

UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE

ESCUELA POLITÉCNICA SUPERIOR DE ELCHE

MÁSTER EN INGENIERÍA INDUSTRIAL



"MEJORA DE UN ROBOT TREPADOR BINARIO (XROBIN):
MECÁNICA, ELECTRÓNICA, SOFTWARE Y
CONSTRUCCIÓN DE UN PROTOTIPO FUNCIONAL"

TRABAJO FIN DE MÁSTER

Septiembre – 2024

AUTOR: Adrián Campos Miralles
DIRECTOR/ES: Adrián Peidró Vidal

Contenido

1 INTRODUCCIÓN.....	9
1.1 PREÁMBULO.....	9
1.2 OBJETIVOS.....	13
1.3 RECURSOS.....	13
1.4 ESTRUCTURA DE LA MEMORIA.....	14
2 ESTADO INICIAL.....	15
2.1 PROTOTIPO INICIAL.....	16
2.2 PUNTOS DE MEJORA DEL PRIMER PROTOTIPO.....	17
3 DESARROLLOS.....	19
3.1 MECÁNICA.....	19
3.1.1 MODULARIZACIÓN.....	20
3.1.2 SERVOMÓDULOS.....	21
3.1.3 BASTIDOR.....	27
3.1.4 MECANISMO TIJERA.....	28
3.1.5 JUNTA PRISMÁTICA-ROTATIVA.....	31
3.1.6 PIES EXTERIORES.....	36
3.1.7 PIE INTERIOR O PLATAFORMA.....	38
3.2 ELECTRÓNICA.....	41
3.2.1 ACTUADORES.....	42
3.2.2 ELECTRÓNICA DE POTENCIA.....	44
3.2.3 ELECTRÓNICA DE CONTROL.....	47
3.2.4 MÓDULO DE COMPUTACIÓN Y COMUNICACIONES.....	51
3.2.5 SEÑALES.....	52
3.2.6 PCB.....	53
3.3 SOFTWARE.....	54
3.3.1 STATESMITH.....	58
3.3.2 QP (QUANTUM PLATFORM).....	60
3.3.3 BOOST-EXT/SML.....	62
3.3.4 MÁQUINA DE ESTADOS PARA EL XROBIN.....	63
4 RESULTADOS.....	68

4.1 CONSTRUCCIÓN DE LOS SERVOMÓDULOS.....	68
4.1.1 MECANIZADO DE LOS COLECTORES MAGNÉTICOS.....	68
4.1.2 MONTAJE DE LOS COLECTORES E IMANES.....	69
4.1.3 ENSAMBLAJE DEL SERVOMOTOR.....	70
4.1.4 CAMPO MAGNÉTICO DE LOS IMANES CONMUTABLES.....	71
4.2 MONTAJE DEL PIE EXTERIOR.....	73
4.3 MONTAJE DEL PIE INTERIOR.....	75
4.4 MONTAJE DEL BALANCÍN Y LA DESLIZADERA.....	76
4.5 MONTAJE DEL MECANISMO TIJERA.....	77
4.6 ELECTRÓNICA PARA EL PROTOTIPO.....	79
4.7 SOFTWARE PARA EL PROTOTIPO.....	82
4.8 ROBOT COMPLETO.....	86
5 CONCLUSIONES.....	91

Índice de tablas

Tabla 1: Características del motor Pololu vs MG996R.....	21
Tabla 2: Características del imán N48 vs N35.....	22
Tabla 3: Señales de control.....	52
Tabla 4: Descripción de los símbolos usados en los estados.....	63
Tabla 5: Acciones permitidas en cada estado.....	63
Tabla 6: Transiciones permitidas en cada estado.....	64
Tabla 7: Tabla de transiciones.....	65
Tabla 8: Resumen de la lista de materiales mecánicos.....	86
Tabla 9: Acrónimos en inglés y español.....	99

Índice de figuras

Figura 1: Imanes conmutables [1].....	10
Figura 2: Robot trepador cartesiano, Sky Cleaner [2].....	11
Figura 3: Robot trepador paralelo, Xrobin [3].....	11
Figura 4: Robot trepador con ruedas, Sumitomo [4].....	12
Figura 5: Descripción geométrica del Xrobin [3].....	15
Figura 6: Estados de traslación del Xrobin [3].....	16
Figura 7: Primer prototipo del Xrobin [3].....	16
Figura 8: Modelo 3D del nuevo prototipo del Xrobin.....	19
Figura 9: Micro-motor Pololu con reducción 1:1000 (a) y Servomotor MG996R (b).....	21
Figura 10: Características del imán original (N48).....	23
Figura 11: Características del nuevo imán (N35).....	23
Figura 12: Vista de sección (a) y vista 3D inferior (b) de un servomódulo.....	24
Figura 13: Vista de explosión de un servomódulo.....	24
Figura 14: Vista 3D inferior (a) y superior, derecha(b) de un servomódulo.....	25
Figura 15: Vista 3D del imán superior con su acoplador.....	25
Figura 16: Vista seccionada de la planta del pie exterior.....	26
Figura 17: Vista lateral (a) y superior (b) de los imanes, colectores magnéticos y un acoplador.....	26
Figura 18: Vista superior de un servomódulo montado sobre un pie exterior....	26
Figura 19: Vista superior del bastidor.....	27
Figura 20: Vista lateral del bastidor.....	27
Figura 21: Vista seccionada de los agujeros en los laterales del bastidor.....	27
Figura 22: Vista 3D del mecanismo de tijera.....	28
Figura 23: Vista superior del mecanismo de tijera.....	28
Figura 24: Vista latera del mecanismo de tijera.....	29
Figura 25: Vista seccionada de un apoyo de un actuador lineal.....	29
Figura 26: Vista lateral seccionada del mecanismo de tijera para comprobar el juego entre los distintos componentes.....	30

Figura 27: Vista detalle seccionada del apoyo del actuador inferior de traslación.....	30
Figura 28: Vista 3D del balancín y la justa prismática-rotativa.....	31
Figura 29: Vista seccionada del balancín y la justa prismática-rotativa.....	31
Figura 30: Vista 3D de la justa prismática-rotativa (a) y vista 3D del balancín (b).....	32
Figura 31: Detalle 1 (a) y 2 (b) de la vista seccionada del balancín y la justa prismática-rotativa.....	32
Figura 32: Vista 3D de la interfaz hexagonal del balancín.....	33
Figura 33: Vista superior de la interfaz hexagonal del balancín, izquierda (a). Vaso hexagonal de carraca, derecha (b).....	33
Figura 34: Detalle 3 de la vista seccionada del balancín y la justa prismática-rotativa.....	34
Figura 35: Vista seccionada de la interfaz entre la junta prismática-rotativa con el bastidor.....	34
Figura 36: Detalle 4 de la vista seccionada del balancín y la justa prismática-rotativa.....	34
Figura 37: Vista 3D de la interfaz entre el bastidor y la junta prismática-rotativa en los extremos del recorrido.....	35
Figura 38: Detalle 5 de la vista seccionada del balancín y la justa prismática-rotativa.....	35
Figura 39: Vista superior del pie exterior.....	36
Figura 40: Vista superior de la base del pie exterior.....	36
Figura 41: Vista frontal seccionada del pie exterior.....	37
Figura 42: Vista frontal seccionada del pie exterior, incluyendo los servomódulos.....	37
Figura 43: Vista 3D del pie interior.....	38
Figura 44: Vista superior del pie interior.....	39
Figura 45: Vista inferior del pie interior.....	39
Figura 46: Vista (a) y (b) de la interfaz entre la PCB, el servomódulo y el pie interior.....	40
Figura 47: Vista seccionada entre la interfaz del pie interior y los pilares.....	40

Figura 48: Diagrama de bloques del Xrobin.....	41
Figura 49: Actuador lineal LS50 de Actuonix.....	42
Figura 50: Extracto del datasheet del LS50.....	42
Figura 51: Extracto del datasheet del MG996R.....	43
Figura 52: BQ25703A (a) y EBQ76920PW (b).....	45
Figura 53: Convertidor Traco TSR.....	46
Figura 54: Módulo convertidor de voltaje genérico.....	46
Figura 55: Diagrama de pines de una placa Bluepill con un procesador STM32F103.....	47
Figura 56: Placa ESP32 con módulo Wifi y Bluetooth integrado.....	48
Figura 57: Pines L293D (a) y Esquemático L293D (b).....	48
Figura 58: Extracto del datasheet del L293D.....	49
Figura 59: Aplicación típica (a) y encapsulado (b) INA138.....	50
Figura 60: Aplicación típica (a) y encapsulado (b) ACS712TELC-05B.....	50
Figura 61: Modelo conceptual de la PCB del Xrobin.....	53
Figura 62: Diagrama de bloques del software de Klipper.....	54
Figura 63: Interfaz de usuario de Mainsail.....	55
Figura 64: Diagrama de una máquina de estados para un teclado.....	57
Figura 65: Flujo de trabajo de Statesmith.....	58
Figura 66: Máquina de estados en draw.io como entrada para Statesmith.....	59
Figura 67: Diagrama de bloques de QP.....	60
Figura 68: Interfaz de usuario de QM.....	60
Figura 69: Propuesta de la máquina de estados principal para el Xrobin.....	66
Figura 70: Propuesta de una sub-máquina para el Xrobin donde se controlan las traslaciones.....	67
Figura 71: Amarre del bruto con un localizador magnético, mordaza (a) y máquina (b), e imagen durante el proceso de mecanizado (c).....	68
Figura 72: Piezas mecanizadas del primer amarre sobre el plano de referencia para el mecanizado (a) y amarre de los colectores magnéticos en una mordaza de aluminio para la segunda operación de mecanizado (b).....	69
Figura 73: Vista superior (a) e inferior (b) de los colectores magnéticos con imanes.....	69

Figura 74: Proceso de ensamblaje de los colectores magnéticos (a) y ensamblaje completado (b).....	70
Figura 75: Servomódulo antes (a) y después (b) de ser ensamblado.....	71
Figura 76: Observación del campo magnético en los estados desactivado [0°] (a) y activado [180°] (b) del imán conmutable original.....	71
Figura 77: Observación del campo magnético en los estados desactivado [0°] (a), [36°] (b), [72°] (c), [108°] (d), [144°] (e) y desactivado [180°] (f) del imán conmutable nuevo.....	72
Figura 78: Pie exterior ensamblado y elementos de fijación.....	73
Figura 79: Vista inferior del pie exterior ensamblado.....	74
Figura 80: Interfaces del pie exterior y el bastidor (a) y ensamblaje de ambas partes (b).....	74
Figura 81: Pie interior completamente ensamblado incluyendo componentes electrónicos.....	75
Figura 82: Vista inferior del pie interior ensamblado.....	75
Figura 83: Subcomponentes del balancín: bieletas, deslizadera, arandela, separador central, tuerca larga M8 y tornillos M8 avellanados.....	76
Figura 84: Balancín ensamblado (a) y ensamblado con la deslizadera (b).....	76
Figura 85: Mecanismo tijera antes de atornillar el actuador de traslación superior (extendido) a la bieleta.....	77
Figura 86: Mecanismo tijera después de atornillar ambos actuadores lineales (contraídos).....	78
Figura 87: Mecanismo tijera después de atornillar ambos actuadores lineales, superior contraído e inferior extendido.....	78
Figura 88: Módulos del robot separados mecánicamente pero conectados eléctricamente.....	79
Figura 89: Componentes electrónicos alojados en el pie interior, no ensamblados.....	80
Figura 90: Componentes electrónicos alojados en el pie interior, no ensamblados.....	80
Figura 91: Placas de conexiones auxiliar del pie interior (a) y bastidor (b), y un extremo de un arnés de cables.....	81

Figura 92: STM32CubeIDE, representación visual de la configuración de pines del microcontrolador.....	82
Figura 93: STM32CubeIDE, tabla de configuración de pines del microcontrolador.....	83
Figura 94: STM32CubeIDE, menú de configuración de los timers del microcontrolador.....	83
Figura 95: Xrobin 2.....	86
Figura 96: Vista frontal del Xrobin 2.....	87
Figura 97: Vista lateral del Xrobin 2.....	87
Figura 98: Xrobin 2 sobre la superficie metálica de un carro de herramientas..	88
Figura 99: Xrobin 2 sobre la superficie metálica de un carro de herramientas, en varias configuraciones: (a), (b), (c), (d).....	89
Figura 100: Xrobin 2 en el estado	90
Figura 101: Xrobin 2 en el estado	90
Figura 102: Carpetas del proyecto: General (a), 1_CAD (b), 2_EE (c), 3_SW (d).....	94
Figura 103: Imagen del esquemático de la PCB para el Xrobin 2.....	98
Figura 104: Imagen del editor de la PCB para el Xrobin 2.....	98

1 INTRODUCCIÓN

1.1 PREÁMBULO

La función de los robots trepadores es realizar trabajos en altura que resultan peligrosos para ser realizados por humanos debido a riesgos tales como el de caída o la exposición a entornos hostiles (altas temperaturas, radiación, químicos, etc.). Estos entornos se encuentran en el sector de la construcción, inspección y mantenimiento de estructuras, plantas de generación eléctrica y entre otros. Aunque existen diferentes tecnologías para adherirse a paredes tales como soluciones mecánicas o neumáticas, este trabajo se centrará en la tecnología de adhesión magnética.

