentes menús automáticos y manuales y activar o desactivar los modos disponibles.

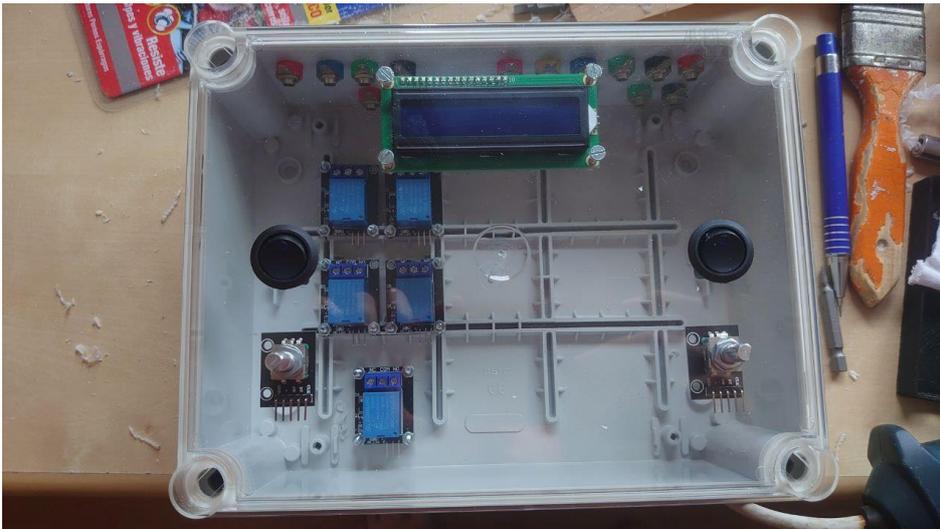
En este caso se precisa de una broca de 7 mm para madera. Como en todos los casos anteriores, se mide y se colocan los agujeros para una correcta disposición del encoder.



*Ilustración 98 Agujeros para colocación de encoders*

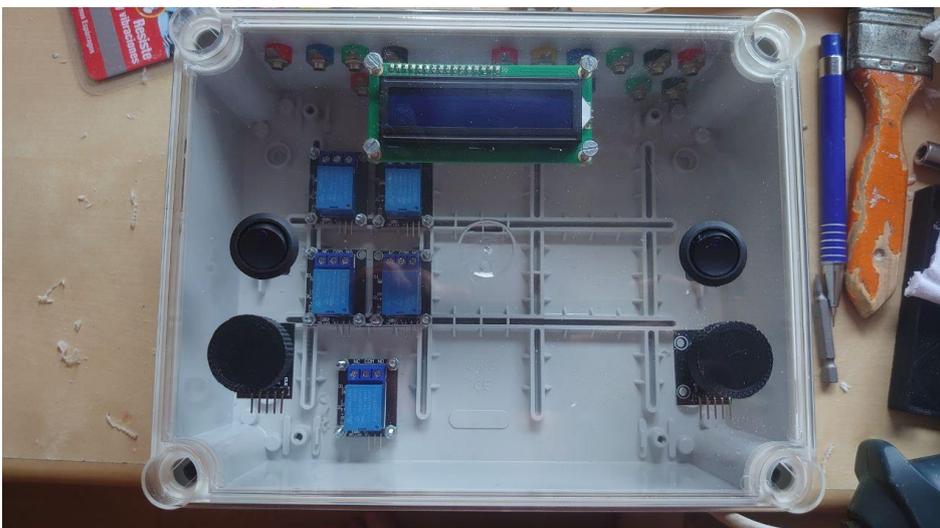
Se instalan en la parte inferior de los interruptores para una fácil comprensión del usuario de que encoder es el que corresponde a manual y cual el que corresponde al modo automático.

Los encoders adquiridos traen consigo mismo el visor hecho y la tuerca para poder realizar la instalación, simplemente se introduce en el agujero realizado y se le coloca la tuerca por la parte superior para sujetarlo con fuerza. Se le añadirá producto fijatornillos CEYS para una correcta sujeción.



*Ilustración 99 Encoders instalados y sujetos con tuerca propia*

Por último, se colocan los botones imprimidos en 3D para poder hacer uso de los encoders.

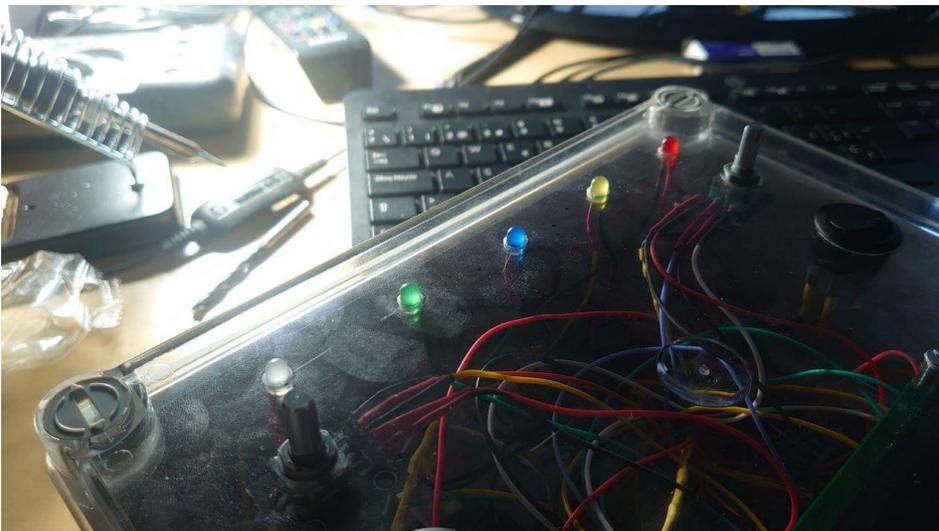


*Ilustración 100 Encoders con botón de impresión 3D colocado*

Ahora, realizaremos 5 agujeros con broca de 5 mm para madera donde alojaremos los LED indicadores del estado del variador de frecuencia.

Mediremos y marcaremos junto con el rotulador los lugares donde agujeraremos para instalarlos. Para poder sujetarlos, al no tener ningún sistema de rosca ni presión, se le añadirá una gotita de Loctite.

A parte, para poder regular el brillo de la LCD se le ha añadido un potenciómetro que regulará la intensidad de la retroiluminación de la pantalla.



*Ilustración 101 LEDs de diferentes colores instalados*

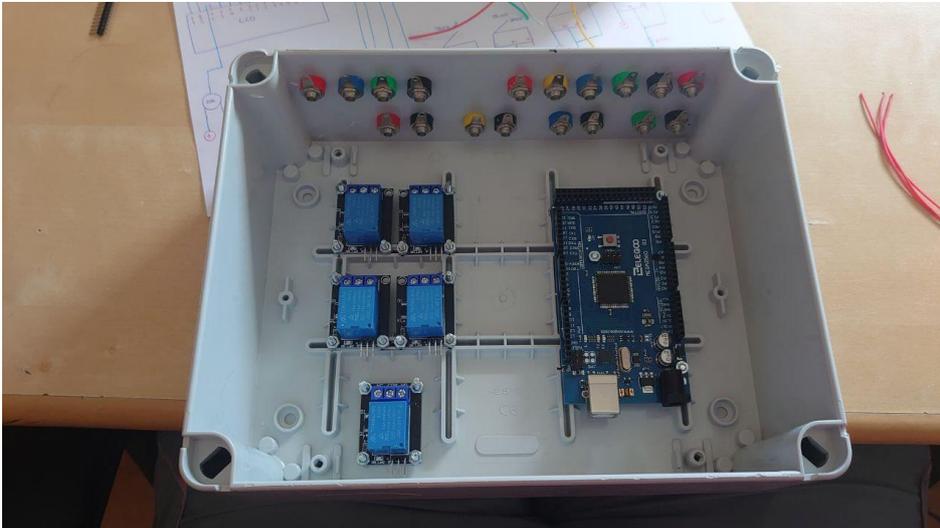


*Ilustración 102 Potenciómetro de 10 k para la retroiluminación de la pantalla*

El potenciòmetro, al no tener, al igual que el LED, ningún sistema de fijación, también se fijará añadiendo unas gotitas de Loctite con mucho cuidado de no pegar la torre giratoria, pues no podríamos regular la potencia.

Por último, como parte del mecanizado de la caja, se instala la placa base Arduino. Utilizaremos broca de 3 mm para madera y tornillería de métrica 3, como en casos anteriores.

La placa de Arduino trae los agujeros hechos de fábrica, pero solo se pueden utilizar, por seguridad, 3 de ellos. Se presenta la placa sobre la caja y se marca con rotulador los agujeros a realizar.



*Ilustración 103 Placa base Arduino instalada en caja ABB*



*Ilustración 104 Parte trasera de la caja, 3 tornillos para placa Arduino*

En este momento, se da por finalizado el mecanizado por completo de la caja ABB y ahora pasa a ser un conjunto de componentes que se juntarán para trabajar en un mismo espacio.

Por último, hacemos un último agujero con broca de 10 mm de madera para la fuente de alimentación de Arduino externa.



*Ilustración 105 Conexión para fuente de alimentación de Arduino*

## 7 MONTAJE ELÉCTRICO

Primero de todo se cableará la parte transparente de la caja, es decir, la pantalla LCD, interruptores, encoders y LEDs.

Se utilizará cable de 0,75 mm<sup>2</sup> de sección para todo el conjunto. Se tratarán tensiones de 5 V y amperajes de no más de 1 A que puede llegar a generar la fuente de alimentación de Arduino. Los cables serán de diferentes colores y se utilizarán el rojo y negro exclusivamente para positivos y negativos. El resto de colores corresponderá con entradas y salidas digitales.



*Ilustración 106 Cableado utilizado a lo largo de la instalación eléctrica, marca TUOFENG*

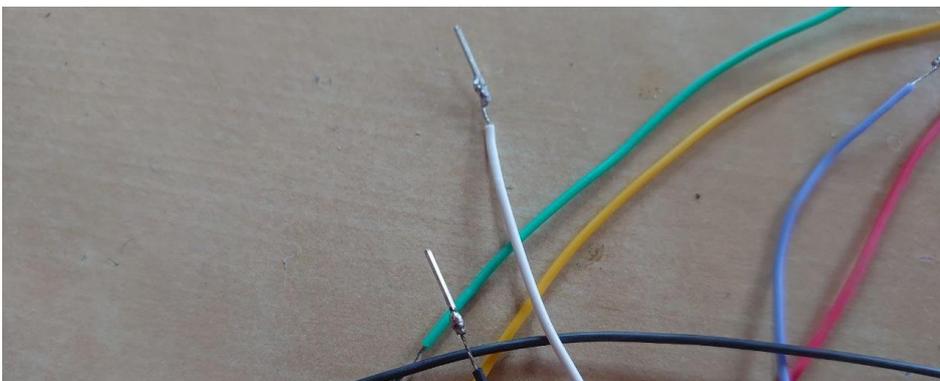
La longitud de los mismos será la misma en cualquier caso excepto en casos de necesidad, para facilitar las conexiones una vez se tenga que poner la tapa delantera. Longitud = 25 mm.

Para asegurarnos un correcto aislamiento, tubos termo retráctiles de 1 mm<sup>2</sup> y 1,5 mm<sup>2</sup> en los casos en los que la soldadura ha quedado muy gruesa.



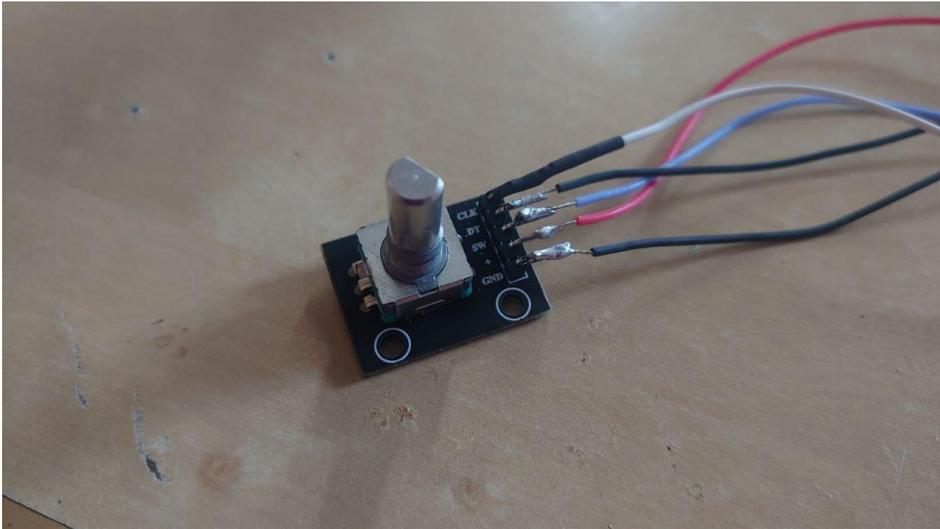
*Ilustración 107 Tubos termo retráctiles pre cortados de EVENTRONIC*

Se comenzará soldando los dos encoders. Estos tienen 2 cables que irán a positivo y negativo y los otros 3 van a la placa de Arduino, por lo tanto, se le suelda pines de conexión.

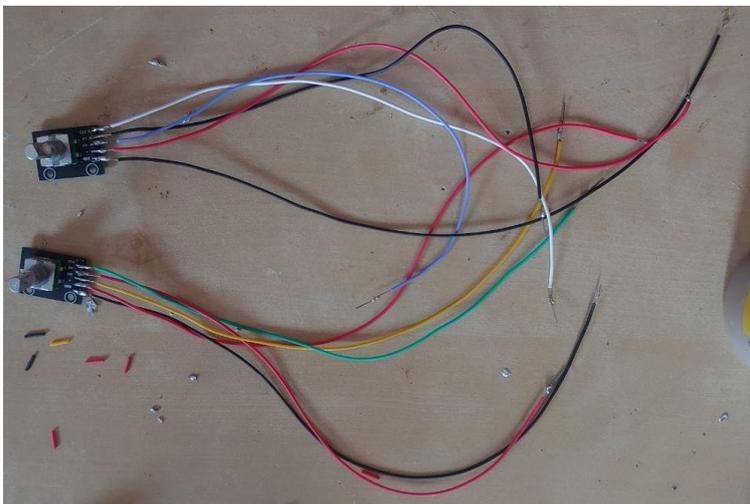


*Ilustración 108 Pines de conexión soldados en cables*

Después se sueldan todos los cables en sus respectivas patas.

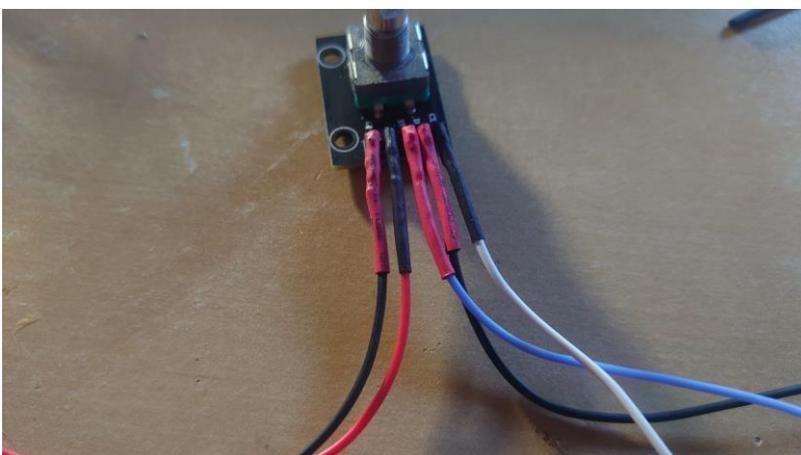


*Il·lustració 109 Encoder con cables soldados*



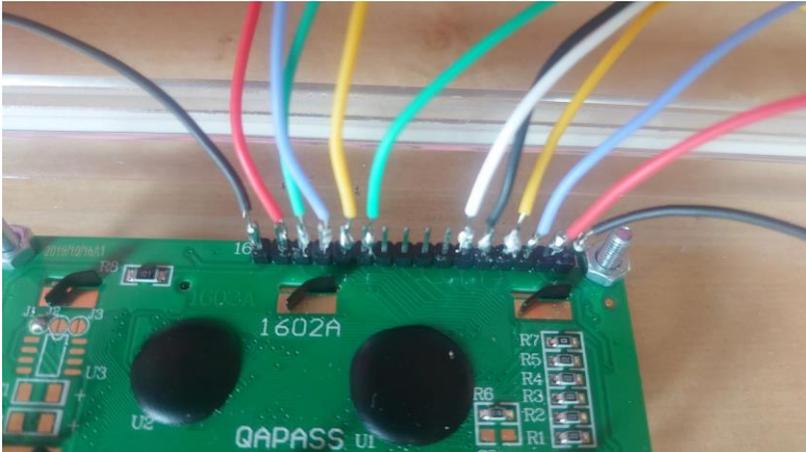
*Il·lustració 110 Encoder de modo automàtic y manual soldados*

Finalmente se añade un pedazo de tubo termo retráctil para aislar las conexiones.