Una primera forma de clasificar estos robots es según el tipo de imanes que usan para adherirse a las superficies metálicas:

- Imanes permanentes: En este caso, los imanes están siempre activados, lo que dificulta el funcionamiento normal del robot. Generando picos de fuerza en el momento de aproximación y separación de la superficie objetivo.
- Electroimanes: El principal inconveniente de este sistema es que necesita consumir energía para mantener la adhesión. Esto es problemático, porque aumenta el peso del robot y reduce el tiempo disponible de la misión debido a la necesidad de portar mayor cantidad de baterías.
- Imanes conmutables: Esta opción permite cambiar el estado de los imanes entre encendido o apagado, lo que permite ahorrar energía y un funcionamiento óptimo. Como desventaja implica la necesidad de un mecanismo para realizar el cambio de estado. En la siguiente imagen se describe un ejemplo de funcionamiento.

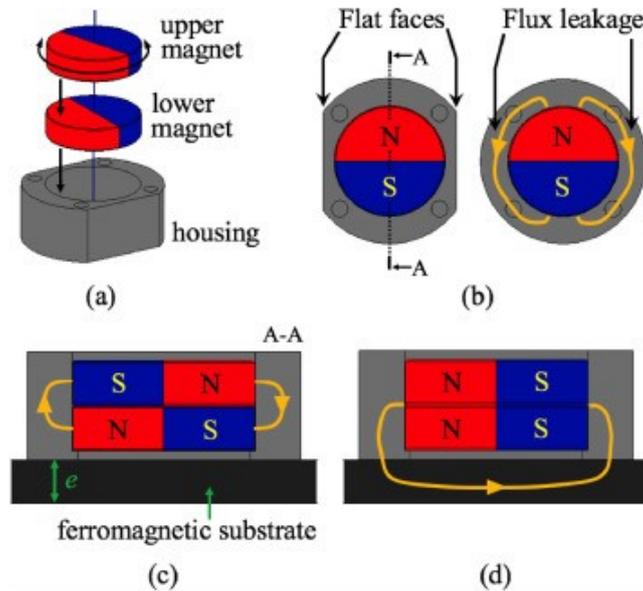


Figura 1: Imanes conmutables [1].

Una segunda forma de clasificar los robots trepadores es según su tipo de control:

- Binarios: El control es sencillo, el actuador puede estar extendido (1 lógico) o recogido (0 lógico). Es el caso del Xrobin.
- Continuos: El control es más complejo, el actuador puede estar en cualquier posición entre su estado extendido y recogido.

En una tercera clasificación, los robots trepadores pueden agruparse según la cadena cinemática que posibilita su locomoción:

- Robots cartesianos: El movimiento de cada dirección de avance es independiente y está controlado por un actuador independiente. En el caso de robots binarios, la resolución máxima es la carrera del actuador. Se pueden anidar actuadores para aumentar la resolución.

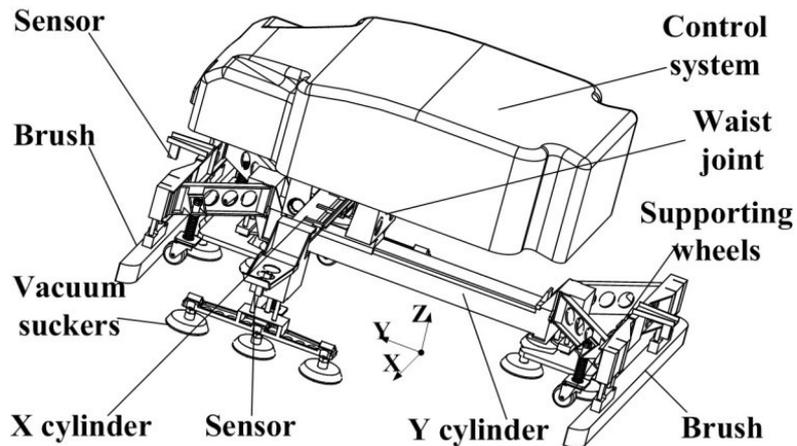


Figura 2: Robot trepador cartesiano, Sky Cleaner [2].

- Robots paralelos: El movimiento del efector final está determinado por una combinación de estados de sus actuadores. Es el caso del Xrobin, un robot binario (que puede ser controlado con señales todo o nada) con una geometría especial que permite obtener 4 veces más resolución que robots cartesianos con solo 2 actuadores. Este es el robot del que trata este Trabajo Final de Máster y que fue desarrollado en primera instancia por los investigadores del laboratorio de Automatización, Robótica y Visión por Computador (ARVC) de la UMH, en el artículo académico [3].

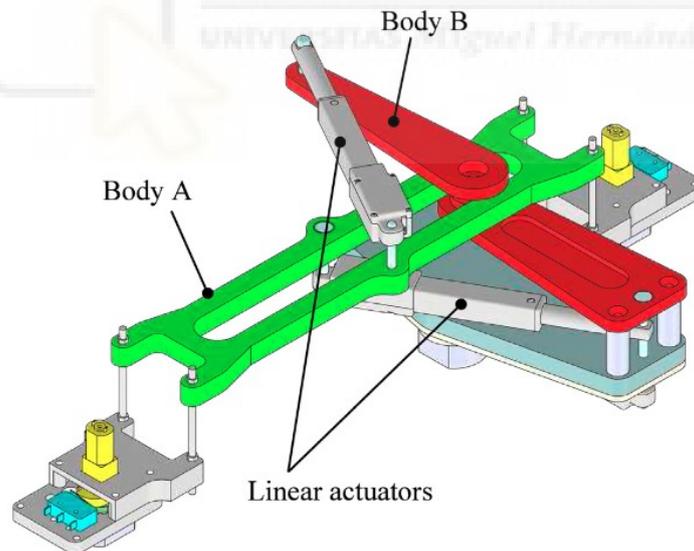


Figura 3: Robot trepador paralelo, Xrobin [3].

- Robots con ruedas: Robot que utiliza ruedas magnéticas para adherirse a la pared. Un ejemplo es este robot de Sumitomo para realizar soldaduras de forma autónoma en paredes curvas de acero.



Figura 4: Robot trepador con ruedas, Sumitomo [4].



1.2 OBJETIVOS

- Proyecto: La mejora del robot binario Xrobin desarrollado por el grupo ARVC de la UMH.
 - Mecánica: Mejora de la estructura, interfaces mecánicas, modularidad y estructura del proyecto en CAD. Se priorizará la facilidad de ensamblaje y la reducción de costes.
 - Electrónica: Propuesta de una arquitectura modular para los diferentes subsistemas (potencia, control, procesado y comunicación) y proponer los componentes y esquema eléctrico para llevarlos a cabo.
 - Software: Propuesta de una arquitectura de bloques basada en los sistemas actuales de impresión 3D y propuesta del uso de máquinas de estados para el control del robot.
 - Prototipo: Se trata de construir un prototipo aplicando en la medida de lo posible las propuestas anteriores.
- Personales: En el ámbito personal, el objetivo del autor de este trabajo es poner a prueba sus habilidades como Ingeniero Industrial en el ámbito de la mecánica y tratar de desarrollar nuevas habilidades de electrónica y software.

1.3 RECURSOS

Una enumeración de las herramientas empleadas durante el proyecto.

- Ofimática:
 - LibreOffice, para la redacción de la memoria.
- CAD:
 - Autodesk Inventor, para el diseño mecánico.
 - KiCad, para el diseño eléctrico.
- Software:
 - STM32cubeIDE, para la generación del software embebido.
 - QM, para la generación de máquinas de estado.
 - Statesmith, para la generación de máquinas de estado.
- Prototipado:
 - OrcaSlicer, para la generación de archivos de impresión 3D.
 - Impresora 3D Voron Trident 350x350.
 - Torno manual (Laboratorio de Robótica de la Universidad Miguel Hernández de Elche).
 - Fresadora CNC del centro de aprendizaje de Thyssenkrupp Presta AG.

1.4 ESTRUCTURA DE LA MEMORIA

Esta memoria se estructura de la siguiente forma:

1. Un preámbulo donde se da el contexto de los robots trepadores.
2. Una descripción del trabajo previo y el prototipo inicial.
3. Un listado de los recursos empleados para el desarrollo del proyecto.
4. El grueso del trabajo está en el apartado “Desarrollos”, donde se describen una por una las mejoras implementadas en mecánica, electrónica y software.
5. A continuación se muestra el resultado de todas las mejoras y el estado final del prototipo.
6. Finalmente se hace un resumen de los aprendizajes del proyecto y trabajos futuros.



2 ESTADO INICIAL

Cabe empezar esta sección mencionando el trabajo previo realizado por los investigadores del grupo ARVC en el artículo académico [3] “Design of a mobile binary parallel robot that exploits nonsingular transitions”.

En este artículo se describe diferentes tipos de robot y se focaliza en los robots binarios. Se describe la cinemática de un robot binario con un mecanismo “2RP R-PR”. En otras palabras dos actuadores lineales cruzados unidos a un efector que está guiado por una junta que es además prismática y rotativa.

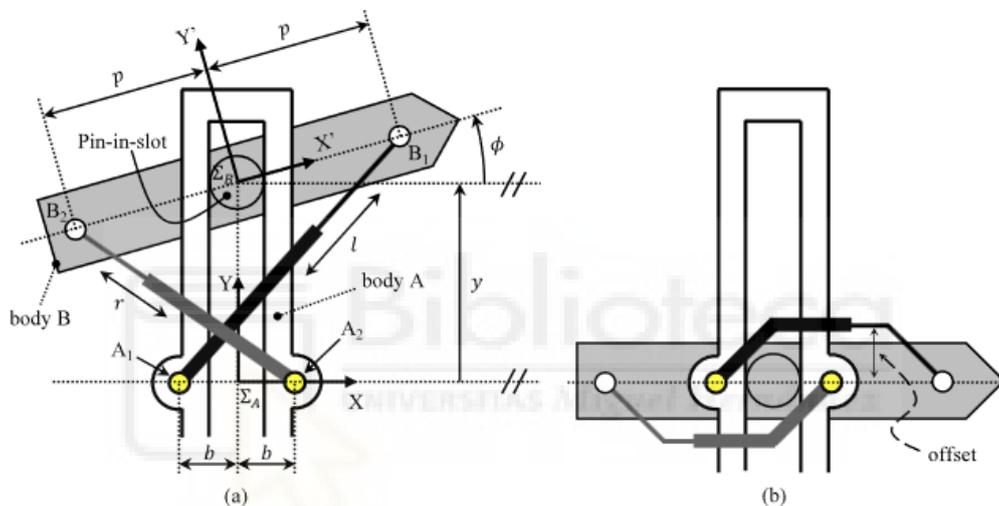


Figura 5: Descripción geométrica del Xrobin [3].

Esta configuración permite evitar el paso por la singularidad ya que la junta prismático-rotativa hace que estos grados de libertad sean dependientes de la posición de los actuadores. De este modo, se pueden adoptar 8 configuraciones, con 2 actuadores en vez de 4 configuraciones, con un sistema convencional.

En la siguiente imagen podemos ver las diferentes configuraciones.

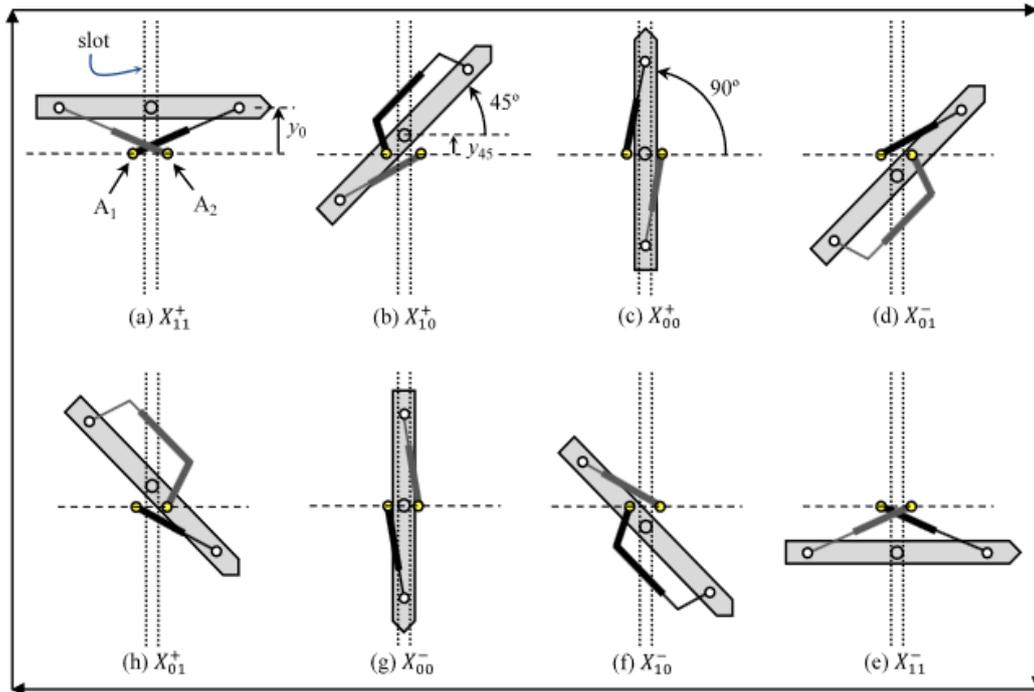


Fig. 4. Eight possible configurations X_{ij}^{\pm} of the 2RPR-PR mechanism when assembling it with crossed binary actuators. In this figure, the representation of body A is simplified as a vertical slot for easing the visualization of the postures. Also to ease the visualization and avoid overlappings, some linear actuators in subfigures (b), (d), (f) and (h) include artificial offsets like Fig. 2b.

Figura 6: Estados de traslación del Xrobin [3].

2.1 PROTOTIPO INICIAL

En esta imagen se muestra el primer prototipo del Xrobin.

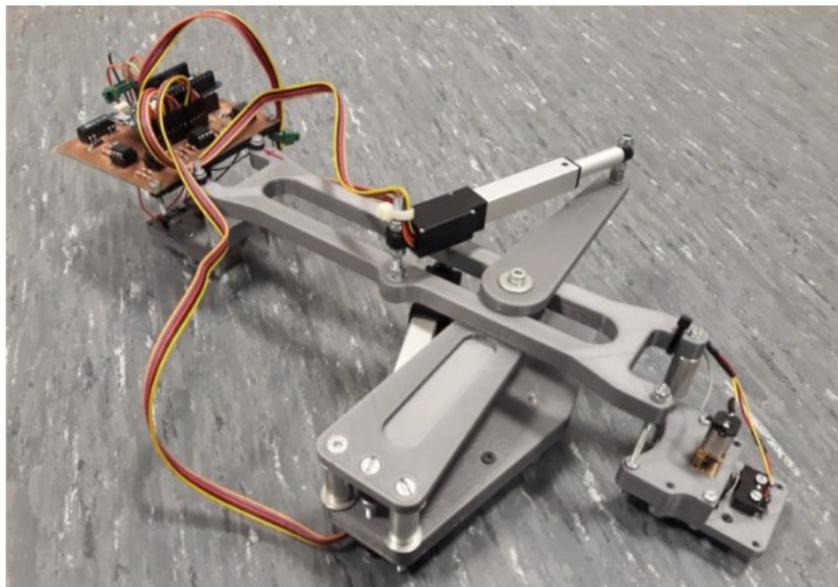


Fig. 15. Photo of the built prototype.

Figura 7: Primer prototipo del Xrobin [3].

El robot se puede dividir en dos subconjuntos tal y como se describe en [3]:

- Cuerpo A: Se corresponde con la deslizadera y los pies exteriores.
- Cuerpo B: Se corresponde con el pie interior y las bieletas que forma el efector central.

La electrónica de control se sitúa sobre uno de los pies exteriores. Se utiliza una placa Arduino UNO.

2.2 PUNTOS DE MEJORA DEL PRIMER PROTOTIPO

Debido al estado de desarrollo temprano del proyecto en cuyo marco se desarrolló el primer prototipo del robot Xrobin, dicho prototipo presentaba una serie de limitaciones que se enumeran a continuación:

- El robot no cuenta con un sistema de elevación que evite el rozamiento del cuerpo móvil con la superficie objetivo mientras se realizan traslaciones.
- La junta prismática-rotativa trata de combinar las dos funciones al mismo tiempo. Esto representa un problema de fabricación y funcionamiento, ya que requiere tolerancias de fabricación y montaje demasiado precisas para conseguir un movimiento fluido.
- Los soportes de los pies exteriores y actuadores están realizados con varillas roscadas. La rigidez y durabilidad de estos componentes es reducida y requieren de un ajuste manual para establecer la altura de estos.
- El robot presenta problemas de estabilidad y/o vuelco debido a que los puntos de apoyo son pequeños. En la práctica, se ha observado que el robot se deslizaba en superficies de metal finas donde la adherencia es más reducida.
- La actuación de los imanes conmutados no está estandarizada. Motores con levas en los pies externos y un piñón cremallera en el interno. Lo que implica interfaces de hardware/software diferentes. Además, la implementación actual requiere de demasiadas piezas y su ajuste no es preciso.
- El robot no cuenta con una fuente de energía propia.
- El robot no provee espacio para incorporar un sensor en el pie interior.
- El robot no cuenta con un sistema modular para adaptar el robot en función de la misión o realizar cambios o reparaciones de forma rápida.

- No existe comunicación inalámbrica. Se requiere la conexión física entre el usuario y el robot para intercambiar información.
- El chasis presenta baja resistencia torsional dada su longitud.
- Algunos componentes como los micro-motores con reductora, los imanes radiales 20x5 mm N48 y actuadores lineales son caros.
- Es posible mejorar la fuerza del imán conmutable si la carcasa de metal tiene un hueco de aire que minimice las fugas de flujo magnético y así maximice la fuerza de adhesión.
- Se pueden eliminar los tornillos que soportan el imán conmutable usando una cavidad impresa en 3D que se adapte a la forma del colector.
- El software está implementado como un bucle secuencial, lo que impide aprovechar al máximo las capacidades del microprocesador y limita la escalabilidad del este.

En las siguientes secciones del trabajo, se tratarán de abordar una por una las limitaciones existentes y poner en práctica las soluciones propuestas en el siguiente prototipo. Nótese que no todas las soluciones propuestas van a poder ser llevadas a cabo en el próximo prototipo; principalmente por las limitaciones del autor. Sin embargo, sí que se mostrará el trabajo realizado hasta la fecha y posibles formas de llevarlas a cabo. Otros estudiantes pueden tomar el relevo y continuar mejorando el producto.

3 DESARROLLOS

3.1 MECÁNICA

En esta sección se realiza una revisión pormenorizada de todas las mejoras realizadas en el apartado mecánico. La siguiente imagen, muestra el resultado del prototipo virtual número 4, el cual constituye la versión definitiva.

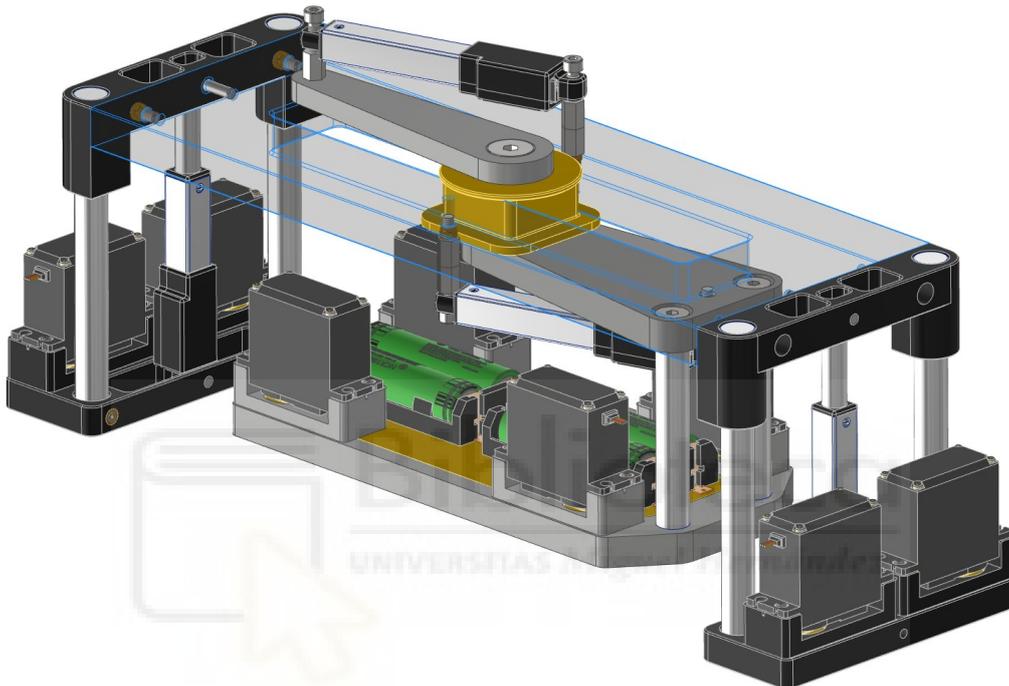


Figura 8: Modelo 3D del nuevo prototipo del Xrobin.

Aunque se ha empleado impresión 3D para la fabricación del prototipo, se ha intentado realizar un diseño que sea mecanizable. Esto implica tener en cuenta los siguientes consideraciones:

- Uso de geometrías sencillas que permitan reducir el número de amarres y el uso de mordazas blandas.
- Tratar de utilizar los mismos valores de radio interior. $R_{interior, min} \geq 3 \text{ mm}$
- Relación profundidad y diámetro de la herramienta menor que 4.
- Usar valores enteros allá donde sea posible para simplificar operaciones manuales.

Ha de tenerse en cuenta que se deben realizar algunos ajustes de diseño para el mecanizado, como por ejemplo:

- La eliminación de los alojamientos para insertos de latón embebidos en el plástico.
- El vaciado de zonas donde se pueda ahorrar material.

3.1.1 MODULARIZACIÓN

Se trata de establecer unidades de diseño independientes que se conecten entre sí a través de interfaces. De esta forma, se pueden desarrollar de forma independiente cada módulo y se facilita la integración de mejoras futuras.

Los módulos son:

- Deslizaderas y bieletas.
- Bastidor.
- Pie exterior.
- Pie interior o plataforma.
- Unidad servomotor con imán conmutable o servomódulo. Este incluye:
 - El servomotor.
 - Los imanes cilíndricos polarizados diametralmente.
 - Los colectores magnéticos.
 - Los acopladores entre servomotor y el imán superior.

Se pueden definir las interfaces importantes entre los módulos.