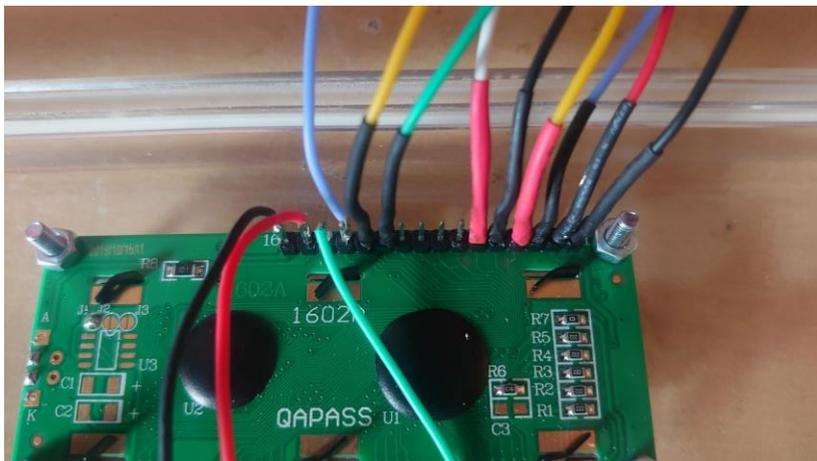


*Il·lustració 111 Encoder con termo retráctil instalado*

Ahora procedemos a realizar el mismo proceso con la pantalla LCD.



*Il·lustració 112 Pins de pantalla LCD soldados*



*Il·lustració 113 Pins de pantalla protegidos con tubo termo retráctil*

En el caso de la pantalla LCD, necesita de una resistencia de  $220\ \Omega$  para proteger la luz de retroiluminación de la propia pantalla. Para montarla de forma discreta y sin que pueda haber cualquier corto circuito, se cortará el cable por la mitad y se pela ambos extremos. Se acorta la distancia inicial de la resistencia y se estaña en ambos lados.



*Il·lustració 114 Estañado de resistència de 220 ohm*

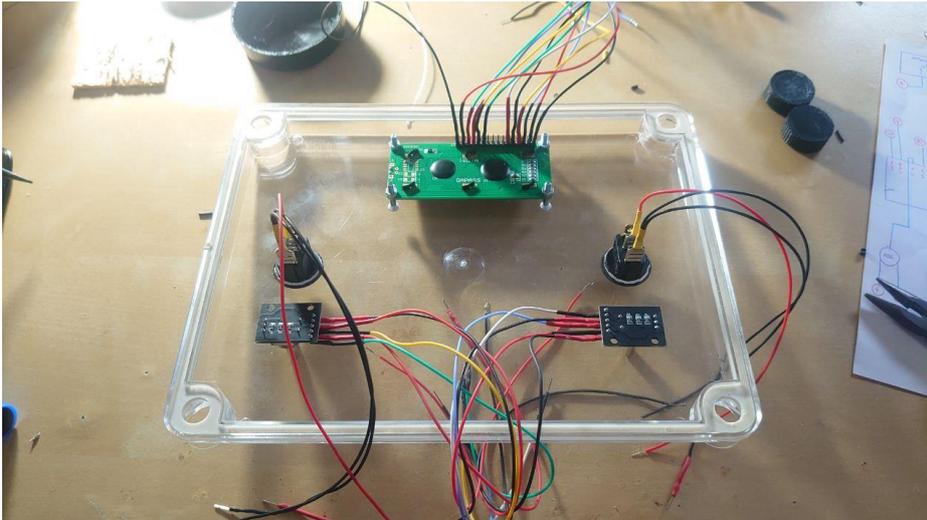
Finalmente se añade tubo termo retráctil para asegurarnos el correcto aislamiento.

Para finalizar la parte delantera de la caja de componentes, se cablea el interruptor que nos llevará al menú manual o automático. Estos precisan de una resistencia de 10 k $\Omega$  conectada al negativo para crear el efecto pull down. Se utilizarán conectores tipo FASTON para conectar el cableado y tubos termo retráctiles para protegerlo.



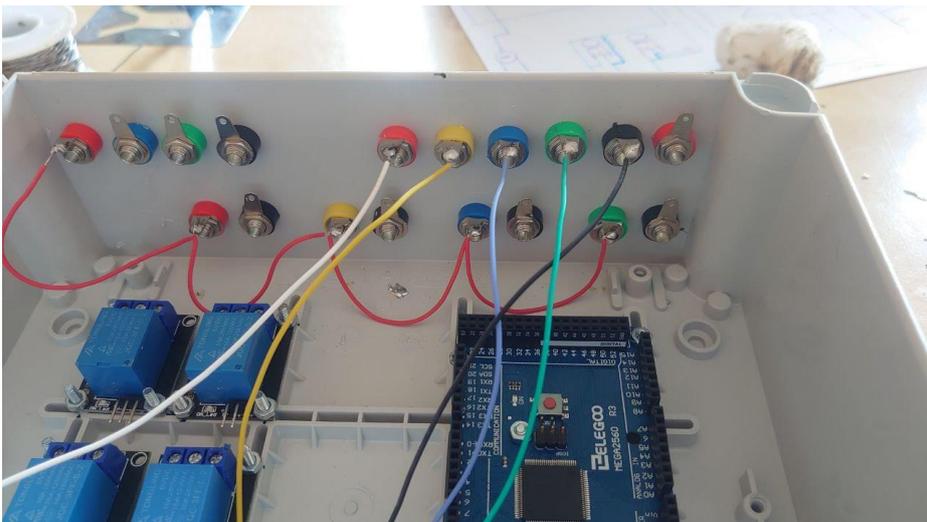
*Il·lustració 115 Interruptor conectado con sus respectivos conectores FASTON y tubo termo retráctil de aislamiento.*

Con todo esto, a falta de cablear los LED junto a los relés, ya tenemos finalizada la tapa delantera de la caja.



*Il·lustració 116 Parte frontal de la caja finalizada*

En este momento se pasa a la parte interior de la caja. Se comenzará cableando las bananas hembras instaladas.

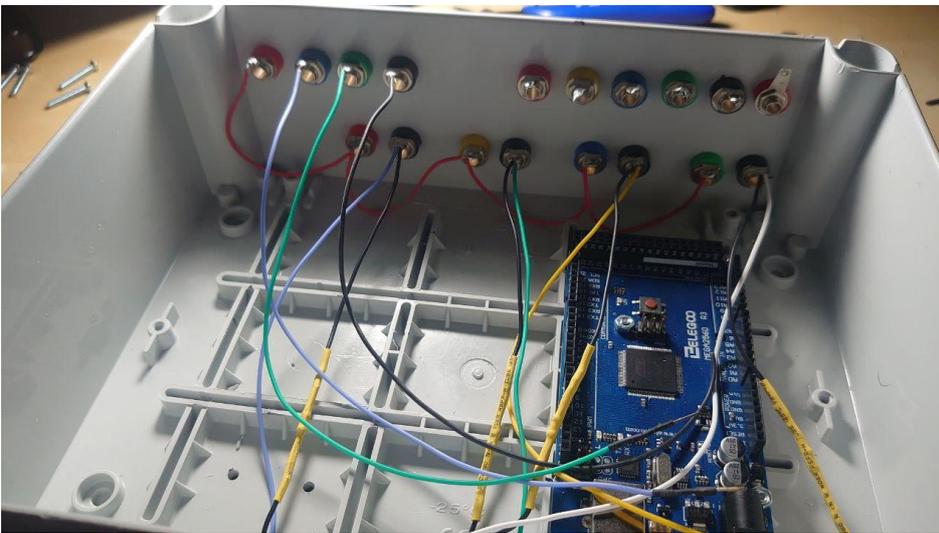


*Il·lustració 117 Cableado de bananas hembra*

Se ha hecho un puente entre los positivos para ahorrar cableado. En el caso de los negativos, al precisar de una resistencia de 10 k $\Omega$  para poder realizar el pull down, no se pueden unir, pues crearíamos una resistencia en serie de 40 k $\Omega$ .