- Bastidor – Soporte de guías de elevación.
- Soporte de guías de elevación – Guías de elevación.
- Pie exterior – Guías de elevación.
- Actuador lineal – Pie exterior / Soporte de guías de elevación.
- Actuador lineal – Bastidor / Bieletas.
- Bastidor – Junta prismática-rotativa.
- Junta prismática-rotativa – Bieletas.
- Guías – Pie exterior.
- Servomódulo – Con la pieza soporte.
- Colectores magnéticos – Con la pieza soporte.
- Servomotor – Acoplador – Imanes.
- Pie interior – PCB.

3.1.2 SERVOMÓDULOS

Los servomódulos se encargan de activar y desactivar los imanes conmutables.

Por un lado, los micro-motores Pololu con una reducción 1:1000 también tienen un elevado coste y su movimiento es lento debido a esta reducción. Se ha empleado como sustituto un servomotor genérico MG996R. Es significativamente más grande pero mucho más barato. Respecto a las dimensiones, el reducido tamaño del motor Pololu presenta retos de integración, manejo y robustez.



Figura 9: Micro-motor Pololu con reducción 1:1000 (a) y Servomotor MG996R (b).

En la Tabla 1 se comparan algunas características de estos motores. Aunque existen incrementos en peso y volumen significativos, en el conjunto del robot estos valores no son tan abultados. Sin embargo, el precio puede ser de hasta 25€ menos por servomódulo.

Tabla 1: Características del motor Pololu vs MG996R.

Parámetro	Motor original	Motor nuevo	Diferencia	Coefficiente
Tamaño (mm)	10x12x28.5	40.7x20.2x38.5	+++	+++
Corriente en reposo sin carga	10 mA	10 mA	0	100%
Peso	10 g	55 g	+ 45 g	550%
Par máximo (6V)	5 kg·cm	11 kg·cm	+6 kg·cm	220%
Precio unitario	30 €	5 €	- 25 €	16.6 %

Por otro lado, cuando se experimentó con el prototipo inicial, se apreció que el robot tendía a deslizarse en paredes verticales cuando usaban solo dos servomódulos. Esto puede ser causado por una combinación de la acción de la gravedad más el momento de vuelco. Por esta razón se ha decidido usar 4 servomódulos en lugar de dos. Además se ha tratado de separar al máximo los imanes para evitar el vuelco.

Tras hacer una investigación sobre la disponibilidad y el precio de los imanes usados en el primer prototipo del Xrobin, se ha llegado a la conclusión de que el precio es demasiado alto y que está disponible en muy pocos proveedores. Por eso se ha decidido buscar una alternativa más económica.

Finalmente, se ha decidido reducir el grado de magnetización de los imanes de N48 a N35 y el diámetro de los imanes de 22 mm a 20 mm. Se espera que la fuerza de atracción sera un 34% inferior (por módulo), sin embargo el precio se reduce en un 85% (por módulo).

Tabla 2: Características del imán N48 vs N35.

Parámetro	Imán original	Imán nuevo	Diferencia	Coefficiente
Grado	N48 (1.4 T)	N35 (1.19 T)	-0.21 T	85 %
Diámetro	20 mm	18 mm	-2 mm	90 %
Espesor	5 mm	5 mm	0 mm	100 %
F @ 0,01 mm	7.18 kg	4.76 kg	-2.42 kg	66.3%
Precio unitario	9.31 €	1.39 €	-7.92 €	14.9%

Teniendo en cuenta que se va a duplicar el número de imanes y que se va a separar el colector magnético (Figura 1) para evitar las fugas de flujo magnético, se espera que la fuerza de sujeción normal a la superficie de contacto sea al menos 1,2 veces mayor al original. Para respaldar esto se debería realizar un ensayo para comprobar empíricamente la fuerza de atracción magnética. En la práctica, la fuerza de sujeción del nuevo prototipo se ha comprobado suficiente sobre una chapa de acero de menor a 1 mm de espesor.

Volviendo al apartado de coste, se puede estimar un ahorro considerable al considerar que se necesitan 8 servomódulos para un solo robot. El ahorro total asciende hasta al rededor de 222.24€. Desglosado:

- Motores: 8u x 30€/u = 240€ contra 8u x 5€/u = 40€, un ahorro de 200€.
- Imanes: 16u x 9.31€/u = 148.96€ contra 16u x 1.39€/u = 22.24€.

Dado que el MG996R cuenta por defecto con control de posición, a estos costes habría que añadir los costes para realizar este control con el motor Pololu.

Según esta herramienta Online [5]:

Calculate Disc

Units

North Marker
 Disabled

Diameter
 mm

Thickness
 mm

Distance
 mm

Polarity

Material

Max Working Temp

Neodymium 48 Diametral Disc Magnet
D: 20mm, T: 5mm
Br (Gauss): 14100 Material grade constant
Surface (Gauss): 3150.85
Lifting power: 7.18kg = 15.83lb
Tolerances: ±0.004"
Plating/Coating: Ni-Cu-Ni
Max Temperature: 80°C = 176°F
Magnetization Direction: (Diametral)
Made In China
Inventory USA

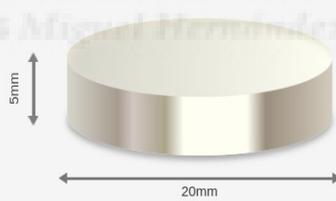


Figura 10: Características del imán original (N48).

Calculate Disc

Units

North Marker
 Disabled

Diameter
 mm

Thickness
 mm

Distance
 mm

Polarity

Material

Max Working Temp

Neodymium 35 Diametral Disc Magnet
D: 18mm, T: 5mm
Br (Gauss): 12100 Material grade constant
Surface (Gauss): 2935.90
Lifting power: 4.76kg = 10.49lb
Tolerances: ±0.004"
Plating/Coating: Ni-Cu-Ni
Max Temperature: 80°C = 176°F
Magnetization Direction: (Diametral)
Made In China
Inventory USA

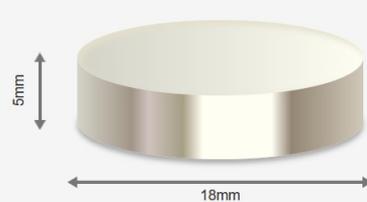


Figura 11: Características del nuevo imán (N35).

La geometría de los colectores magnéticos y la dirección de las fuerzas durante el funcionamiento evita que estas se transmitan al servomotor, ya que se transmiten directamente a la base. Lo que permite sujetar el servomotor con un tornillo de M3. Un pasador liso de 3 mm se encarga de fijar la posición.

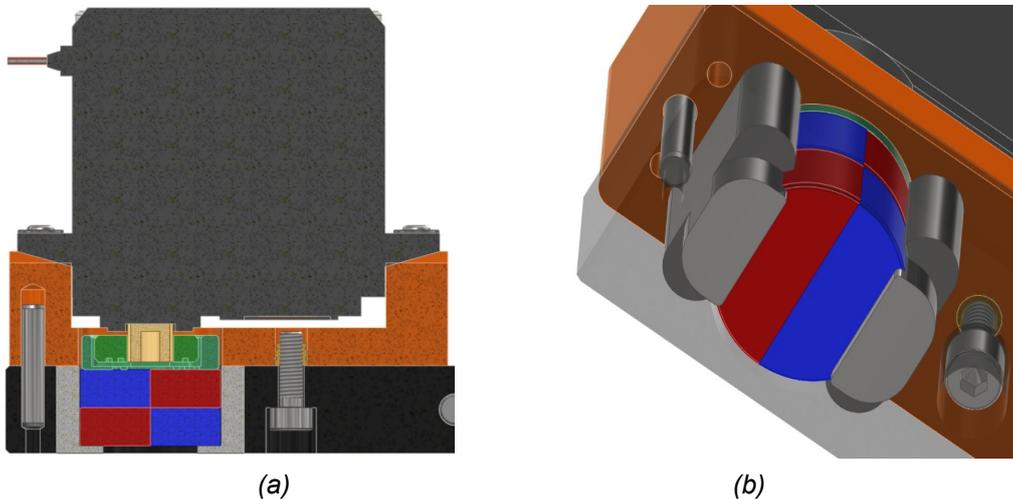


Figura 12: Vista de sección (a) y vista 3D inferior (b) de un servomódulo.

Se ha decidido hacer un acoplador poka-yoke para evitar el ensamblaje en una posición inadecuada. Los servomotores e imanes deben ser ensamblados en la siguiente configuración.

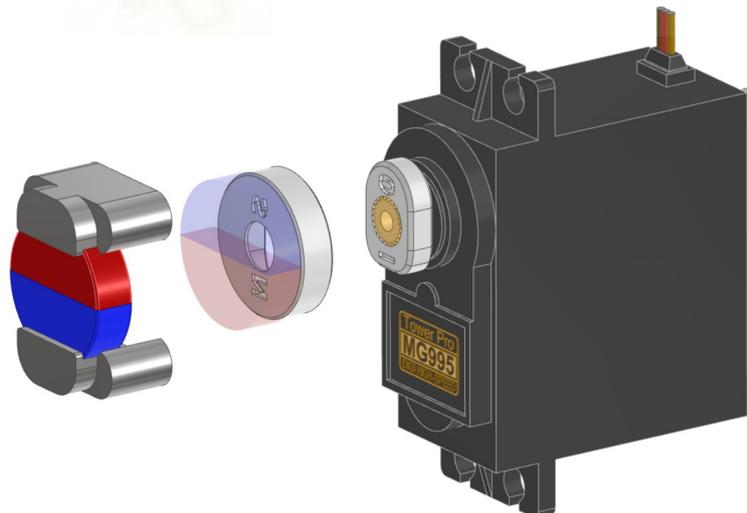


Figura 13: Vista de explosión de un servomódulo

Estas son las instrucciones para la preparación de los submódulos.

1. Poner el servomotor en 0°.
2. Ensamblar el acoplador del servomotor en la posición vertical.
3. Ensamblar el soporte del motor.

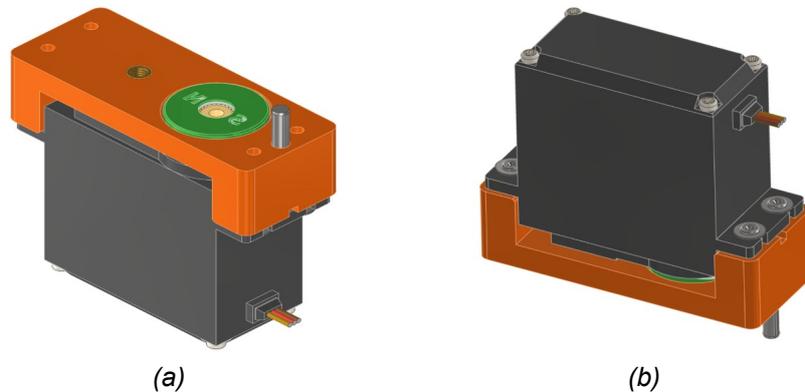


Figura 14: Vista 3D inferior (a) y superior, derecha (b) de un servomódulo.

4. Localizar el polo sur (azul) del imán superior y pegar el acoplador del imán, tratando de alinear la letra S con la mitad del imán. Se recomienda utilizar un patrón impreso en 3D para facilitar la colocación en el centro.

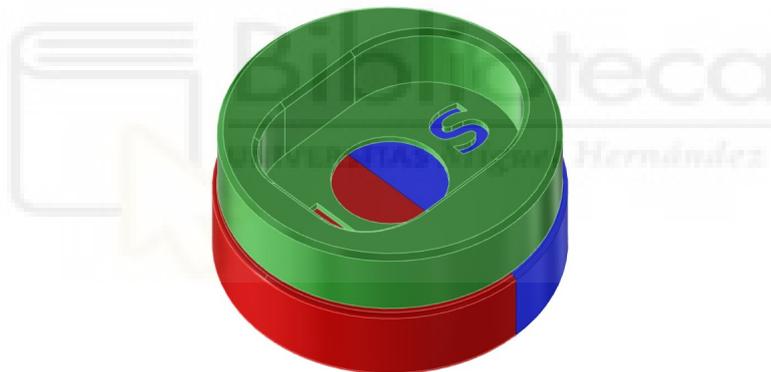


Figura 15: Vista 3D del imán superior con su acoplador.

5. Introducir los colectores magnéticos en las ranuras de la base del pie.
6. Localizar el polo norte del imán inferior e introducir el imán mirando hacia el exterior del pie (o como se indique en el CAD). Aunque la propia fricción debería mantener el imán en posición, es recomendable añadir una gota de pegamento. También es recomendable marcar con rotulador permanente una letra N en el colector magnético para indicar la posición del Norte (rojo).

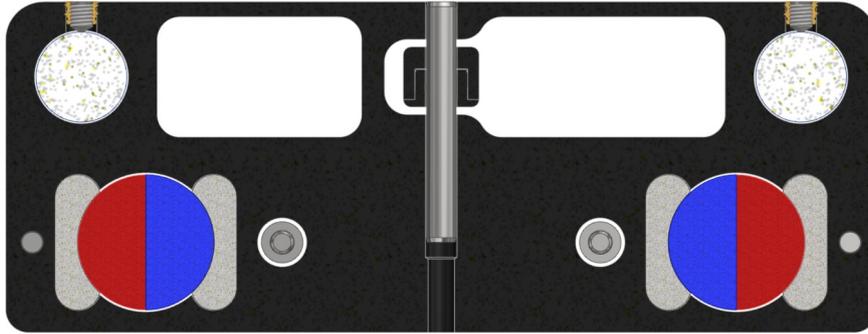


Figura 16: Vista seccionada de la planta del pie exterior.

7. Colocar el imán superior sobre el inferior en la posición desactivada.

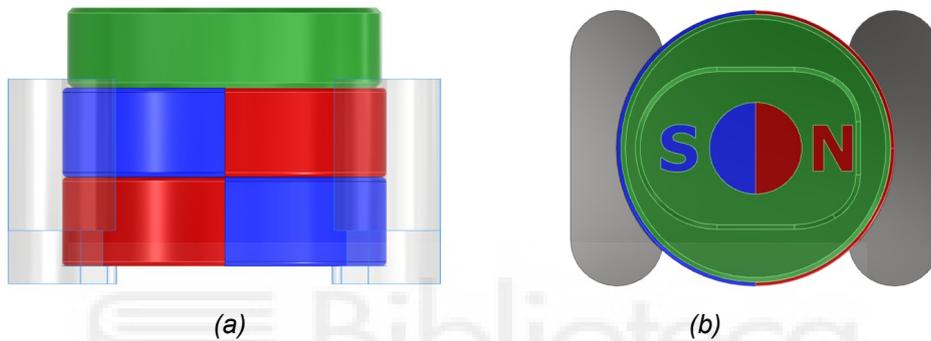


Figura 17: Vista lateral (a) y superior (b) de los imanes, colectores magnéticos y un acoplador.

8. El paso final consiste en posicionar, insertar y acoplar el servomotor con soporte sobre el imán superior y atornillar el conjunto desde abajo.

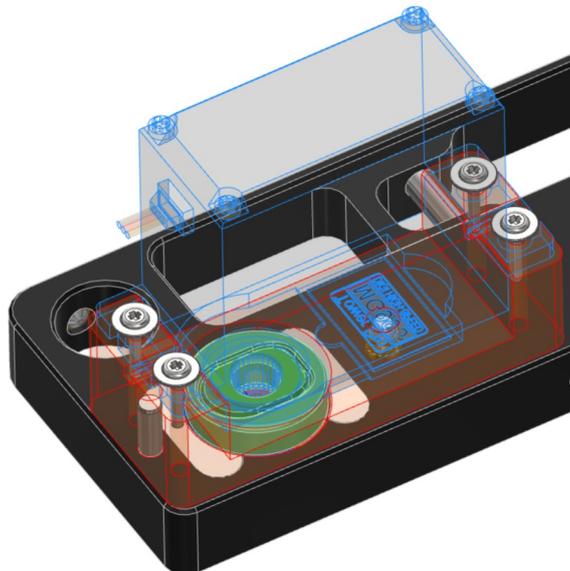


Figura 18: Vista superior de un servomódulo montado sobre un pie exterior.

3.1.3 BASTIDOR

El bastidor es una placa rectangular con una ranura que sirve como deslizadera de la junta prismática-rotativa. Desde la vista superior podemos ver dos agujeros, que servirán de apoyo a los actuadores lineales de traslación.

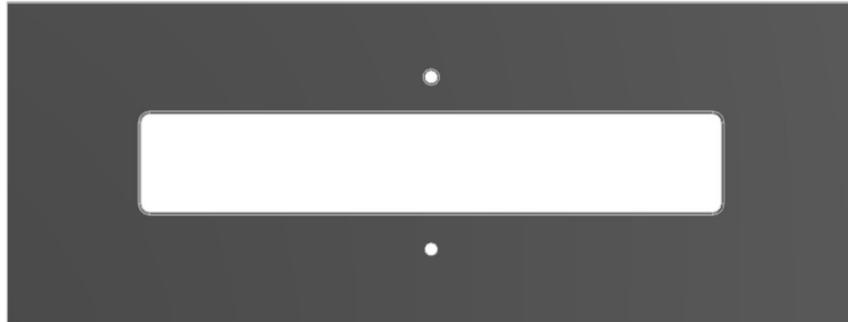


Figura 19: Vista superior del bastidor.

Desde la vista lateral se pueden ver la interfaz con entre el pie exterior y el bastidor, que consta de:

- El agujero central es un pasador liso M4 que cumple la función de posicionar el pie exterior con respecto al bastidor.
- Los agujeros exteriores son 2 alojamientos para insertos de latón de M4.



Figura 20: Vista lateral del bastidor.

En vista de sección:

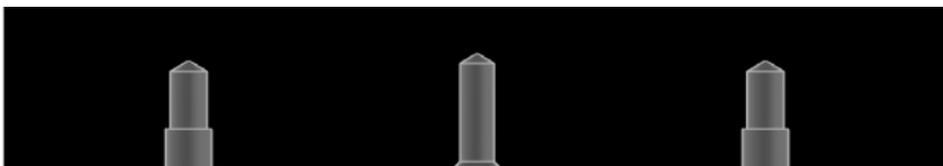


Figura 21: Vista seccionada de los agujeros en los laterales del bastidor.

3.1.4 MECANISMO TIJERA

La geometría de este mecanismo viene dada por el diseño inicial en artículo académico [3].

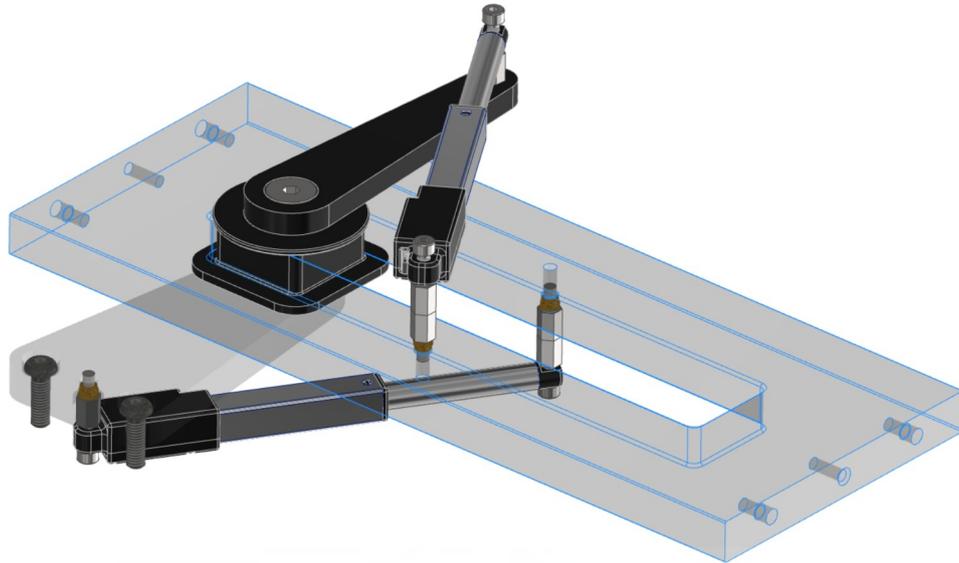


Figura 22: Vista 3D del mecanismo de tijera.

La orientación de los actuadores no es al azar, ya que se está tratando de minimizar la longitud de los cables. Todos estos parten del pie interior.

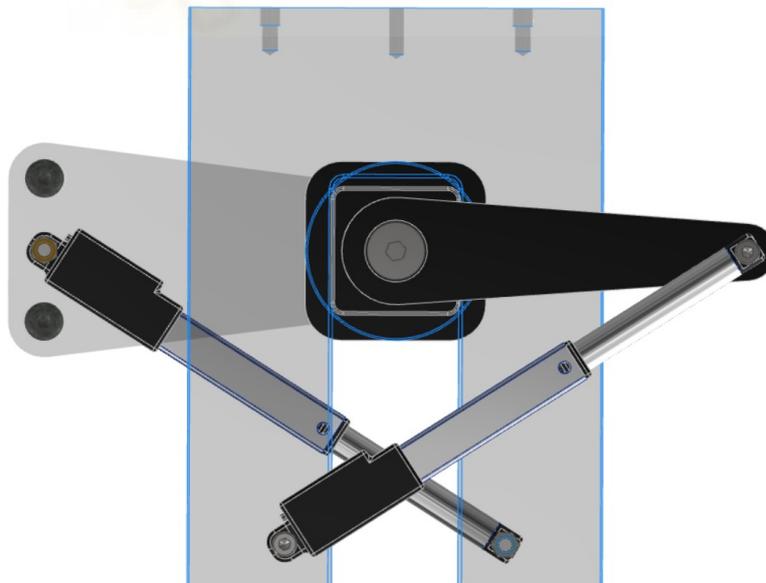


Figura 23: Vista superior del mecanismo de tijera.

La mejora del mecanismo consiste en el uso de componentes estándar como tornillos y separadores hexagonales para materializar los apoyos de los actuadores lineales de traslación.

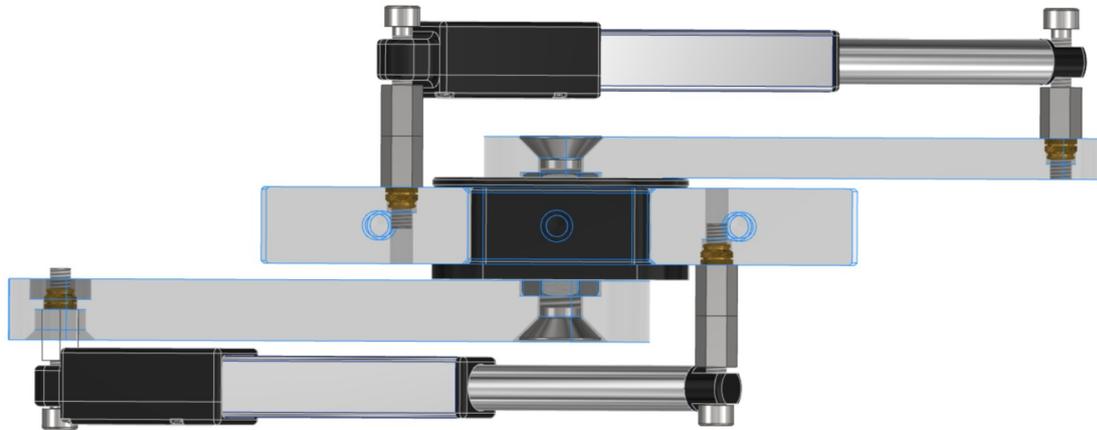


Figura 24: Vista lateral del mecanismo de tijera.

Al atornillar el tornillo hasta el fondo del separador hexagonal, se consigue una junta rotativa muy sencilla con solo dos componentes.

Adicionalmente se pueden usar elementos auxiliares para eliminar el juego entre las piezas y hacer que el actuador quede perfectamente horizontal.

En caso de mecanizar el diseño, se podría ajustar el separador y tornillo para evitar el uso de estos elementos adicionales.