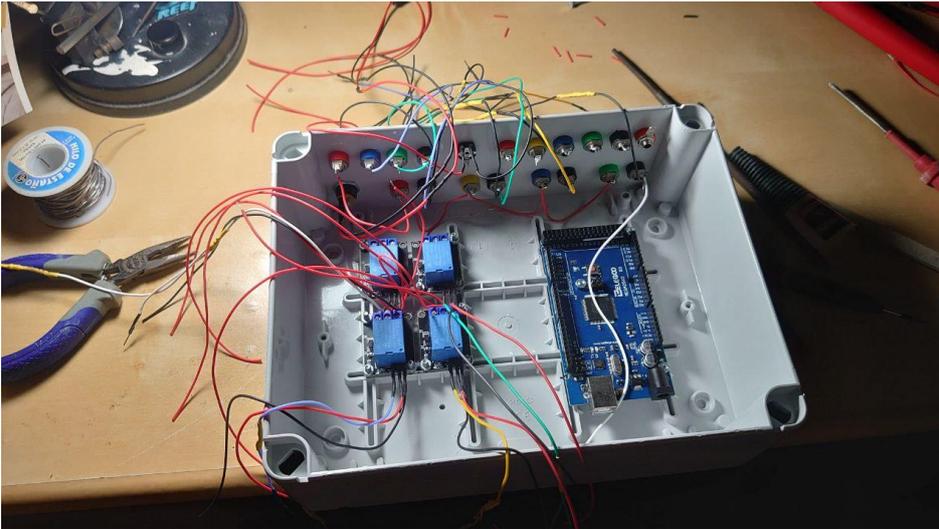
*Il·lustració 118 Creació de 4 cables per als negatius de tots els finals de carrera*



*Il·lustració 119 Finalització de cableado de bananes hembra*

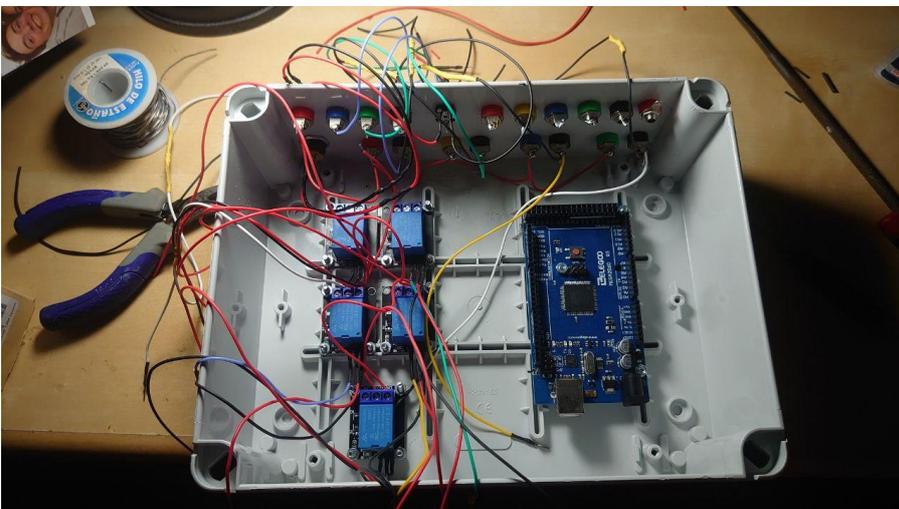
Finalmente, las bananas hembras del variador de frecuencia, al tener que ir conectadas al relé, se decidió conectarlas una vez instalados estos.

Ahora se pasa a instalar los relés, previamente cableados. Estos tienen 1 negativo, un positivo y un pin de señal (que será el que Arduino utilice para determinar su activación). En este pin de señal, para poder indicar al usuario que es lo que está activado, se añade el LED, que hará de comunicador visual. De este pin de señal, entonces, deberá salir una bifurcación hacia un LED. Antes de llegar al led se deberá instalar una resistencia de  $220 \Omega$  para proteger el LED. Finalmente, un cable negro saldrá de la pata negativa del LED hacia el polo negativo de Arduino, cerrando así el circuito.



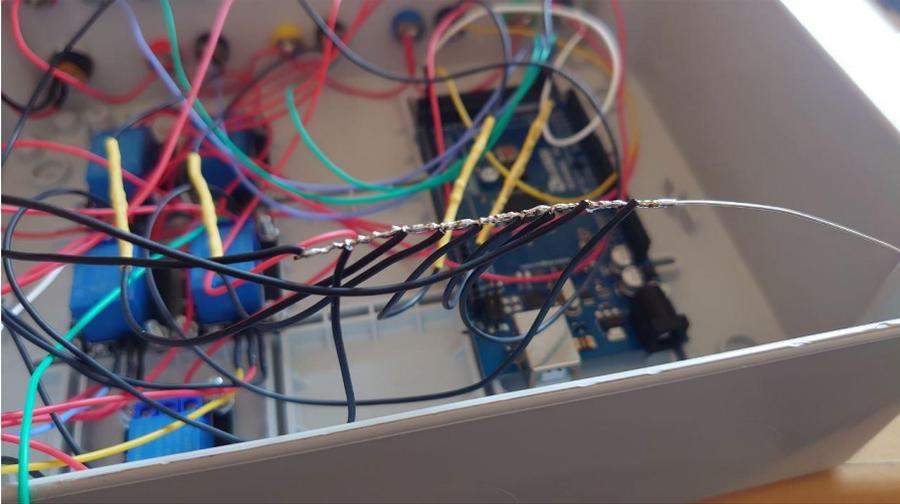
*Ilustración 120 Instalación y cableado de relés*

A estos relés, como se ha comentado anteriormente, se les cableará hacia el variador de frecuencia. En el borne del relé NO se le llevará la fuente de alimentación de 24 V que nos ofrece el variador y del común va hacia la banana hembra que a su vez irá a la entrada digital del variador de frecuencia para mandar las diferentes órdenes a realizar. Para ahorrar cableado se ha puenteado NO entre relés para conseguir 24 V .

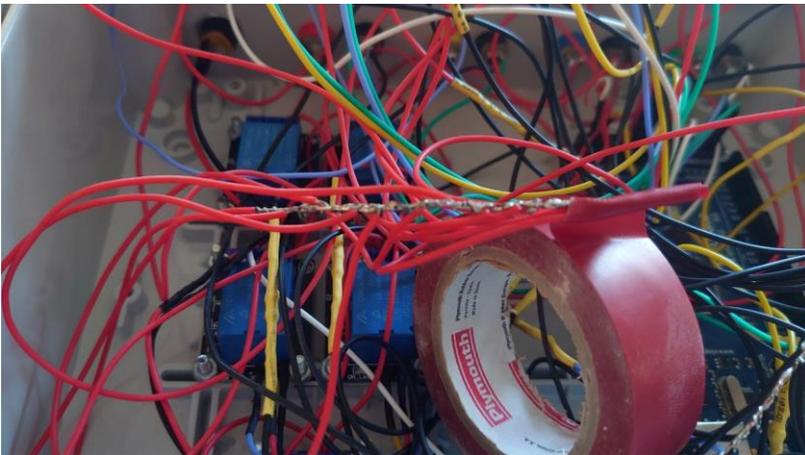


*Ilustración 121 Cableado interno de los 5 relés al completo*

Por último, quedará unir todos los negativos y los positivos y conectar los pines a la placa de Arduino. Para conectar negativos y positivos se ha hecho una especie de embarrado en el que van todos los cables soldados.



*Il·lustració 122 Embarrado de polo negativo conectado a GND de Arduino*



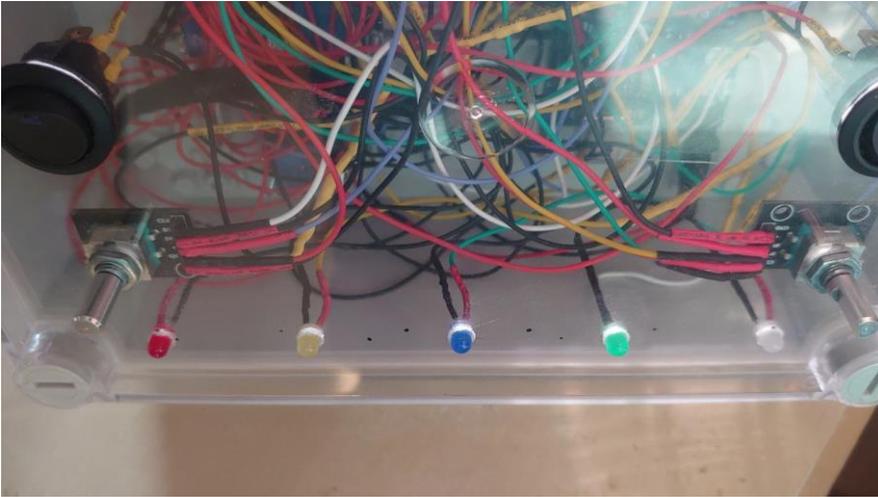
*Il·lustració 123 Embarrado de polo positivo conectado a 5V de Arduino*

Posteriormente, se protegen ambos embarrados con cinta aislante de color negra y roja para diferencia el polo negativo y el polo positivo.