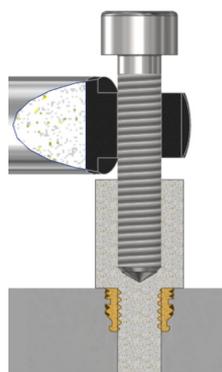


Figura 25: Vista seccionada de un apoyo de un actuador lineal

Especialmente en la parte inferior, es necesario considerar el juego entre las diferentes piezas:

- 1.5 mm entre el bastidor y el actuador lineales.
- 4.5 mm entre el servomotor y el actuador lineales.

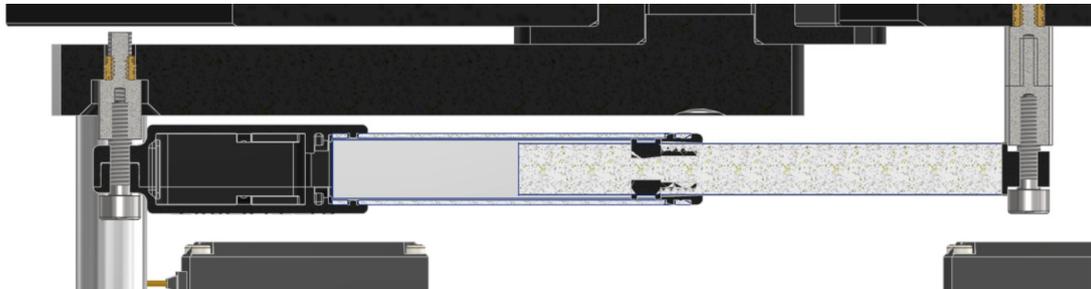


Figura 26: Vista lateral seccionada del mecanismo de tijera para comprobar el juego entre los distintos componentes

Si se producen problemas de interferencia, se puede cortar el tornillo o introducir unas arandelas en la base de los separadores hexagonales.

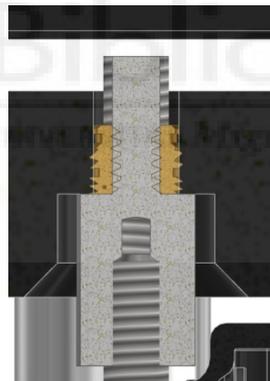


Figura 27: Vista detalle seccionada del apoyo del actuador inferior de traslación.

3.1.5 JUNTA PRISMÁTICA-ROTATIVA

Este conjunto de piezas es uno de los componentes clave, ya que debe cumplir varios requisitos:

- Función de deslizamiento y rotación en un mismo conjunto.
- Juego reducido.
- Funcionamiento fluido.
- Resistencia a la flexión.
- Transmisión de par.

Se han coloreado los dos subconjuntos que forman esta junta:

- El balancín, formado por las bieletas (morado), el casquillo separador (rojo), la tuerca larga de M8 y los tornillos avellanados de M8.
- La deslizadera, formada por el bloque central (verde) y la arandela (verde claro).

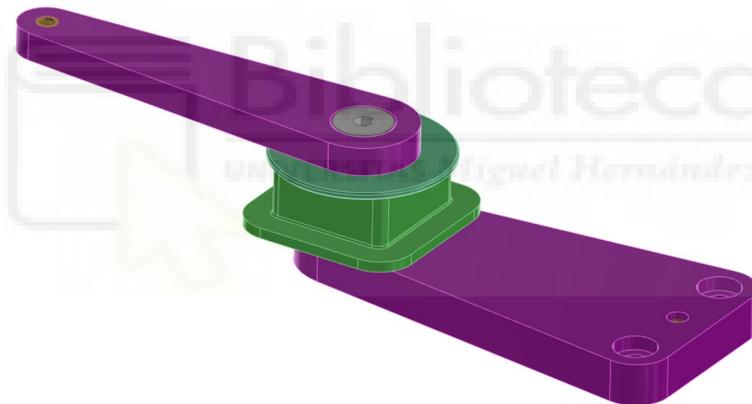


Figura 28: Vista 3D del balancín y la junta prismática-rotativa

En esta vista de sección se pueden ver todos los elementos involucrados. Nótese que primero se debe colocar la deslizadera y una mitad del balancín para luego cerrar el conjunto con la otra mitad del balancín.

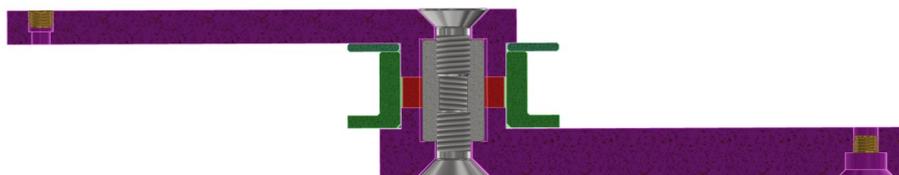
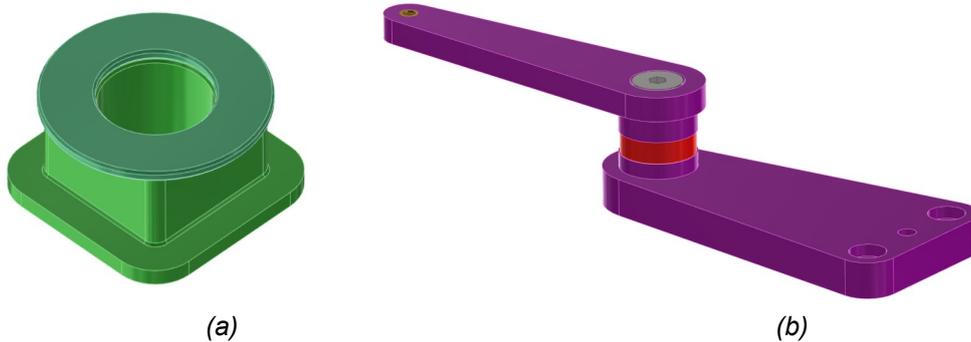


Figura 29: Vista seccionada del balancín y la junta prismática-rotativa

Ahora se describirá en detalle los rasgos característicos de cada conjunto y la interacción entre ellos y el bastidor.



(a) (b)
Figura 30: Vista 3D de la justa prismática-rotativa (a) y vista 3D del balancín (b).

La función de rotación depende exclusivamente de la interfaz cilíndrica entre las bieletas del balancín (morado) y el bloque central (verde). Se ha evitado el contacto con el casquillo separador (rojo) para mantener el número de superficies de contacto en dos.



(a) (b)
Figura 31: Detalle 1 (a) y 2 (b) de la vista seccionada del balancín y la justa prismática-rotativa.

La unión de la bieleta superior con la inferior se realiza a través de una tuerca larga de M8. La interfaz hexagonal permite la transmisión de par y los tornillos avellanados de M8 mantienen unido el conjunto, haciendo que se comporte como un único sólido. Se han empleado tornillos avellanados para maximizar la cantidad de material en las bieletas y que no sobresalgan.

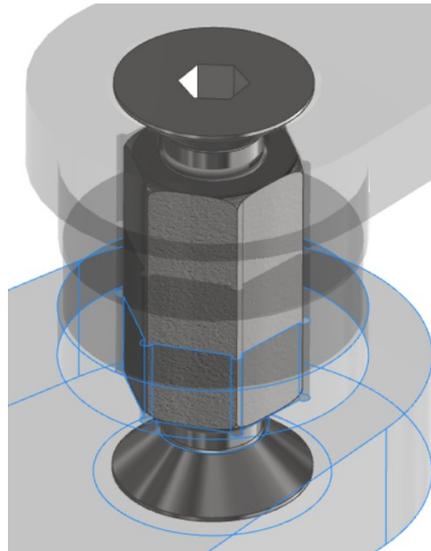


Figura 32: Vista 3D de la interfaz hexagonal del balancín.

En las interfaces hexagonales, se han provisto unos desahogos en los vértices para facilitar el ensamblaje y permitir un mejor ajuste sobre las caras planas. Este tipo de diseño es ampliamente utilizado en los vasos hexagonales de carraca.

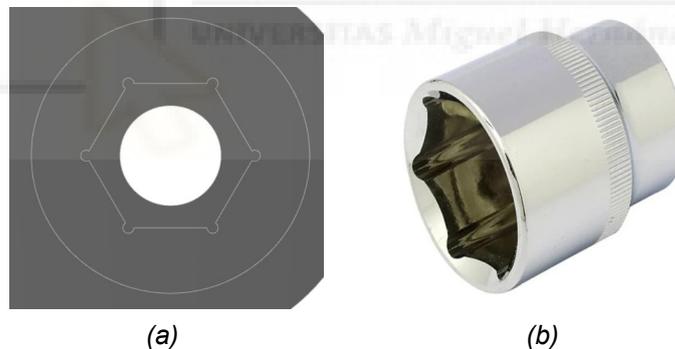


Figura 33: Vista superior de la interfaz hexagonal del balancín, izquierda (a). Vaso hexagonal de carraca, derecha (b).

Nótese que el conjunto se cierra a través de la cadena “tornillo, bieleta, anillo, bieleta, tornillo”. Lo que deja la tuerca central libre en la dirección axial. Esto permite realizar ajustes cambiando únicamente el anillo central.

La arandela también puede ser usada como segundo elemento de ajuste.

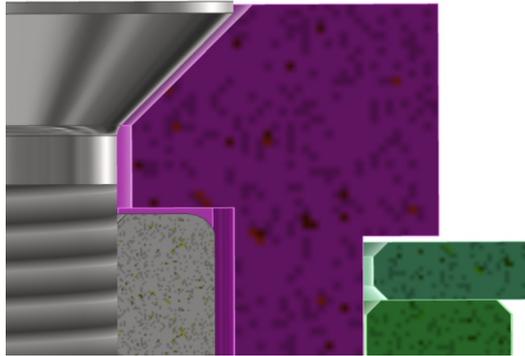


Figura 34: Detalle 3 de la vista seccionada del balancín y la junta prismática-rotativa

La deslizadera está restringida en la dirección transversal por la interfaz entre la deslizadera (verde) y el bastidor (gris transparente). No hay ningún elemento de ajuste.



Figura 35: Vista seccionada de la interfaz entre la junta prismática-rotativa con el bastidor.

Para evitar el vuelco, se ha provisto a la deslizadera de unas superficies planas que contactan la parte inferior del bastidor y de una arandela en la parte superior. Esta permite ajustar el juego axial cambiando su espesor.



Figura 36: Detalle 4 de la vista seccionada del balancín y la junta prismática-rotativa

Nótese que en la parte inferior se producen mayores esfuerzos que en la zona superior, dado que el lado inferior aguanta todo el robot y el superior, sólo se encarga de transmitir la fuerza de uno de los actuadores lineales.

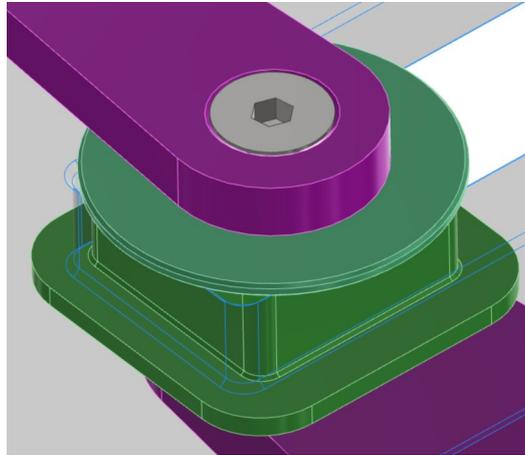


Figura 37: Vista 3D de la interfaz entre el bastidor y la junta prismática-rotativa en los extremos del recorrido.

Por último, cabe destacar que los chaflanes son indispensables para evitar el contacto con los radios interiores producidos por el mecanizado o la impresión 3D. Aunque es sutil, es un elemento clave para obtener un funcionamiento fluido.

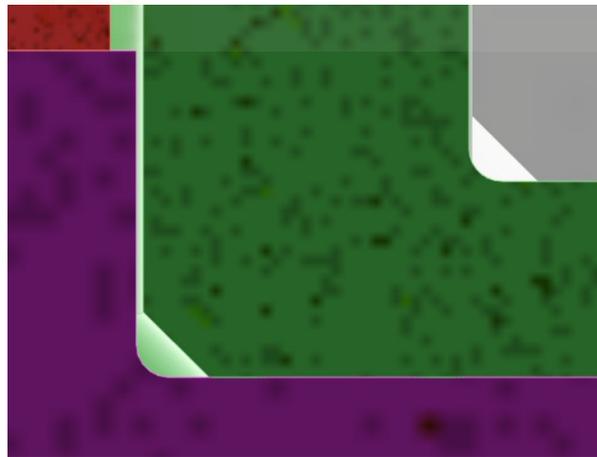


Figura 38: Detalle 5 de la vista seccionada del balancín y la junta prismática-rotativa

3.1.6 PIES EXTERIORES

Los pies exteriores se encargan de realizar los movimientos de elevación y de adhesión del bastidor. Se componen de:

- Un acoplador con el chasis.
- La base del pie.
- 2 servomódulos.
- Un actuador lineal de elevación.
- Elementos de sujeción.

En esta vista superior se pueden ver cómo de compacto es el ensamblaje. Existe un pequeño espacio entre el acoplador al bastidor y los servomódulos para que no interfieran durante el ensamblaje.

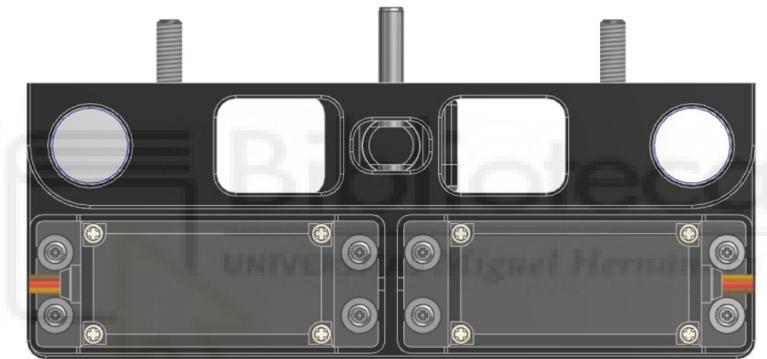


Figura 39: Vista superior del pie exterior.

En la base, se puede ver la interfaz con los colectores magnéticos, servomódulos, guías y el apoyo del actuador lineal. La asimetría en el soporte del actuador lineal sirve para evitar la interferencia con el cable del actuador.

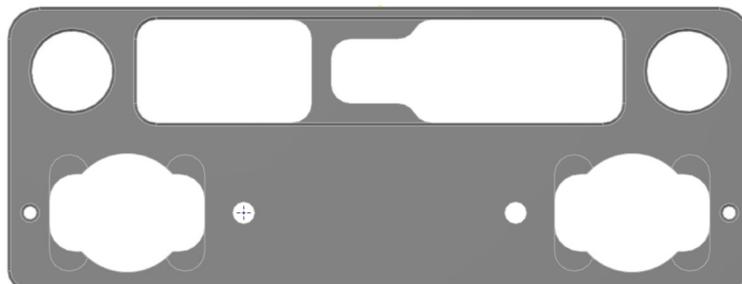


Figura 40: Vista superior de la base del pie exterior.

En esta sección de la vista frontal se puede apreciar:

- El actuador lineal está soportado por pasadores lisos de 4mm.
- La sujeción inferior de las guías se realiza mediante un ajuste fino entre la guía y el soporte combinado con un tornillo de cabeza perdida.
- El apoyo deslizante de la guía está formado por dos superficies cilíndricas estrechas separadas entre sí. Esto permite que el la fricción sea mínima mientras se mantiene la resistencia al momento flector.

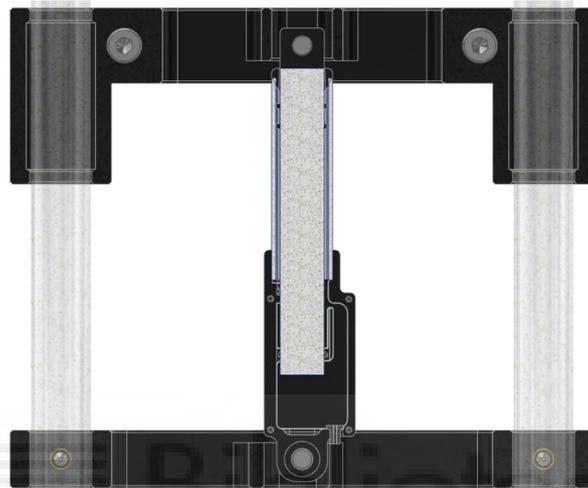


Figura 41: Vista frontal seccionada del pie exterior.

La vista seccionada permite ver la disposición de los servomódulos, colocados los más cerca posible de los bordes para aumentar la resistencia al vuelco.

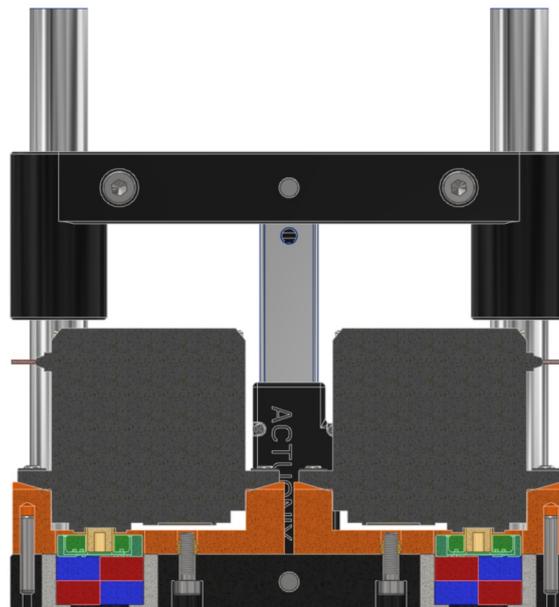


Figura 42: Vista frontal seccionada del pie exterior, incluyendo los servomódulos.

3.1.7 PIE INTERIOR O PLATAFORMA

El pie interior o plataforma, es el efector final del robot, donde se encuentran los componentes más importantes. Su función estructural es la de alojar los módulos de electrónica y mantener el robot pegado cuando el pie exterior está recogido.

Está compuesto por:

- El bastidor del pie interior.
- 4 servomódulos.
- 2 Pilares.
- Elementos de sujeción.
- Componentes electrónicos (Baterías, PCB principal y sub-módulos).

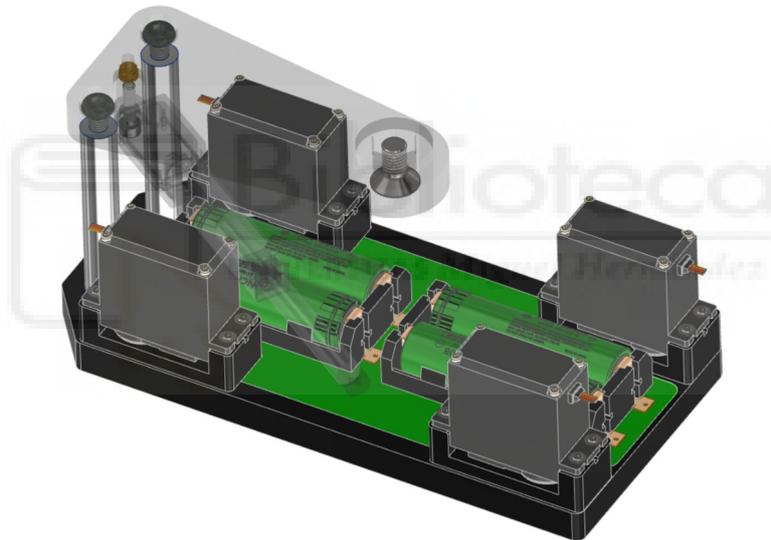


Figura 43: Vista 3D del pie interior.

Se han colocado los motores lo más separados posible para maximizar el espacio en él disponible para otros componentes y mejorar la resistencia al vuelco. En este robot, se han colocado 4 baterías de litio 18650 aunque es posible elegir otra configuración.

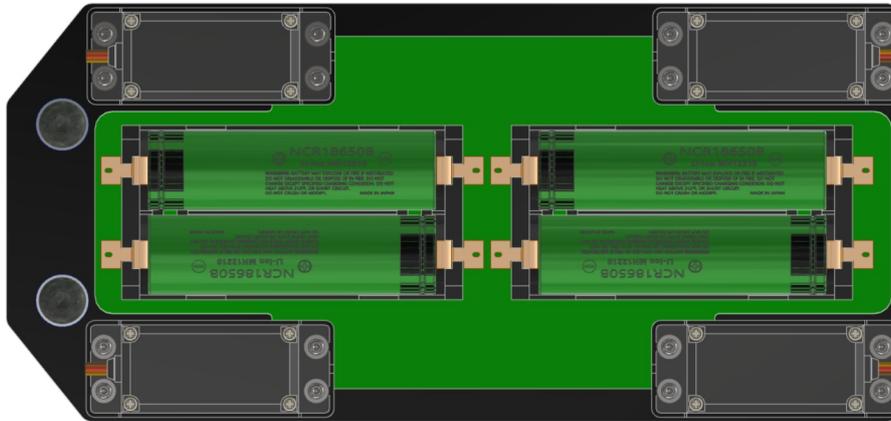


Figura 44: Vista superior del pie interior.

El espacio inferior del pie está pensado para colocar el módulo de computación, el sensor de misión y el módulo de control. Se ha colocado un Raspberry Pi Compute Module 4 para dar una idea del espacio disponible.

Como sensor de misión se entiende cualquier sensor que vaya a ser montado en el robot para una tarea específica. Por ejemplo, un sensor de ultrasonidos para detectar defectos internos en la pared metálica.

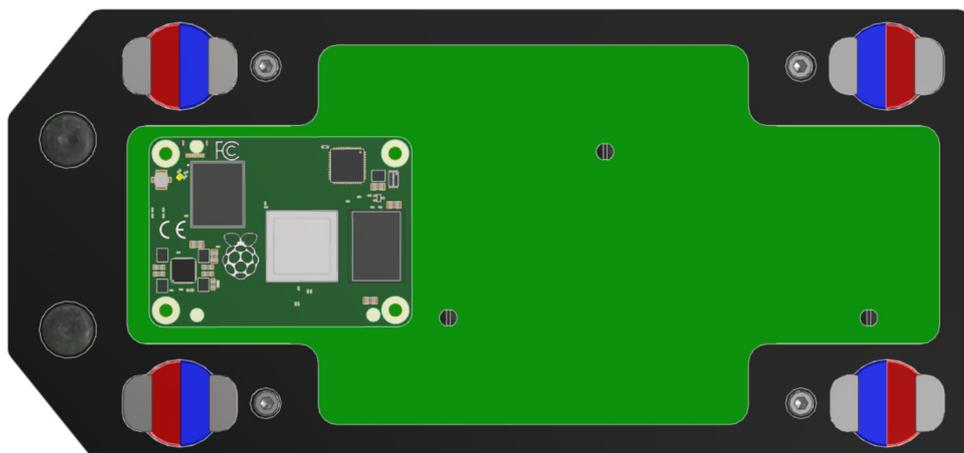


Figura 45: Vista inferior del pie interior.

La PCB se mantiene en posición mediante una cajera en la pieza con la misma forma de ésta y el soporte del servomotor como se puede ver en las siguientes imágenes.

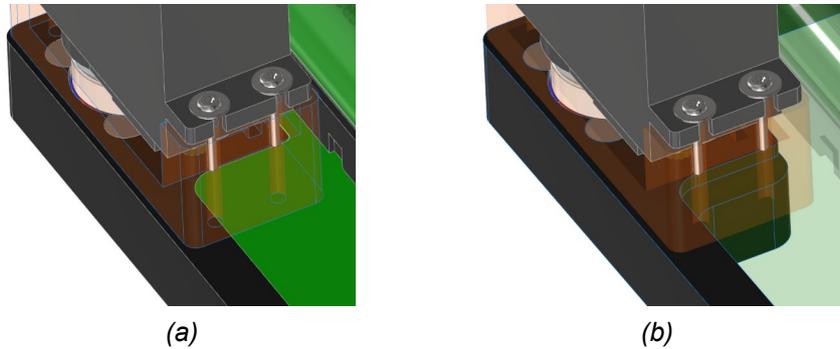


Figura 46: Vista (a) y (b) de la interfaz entre la PCB, el servomódulo y el pie interior.