*Il·lustració 124 Encintado con cinta aislante del embarrado negativo o GND*

Colocamos los LED en la parte frontal transparente de la caja. Para su colocación no es necesario el uso de pegamentos especiales, pues quedan bien sujetos a presión.

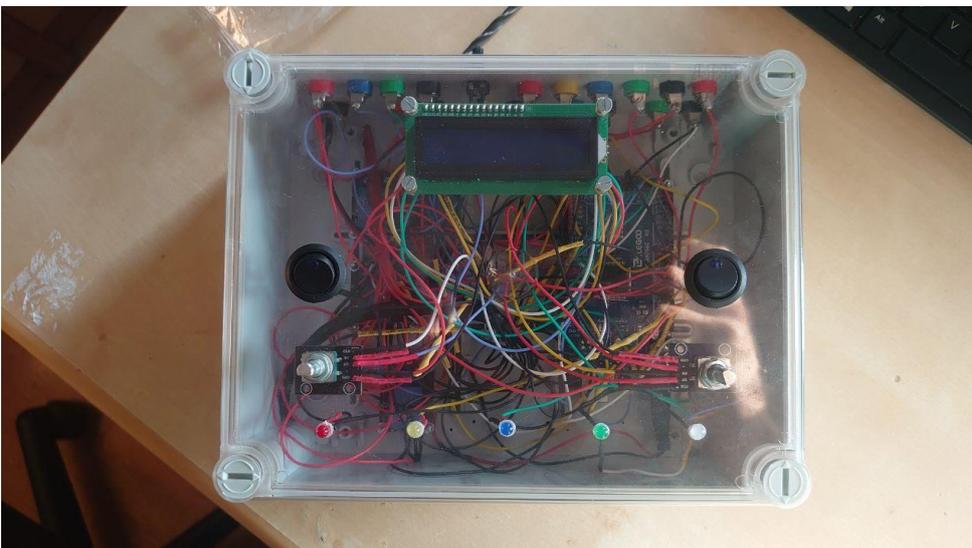


*Ilustración 125 LED's instalados en parte frontal*

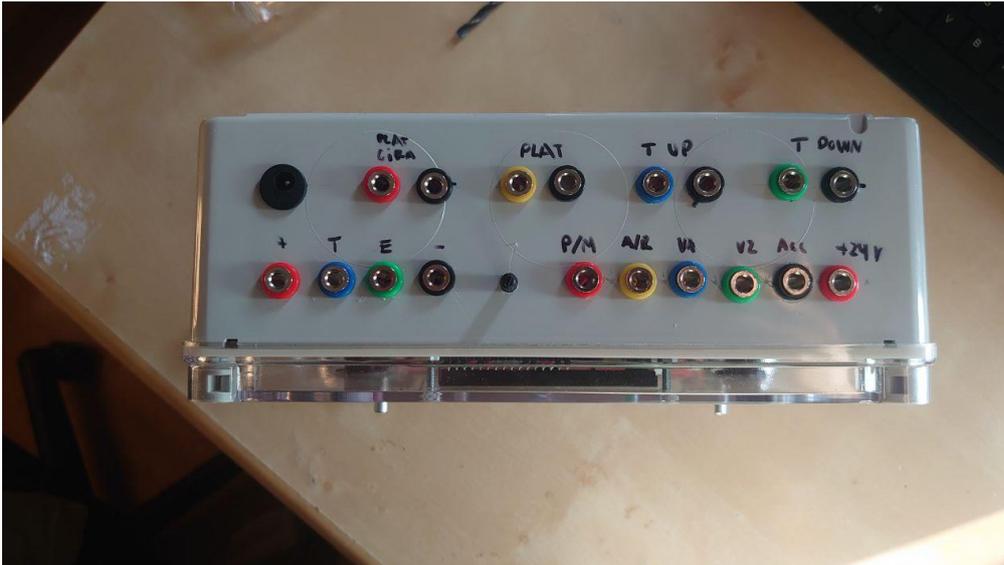
Finalizamos la instalación eléctrica comprobando un correcto funcionamiento de la caja de componentes y se cierra con los tornillos de plástico que vienen incluidos en la caja.

Para el reconocimiento de las entradas y salidas que se ha instalado en la caja de componentes, se marcarán, con la ayuda de un rotulador de tinta fija, los diferentes terminales tipo banana.

Una vez hecho todas las comprobaciones pertinentes, se da por ensamblada la caja de componentes.



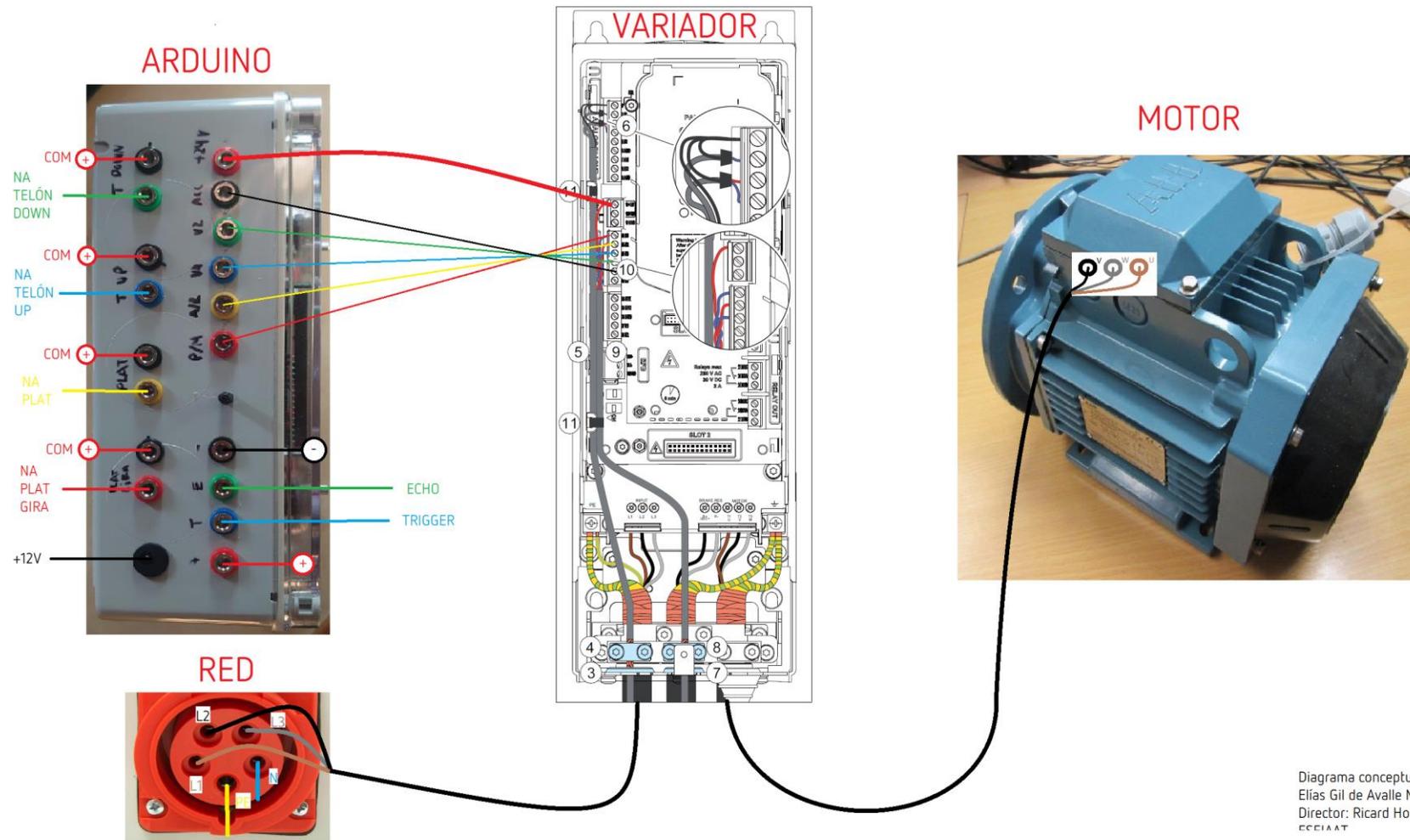
*Ilustración 126 Caja de componentes ABB finalizada*



Il·lustració 127 Conexiones de entradas y salidas digitales y fuente externa de 24 Vdc hacia los relés



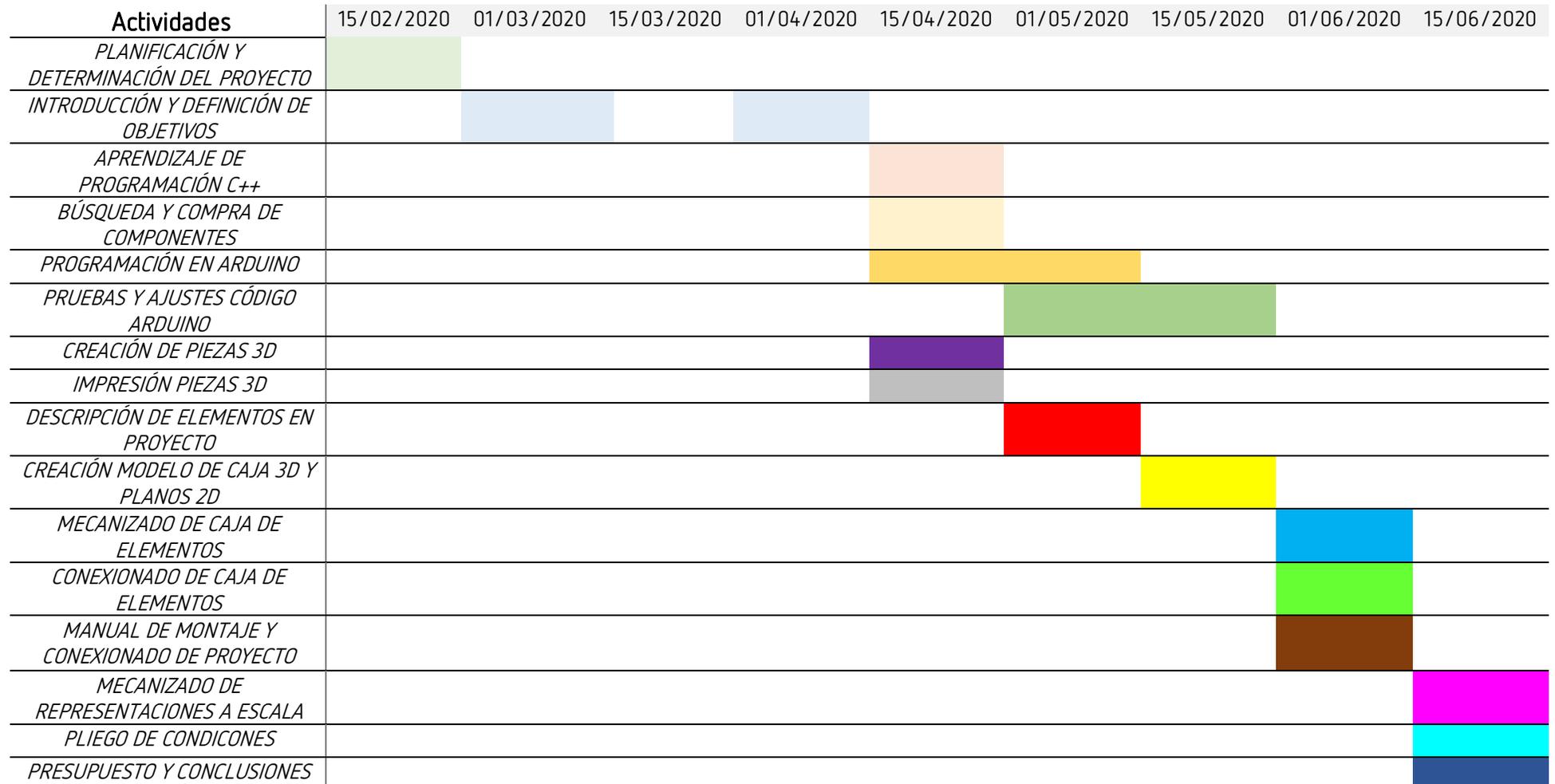
Il·lustració 128 Parte trasera de la caja de componentes con los visos de fijación vistos



Il·lustració 129 Diagrama conceptual de la instal·lació

Diagrama conceptual  
 Elias Gil de Avallé Muñoz  
 Director: Ricard Horta Bernús  
 CECIAAT

## 8 DIAGRAMA DE GANTT



## 9 IMPACTO MEDIOAMBIENTAL

El impacto ambiental es el impacto de las actividades humanas en el medio ambiente. El comportamiento de las personas a menudo puede cambiar la naturaleza y los ecosistemas. Por lo tanto, el diseño, la ejecución y la aplicación de cualquier proyecto deben conservarse y mantenerse para lograr el mínimo impacto en el medio ambiente.

Se cree que, para llevar a cabo un estudio de impacto ambiental completo, se requiere el trabajo de un equipo multidisciplinario especializado en muchos temas, por lo tanto, en el proyecto en cuestión, la evaluación de impacto ambiental que se realizará a continuación será básica debido a que no es el objeto principal del proyecto.

Debido a la pandemia que ha producido el virus SARS-CoV-2 y las limitaciones que esto supone, se ha comprado cantidad de material en exceso al proveedor por su venta en packs. Los pulsadores, hembras de banana de 4mm, encoders, etc de materiales que han sobrado, se depositarán en la universidad para un futuro uso de ellos. Los materiales que han resultado defectuosos o inservibles se han reciclado en su respectivo punto de reciclaje.

En caso de necesidad de desarticulación del equipo, los componentes deberán ser tratados uno por uno:

- Caja de plástico → Reciclaje junto a los plásticos
- Cableado → Punto verde para que posteriormente, una máquina separe el material plástico del cobre.
- Equipos electrónicos → Deberán ser gestionados por un punto verde

Si estos fueran gestionados de forma incorrecta, podría provocar agotamiento prematuro de materias primas, consumo de energía extra y contaminación.

El uso de un variador de frecuencia contribuye de forma directa a la eficiencia energética, haciendo un buen uso de la energía consumida y optimizándola.

## 10 SEGURIDAD DE USO

La operación o instalación incorrecta del equipo puede causar lesiones personales y daños a la propiedad. En particular, operar bajo tensión puede causar una descarga eléctrica a la persona que lo manipule e incluso la muerte. La instalación o el mantenimiento incorrectos también pueden provocar incendios.

Leer detenidamente el pliego de condiciones adjunto es encarecidamente necesario antes de utilizar Arduino. Durante toda la vida útil del dispositivo, se deberán seguir las correctas prácticas de manipulación e instalación. En particular, deben observarse las normas de instalación indicadas en el "Código Eléctrico Nacional".

Para este equipo se deberá trabajar con una fuente de tensión de 12V y una corriente máxima de 1A. Así mismo el conector Jack deberá ser de 2,1 mm.

La caja, pese a ser IP 56, deberá ser utilizada exclusivamente en interiores, nunca en el exterior.

## 11 CONCLUSIONES

En este proyecto se ha buscado una solución compacta a partir de un sistema integrado de control para artes escénicas con capacidad de controlar un variador de frecuencia ABB que a su vez dominará un motor trifásico. En este caso se trabajó con el modelo ACS-580-A4 y un motor trifásico de 1,1 kW.

Finalmente, se ha conseguido una solución práctica para los estudiantes con la cual puedan trabajar y practicar con solvencia, pues se ha tenido en cuenta el nivel de conocimiento que podría tener una persona sin conocimientos previos en la materia. El sistema es sencillo e intuitivo y cuenta con unas estructuras de representación a pequeña escala en las que, el estudiante, podrá constatar en la práctica el funcionamiento real del mismo.

En un principio, el sistema debía ser controlado por un autómata ABB mucho más potente que el sistema utilizado finalmente, el Arduino. Esto ha sido ocasionado por las dificultades de no poder acudir a la universidad a realizar pruebas en el laboratorio con este autómata a causa de la Covid-19. Gracias, en parte, a esta situación generada por el virus, hemos conseguido crear un equipo algo menos potente, pero con las mismas capacidades que se pretendía en un inicio, que era un control del motor sencillo e intuitivo (ahorrándonos aproximadamente unos 1.900 €- 2.000 € que cuesta además el autómata de ABB respecto al Arduino).

Es cierto que crear esta herramienta con Arduino y no con el autómata ABB es más laborioso, pues se tiene que escribir todo el código fuente para indicarle a la placa base si está detectando o no una entrada o salida digital o para dar instrucciones. El autómata de ABB ya dispone de un programa específico con una serie de bloques

prediseñados que generan una facilidad extra a la hora de programar las funciones a realizar por el equipo.

De igual manera, también ha sido una dificultad adicional a la hora de llevar a cabo el proyecto, pues se partía de conocimientos nulos en programación de IDE Arduino (que finalmente han sido adquiridos a nivel básico). De haber tenido un conocimiento más avanzado en programación IDE Arduino se podría crear un código más reducido, limpio y fluido y lo más importante, con una ocupación de memoria inferior.

A lo largo del proyecto se ha ido perfeccionando el sistema para que sea lo más intuitivo y eficaz posible. Uno de los problemas más importantes que se han generado durante el transcurso del proyecto es el mecanismo del pulsador que generaba flancos negativos y positivos que impedían que Arduino interpretara correctamente las señales. Fue corregido mediante la técnica de las resistencias de *PULL UP & PULL DOWN*.

Otro de los problemas que genera el utilizar Arduino, es que es una placa base que maneja muy poca corriente y que a veces es insuficiente para poder abastecer a todos los equipos necesarios. Esto genera reinicios puntuales del sistema.

En general, pese a toda la problemática que ha generado la situación del SARS-CoV-2, se ha logrado sortear los aspectos más críticos del proyecto como por ejemplo acudir al laboratorio para poder crear toda la caja de conexiones o tener que cambiar por completo el sistema de gestión del convertidor.

A la espera de poder probar si el útil de prácticas para los alumnos de la ESTAE funciona con el convertidor, se puede confirmar que toda la parte electrónica se encuentra en pleno funcionamiento, pues se han realizado pruebas a lo largo del proyecto con leds para comprobar si daba la señal de 5V que se requería. Las últimas pruebas realizadas constatan que el dispositivo se desenvuelve perfectamente.

## 12 BIBLIOGRAFÍA

- [1] E. Super and S. Kit, “Elegoo Super Starter Kit para UNO.”
- [2] “Alquiler plataforma giratoria.”  
[http://www.pascualinestructures.com/portfolio\\_page/giratorio-3mts/](http://www.pascualinestructures.com/portfolio_page/giratorio-3mts/) (accessed May 20, 2020).
- [3] “Velocidad de ejecución de instrucciones (Solucionado).”  
<https://forum.arduino.cc/index.php?topic=86962.0> (accessed May 19, 2020).
- [4] “🤖 Funciones arduino void loop () y void setup () | Proyectos con Arduino.”  
<https://proyectosconarduino.com/curso/funciones-arduino-void-loop-y-void-setup/> (accessed May 19, 2020).
- [5] “Descubre el Palau de les Arts | Ciutat de les Arts i les Ciències.”  
<https://www.cac.es/es/palau-de-les-arts/Descubre-y-conoce-el-Palau-de-les-Arts/descubre-el-palau.html> (accessed May 17, 2020).
- [6] “Palacio de las Artes Reina Sofía de Valencia | Actividades con niños.”  
<https://saposyprincesas.elmundo.es/actividades-ninos/valencia/cultura/monumentos/palacio-de-las-artes-reina-sofia/> (accessed May 17, 2020).
- [7] “El edificio | Teatro Real.” <https://www.teatroreal.es/es/edificio#ficha-técnica-18607-1> (accessed May 17, 2020).
- [8] “Los engranajes del Teatro Real al descubierto | Sobre Arquitectura y más | Desde 1998.” <https://www.metalocus.es/es/noticias/los-engranajes-del-teatro-real-al-descubierto> (accessed May 17, 2020).
- [9] “Home | Teatro Real.” <https://www.teatroreal.es/es> (accessed May 17, 2020).
- [10] “Sistema de elevación de plataformas utilizado por ESF Santiago Fajardo. - Santiago Fajardo.” <https://www.santiagofajardo.com/spiralift-sistema-elevacion/> (accessed May 17, 2020).
- [11] “Tramoya. Teatros del Canal | www.urdaci.es | Alfredo Urdaci | Flickr.”  
<https://www.flickr.com/photos/urdaci/4358572875> (accessed May 17, 2020).
- [12] “(55) Equipamiento escénico / Stage Equipment - YouTube.”  
[https://www.youtube.com/watch?v=\\_q\\_SXW0Xn4](https://www.youtube.com/watch?v=_q_SXW0Xn4) (accessed May 17, 2020).
- [13] “(No Title).” [https://incual.mecd.es/documents/20195/94271/ART524\\_3+-+Q\\_Documento+publicado/e8980461-5a3c-41b0-8a2f-ace84abefc93](https://incual.mecd.es/documents/20195/94271/ART524_3+-+Q_Documento+publicado/e8980461-5a3c-41b0-8a2f-ace84abefc93) (accessed May 17, 2020).
- [14] “(No Title).”

- [15] “TECNOLOGÍA SPIRALIFT Archives - Paco Spiralift - Automotive & Industrial Lifting Solutions.”  
<https://www.pacospiralift.com/es/solutions/tecnologia-spiralift/> (accessed May 17, 2020).
- [16] “INICIO - Paco Spiralift - Automotive & Industrial Lifting Solutions.”  
<https://www.pacospiralift.com/es/> (accessed May 17, 2020).
- [17] “Spiralift.” <https://www.galasytems.com/es/spiralift-es/> (accessed May 17, 2020).
- [18] “STAGE. Enginyeria escènica. Enginyeria per a representacions escèniques.”  
<http://ca.strong.es/strongstage.asp> (accessed May 17, 2020).
- [19] “chemtrolstage - Chemtrol | Theatres & Global Stage Solutions.”  
<https://chemtrolstage.com/en/> (accessed May 17, 2020).
- [20] “(No Title).” [https://retom.es/pdf/UNE-CWA\\_15902-1\\_2014.pdf](https://retom.es/pdf/UNE-CWA_15902-1_2014.pdf) (accessed May 17, 2020).
- [21] “norma española,” 2014. Accessed: 17-May-2020. [Online]. Available: [www.aenor.es](http://www.aenor.es).
- [22] “Telones de teatro. Ciclorama teatro. Cortinas acústicas.”  
[https://retom.es/equipamientos\\_escenicos/textiles\\_escenicos.php](https://retom.es/equipamientos_escenicos/textiles_escenicos.php) (accessed May 17, 2020).
- [23] “Zona Maker - Resistencias de Pull-Up y Pull-Down.”  
<https://www.zonamaker.com/electronica/intro-electronica/teoria/resistencias-de-pull-up-y-pull-down> (accessed May 13, 2020).
- [24] “LCD 16x2 | HeTPro.” <https://hetpro-store.com/lcd-16x2-blog/> (accessed May 13, 2020).
- [25] “Rele o Relevador. Qué es, Tipos, Funcionamiento Aprende Facil.”  
<https://www.areatecnologia.com/electricidad/rele.html> (accessed May 13, 2020).
- [26] “Manejar cargas de más de 220V con Arduino y salida por relé.”  
<https://www.luisllamas.es/arduino-salida-rele/> (accessed May 13, 2020).
- [27] “(45) Arduino desde cero en Español - Capítulo 22 - Codificador rotatorio KY-040 rotary encoder - YouTube.”  
<https://www.youtube.com/watch?v=fnDHQ6YcxTs> (accessed May 10, 2020).
- [28] “Qué es C++: Características y aplicaciones | OpenWebinars.”  
<https://openwebinars.net/blog/que-es-cpp/> (accessed May 10, 2020).
- [29] “Arduino IDE - Wikipedia, la enciclopedia libre.”  
[https://es.wikipedia.org/wiki/Arduino\\_IDE](https://es.wikipedia.org/wiki/Arduino_IDE) (accessed May 10, 2020).

- [30] “Qué es Arduino, cómo funciona y qué puedes hacer con uno.”  
<https://www.xataka.com/basics/que-arduino-como-funciona-que-puedes-hacer-uno> (accessed May 10, 2020).
- [31] “» Saber Más Sobre Arduino Mega 2560 | PatagoniaTec.”  
<https://saber.patagoniatec.com/2014/06/arduino-mega-2560-atmega-mega-arduino-clon-compatible-argentina-tutorial-basico-informacion-arduino-argentina-ptec/#Las-caracteristicas-principales-de-Arduino-MEGA2560> (accessed May 10, 2020).
- [32] “¿Por qué elegir un variador de frecuencia y no un reductor? | AUTYCOM.”  
<https://www.autycom.com/variador-de-frecuencia-en-lugar-de-un-reductor/> (accessed May 10, 2020).
- [33] “Motorreductores y reductores de velocidad mecánicos.”  
<https://tercesa.com/noticias/que-es-un-motorreductor/> (accessed May 10, 2020).
- [34] “Motores monofásicos, bifásicos y trifásicos: todo lo que necesitas saber – Blog CLR.” <https://clr.es/blog/es/motores-monofasicos-bifasicos-trifasicos/> (accessed May 09, 2020).
- [35] “Motor asíncrono - Monografias.com.”  
<https://www.monografias.com/trabajos91/motor-asincrono/motor-asincrono.shtml> (accessed May 09, 2020).
- [36] “Faraday’s Law.” <http://hyperphysics.phy-astr.gsu.edu/hbasees/electric/farlaw.html> (accessed May 09, 2020).
- [37] “Motores eléctricos síncronos y motores asíncronos.”  
<https://reyvarsur.com/motores-electricos-sincronos-y-motores-asincronos/> (accessed May 09, 2020).
- [38] “Dispositivo de parada de emergencia ES... | EUCHNER – More than safety.”  
<https://www.euchner.de/es-es/productos/dispositivos-de-parada-de-emergencia/dispositivo-de-parada-de-emergencia-es/> (accessed May 09, 2020).
- [39] “ACS580 / Datos técnicos.”
- [40] “(No Title).” <http://www.nduonline.cl/fichas/13140583AB.pdf> (accessed May 09, 2020).
- [41] “ACS580-01-07A2-4 - ACYR SYSTEMS.” <http://www.acyr.es/variador-acs580/10355-acs580-01-07a2-4.html> (accessed May 08, 2020).
- [42] “Convertidores / Variadores de Frecuencia | ABB.” <https://new.abb.com/drives/es> (accessed May 07, 2020).
- [43] “Variadores de frecuencia, uso y aplicaciones - Libros Eumed.net.”  
<https://www.eumed.net/libros/1752/index.html> (accessed May 07, 2020).

- [44] “(10) CONTADORES DE PULSOS ASCENDENTES Y DESCENDENTE PARA ARDUINO|| BIEN EXPLICADO 2019 - YouTube.”  
<https://www.youtube.com/watch?v=qQUYggDJRVw> (accessed Apr. 25, 2020).
- [45] “Arduino - Sketch.” <https://www.arduino.cc/en/Tutorial/Sketch> (accessed Apr. 22, 2020).
- [46] “BUMAT | productos | plataformas giratorias | dsma30-300.”  
<http://www.bumat.com/sp/produkte/drehscheiben/dsma30/dsma30.html>  
(accessed Apr. 22, 2020).
- [47] “Error de compilación con LiquidCrystal![solucionado].”  
<https://forum.arduino.cc/index.php?topic=94285.0> (accessed Apr. 22, 2020).
- [48] “Arduino - LiquidCrystal.” <https://www.arduino.cc/en/Reference/LiquidCrystal>  
(accessed Apr. 22, 2020).
- [49] “Escenarios giratorios - Plató Giratorio 360° | Fabricación de plataformas giratorias | Pellicertech.” [http://www.pellicertech.com/es/productos/plataformas-giratorias/5/escenarios-giratorios---plato-giratorio-360\\_](http://www.pellicertech.com/es/productos/plataformas-giratorias/5/escenarios-giratorios---plato-giratorio-360_) (accessed Apr. 22, 2020).
- [50] “Espacios de usos múltiples.”  
<https://www.galasystems.com/es/soluciones/espacios-de-usos-multiples/>  
(accessed Apr. 22, 2020).
- [51] “Arduino: Usando la función millis() en lugar de delay() | Robots Didácticos.”  
<http://robots-argentina.com.ar/didactica/arduino-usando-la-funcion-millis-en-lugar-de-delay/> (accessed Apr. 21, 2020).
- [52] “Ciclo while en C++. Aprender a programar ciclos while en C++.”  
<https://www.programarya.com/Cursos/C++/Ciclos/Ciclo-while> (accessed Apr. 19, 2020).
- [53] “SENSORES: Final de carrera y R de 10 k - YouTube.”  
<https://www.youtube.com/watch?v=st-iYZzR0tw> (accessed Apr. 19, 2020).
- [54] “Arduino desde cero en Español - Capítulo 22 - Codificador rotatorio KY-040 rotary encoder - YouTube.” <https://www.youtube.com/watch?v=fnDHQ6YcxTs>  
(accessed Apr. 18, 2020).
- [55] “Alimentación Arduino | Aprendiendo Arduino.”  
<https://aprendiendoarduino.wordpress.com/2016/11/09/alimentacion-arduino/>  
(accessed Apr. 18, 2020).
- [56] “Menu LCD+ Encoder + click encoder... como hacerlo?”  
<https://forum.arduino.cc/index.php?topic=287231.0> (accessed Apr. 18, 2020).

- [57] “Qué son y cómo usar interrupciones en Arduino.”  
<https://www.luisllamas.es/que-son-y-como-usar-interrupciones-en-arduino/>  
(accessed Apr. 18, 2020).
- [58] “Menú con submenús para LCD en Arduino + Encoder - el blog de giltesa.”  
<https://giltesa.com/2017/07/29/menu-submenus-lcd-arduino-encoder> (accessed Apr. 18, 2020).
- [59] “Diferencia para definir entradas digitales.”  
<https://forum.arduino.cc/index.php?topic=547172.0> (accessed Apr. 18, 2020).
- [60] “Para qué sirve un enum?” <https://forum.arduino.cc/index.php?topic=488064.0>  
(accessed Apr. 18, 2020).
- [61] “Arduino en español: Boolean.”  
<http://manueldelgadocrespo.blogspot.com/p/boolean.html> (accessed Apr. 18, 2020).
- [62] “Cómo usar un Encoder con un Arduino – Javier Loureiro.”  
<https://javierloureiro.wordpress.com/2014/04/24/como-usar-un-encoder-con-un-arduino/> (accessed Apr. 18, 2020).
- [63] “Rotary Encoders | Tienda y Tutoriales Arduino.”  
<https://www.promotec.net/rotary-encoders/> (accessed Apr. 18, 2020).
- [64] “APRENDER A USAR UN DISPLAY LCD – DIYMakers.”  
<http://diymakers.es/aprender-usar-un-display-lcd/> (accessed Apr. 18, 2020).
- [65] “Estructura de Control ( if-else ) y ( Bucles) — Steemit.”  
<https://steemit.com/spanish/@neliobatis/estructura-de-control-if-else-y-bucles>  
(accessed Apr. 14, 2020).
- [66] “Drive composer - Herramientas de PC | ABB.”  
<https://new.abb.com/drives/es/herramientas/drive-composer> (accessed Apr. 14, 2020).
- [67] “Zona Maker - Resistencias de Pull-Up y Pull-Down.”  
<https://www.zonamaker.com/electronica/intro-electronica/teoria/resistencias-de-pull-up-y-pull-down> (accessed Apr. 14, 2020).
- [68] “Funcionamiento de variadores escalar o vectorial | Formación para la Industria 4.0.” <https://automatismoindustrial.com/d-automatizacion/1-7-arrancadores-electronicos-y-variadores-de-velocidad/funcionamiento-de-variadores-escalar-o-vectorial/> (accessed Apr. 14, 2020).
- [69] “(1) Arduino desde cero en Español - Capítulo 1 - Introducción Arduino e inicio del Curso - YouTube.”  
<https://www.youtube.com/watch?v=eBVvD85M12c&list=PLkjnQ3NFTPnY1eNyLDGi547gkVui1vyn2> (accessed Apr. 14, 2020).

- [70] “Curso de Arduino aprende a programar desde cero.”  
<https://programarfacil.com/blog/arduino-blog/curso-de-arduino/> (accessed Apr. 14, 2020).
- [71] “🏆 5 canales de Youtube de Cómo aprender a Programar en Arduino.”  
<https://descubrearduino.com/aprender/> (accessed Apr. 14, 2020).
- [72] “ec3a200ea1827cc2616b180d743274f0d48dc0d2 @ sensores-de-medida.es.”  
[Online]. Available: <https://sensores-de-medida.es/medicion/sensores-y-transductores/sensores-de-distancia/>.
- [73] “¿Sabes cómo funciona un ascensor?”  
<https://columnacero.com/tecnologia/25199/sabes-como-funciona-un-ascensor/>  
(accessed Apr. 05, 2020).
- [74] “Sensores de distancia - Sensing, Sensores de Medida.” <https://sensores-de-medida.es/medicion/sensores-y-transductores/sensores-de-distancia/> (accessed Apr. 05, 2020).
- [75] “(92) YouTube.” [https://www.youtube.com/watch?v=cHnnka\\_E4X0](https://www.youtube.com/watch?v=cHnnka_E4X0) (accessed Mar. 28, 2020).
- [76] T. Acs, “ABB general purpose drives ABB general purpose drives ,” pp. 1–2.
- [77] “Motores y generadores | ABB.” <https://new.abb.com/motors-generators/es>  
(accessed Mar. 28, 2020).