La conexión con los pilares se realiza a través de unos tornillos de cabeza redondeada M6 para aumentar la cantidad de material.



Figura 47: Vista seccionada entre la interfaz del pie interior y los pilares.

3.2 ELECTRÓNICA

Al igual que la mecánica y el software, el diseño electrónico está concebido para ser modular. Se pretende dividir las funciones en 5 bloques:

- Electrónica de potencia.
- Electrónica de control.
- Módulo de computación y comunicaciones.
- Actuadores y Sensores.
- Sensor de misión.

En el diagrama de bloques podemos ver las relaciones entre los diferentes módulos:

- El módulo de potencia debe disponer de PD para disponer de carga rápida con con cargador USB-C.
- La comunicación entre los módulos se realiza mediante interfaces serie, ya sea, USB, UART o I2C.
- El módulo de potencia se encarga de generar los voltajes necesarios para el resto de componentes.
- El MCU se encarga del hardware a bajo nivel.
- El módulo de computación es el núcleo del robot y se encarga de las comunicaciones con el exterior y el sensor de misión. También puede hacer de MCU.

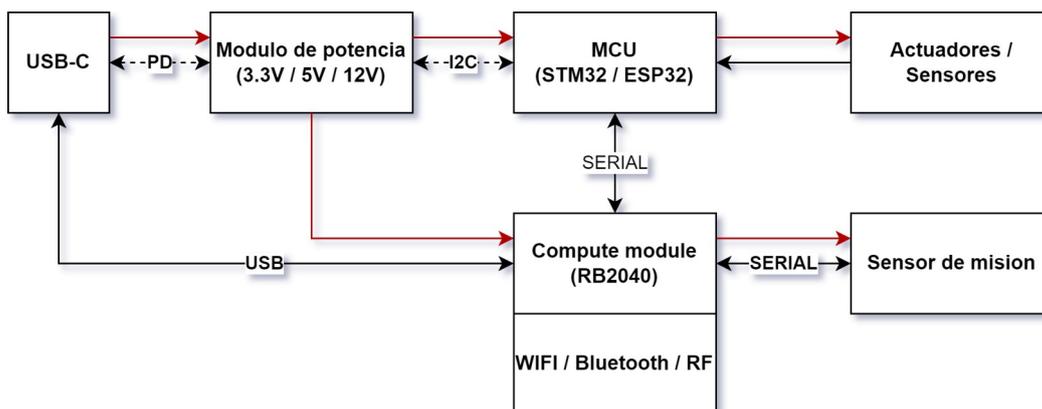


Figura 48: Diagrama de bloques del Xrobin.

3.2.1 ACTUADORES

Actuadores lineales

- Actuonix LS50: Actuador lineal con 50 mm de recorrido empleado para accionar los mecanismos de traslación. Motor DC12V. El motor se detiene al llegar al final del recorrido.
- Actuonix LS30: Actuador lineal con 30 mm de recorrido empleado para accionar los mecanismos de elevación.



Figura 49: Actuador lineal LS50 de Actuonix.

Este es un extracto del datasheet del dispositivo donde se pueden ver sus características principales:

L12 Specifications				
Gearing Option	50:1	100:1	210:1	
Peak Power Point	17N @ 14mm/s	31N @ 7mm/s	62N @ 3.2mm/s	
Peak Efficiency Point	10N @ 19mm/s	17N @ 10mm/s	36N @ 4.5mm/s	
Max Speed (no load)	25mm/s		13mm/s	
Max Force (lifted)	22N		42N	
Back Drive Force (static)	12N		22N	
Stroke Option	10 mm	30mm	50mm	100mm
Mass	28 g	34 g	40 g	56 g
Repeatability (-I, -R, -P&LAC)	±0.1 mm	±0.2 mm	±0.3 mm	±0.5 mm
Max Side Load (extended)	50N	40N	30N	15N
Closed Length (hole to hole)	62mm	82mm	102mm	152mm
Potentiometer (-I, -R, -P)	1kΩ±50%	3kΩ±50%	6kΩ±50%	11kΩ±50%
Voltage Option	6VDC			12VDC
Max Input Voltage	7.5V			13.5V
Stall Current	460mA			246mA
Standby Current (-I/-R)	7.2mA			3.3mA
Operating Temperature	-10°C to +50°C			
Potentiometer Linearity	Less than 2.00%			
Max Duty Cycle	20 %			
Audible Noise	55dB @ 45cm			
Ingress Protection	IP-54			
Mechanical Backlash	0.2mm			
Limit Switches (-S)	Max. Current Leakage: 8uA			
Maximum Static Force	200N			

Figura 50: Extracto del datasheet del LS50.

Servomotor

Un servomotor MG996R (Figura 9.b) con un rango de movimiento entre 0° y 180° encargados de activar/desactivar los imanes conmutados. Controlado por una señal PWM de 50Hz con un duty cycle de 1ms para 0° y 2 ms para 180°. Figura 31.

Las razones por las que se ha seleccionado este servomotor son las siguientes:

- La simplicidad del control.
- La reducción del número de componentes (motor Pololu con reductor 1000:1, leva, final de carrera, cables).
- El precio: estos motores están disponibles por 4 €/unidad.
- El par resistente con el motor desactivado es mayor que el par que ejercen los imanes para desactivarse.

Specifications

- Weight: 55 g
- Dimension: 40.7 x 19.7 x 42.9 mm approx.
- Stall torque: 9.4 kgf·cm (4.8 V), 11 kgf·cm (6 V)
- Operating speed: 0.17 s/60° (4.8 V), 0.14 s/60° (6 V)
- Operating voltage: 4.8 V a 7.2 V
- Running Current 500 mA – 900 mA (6V)
- Stall Current 2.5 A (6V)
- Dead band width: 5 μs
- Stable and shock proof double ball bearing design
- Temperature range: 0 °C – 55 °C

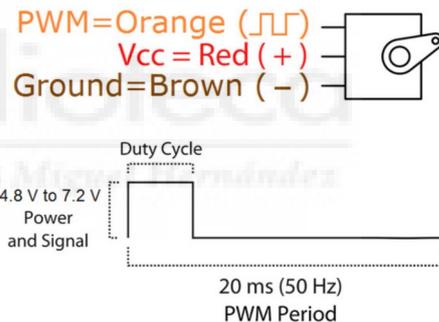


Figura 51: Extracto del datasheet del MG996R.

Existen alternativas como el servomotor SG-90, de menor tamaño y peso, sin embargo, el par resistente con el motor desactivado no es suficiente para mantener el imán activado. Tener el motor activado continuamente consumiría energía innecesariamente.

3.2.2 ELECTRÓNICA DE POTENCIA

Este módulo se encarga de la gestión de la energía dentro de la placa. Lo que consiste en: la gestión de las baterías y la conversión de voltajes para alimentar a los diferentes componentes del robot.

El robot contiene espacio para 4 baterías de litio 18650 en serie, que suman una capacidad de 2.6 Ah a 14.8 V nominales, aproximadamente 38.5 Wh.

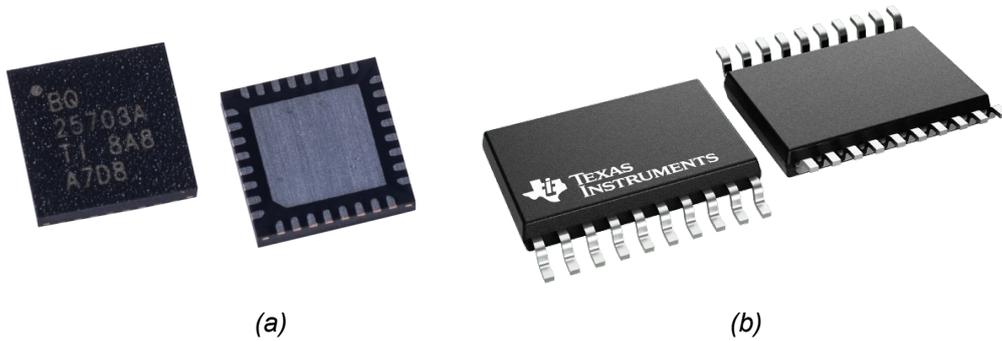
Se necesitan varios nivel de voltajes:

- 12V para los actuadores lineales.
- 5V para los servomotores, la electrónica de control y de procesado.
- 3.3V pueden ser requeridos para alimentar el microcontrolador.

Algunos microcontroladores o placas de desarrollo ya incluyen un regulador lineal de 5V a 3.3V. Si no se sobrepasa la capacidad de salida, puede ser una opción a tener en cuenta para evitar el uso de un conversor de corriente adicional.

La gestión de la carga está a cargo de dos circuitos integrados de Texas Instruments.

- BQ25703A: Un PMIC, que realiza las funciones cargador y controlador de administración de energía de batería para dispositivos portátiles, compatible con baterías de 1 a 4 celdas de ión de litio o polímero de litio. Ofrece carga rápida mediante USB y admite entradas de hasta 26 V. Además, integra características como la detección de tipo de adaptador, la protección térmica y la optimización de la eficiencia de carga. Se presenta en un encapsulado VQFN32.
- BQ76920PW: Un BMS es un monitor de batería de 3 a 5 celdas diseñado para sistemas de baterías de celdas de ión de litio o polímero de litio. Ofrece funciones de protección como sobrevoltaje, subtensión y monitoreo de temperatura, junto con comunicación I2C para la gestión del sistema de batería. Se presenta en un encapsulado TSSOP-20.



(a) (b)
Figura 52: BQ25703A (a) y EBQ76920PW (b).

Además de estos componentes claves, se debe proporcionar la electrónica de soporte para el correcto funcionamiento de estos. Se hace hincapié en la capacidad de los IC para comunicarse con el Módulo de computación ya que permite obtener información a tiempo real del estado del robot.

La conversión de corriente corre a cargo de unos convertidores conmutados de la marca Traco:

- TSR 1-2465: un regulador de conmutación DC-DC de 1 A que convierte un voltaje de entrada de 6,5 a 36 V a una salida fija de 5 V. Ofrece alta eficiencia de hasta el 96%, bajo ruido de salida, y es ideal para aplicaciones que requieren una alternativa eficiente a los reguladores lineales como alimentación de sensores, microcontroladores, y módulos de comunicación. Encapsulado SIP-3.
- TSR 1-2433E: un regulador de conmutación DC-DC de 1 A que convierte un voltaje de entrada de 6,5 a 36 V a una salida fija de 3,3 V. Ofrece alta eficiencia de hasta el 94%, bajo consumo de energía, y una baja disipación térmica, lo que lo hace adecuado para aplicaciones en las que se requiere una fuente de alimentación estable y eficiente para dispositivos electrónicos sensibles. Encapsulado SIP-3.



Figura 53: Convertidor Traco TSR.

Cabe la posibilidad de usar módulos prefabricados con convertidores conmutados, diseñar los convertidores desde cero o incluso reguladores lineales para aquellos componentes que no requieran un consumo eléctrico elevado.



Figura 54: Módulo convertidor de voltaje genérico.

3.2.3 ELECTRÓNICA DE CONTROL

Con electrónica de control se hace referencia al conjunto de componentes encargados únicamente de controlar el movimiento del robot.

El módulo se compone de:

- Un microcontrolador (por ejemplo STM32F103C8T6).
- Finales de carrera: mecánicos, ópticos, magnéticos, control de corriente, etc.
- Puentes H (por ejemplo L293D).

MCU

El microprocesador se encargará de las funciones básicas tales como activar o desactivar señales, generar PWM, etc. Además, tendrá conexión directa con un microprocesador, el PMIC y el BMS.

Se propone el uso de una bluepill con un procesador STM32F103x, porque tiene muchas opciones de conectividad (UART, I2C, USB, Timers, PWM, GPIO, etc.), es compacto y puede ser programado usando STM32CubeIDE, lo que simplifica significativamente la programación y la accesibilidad.

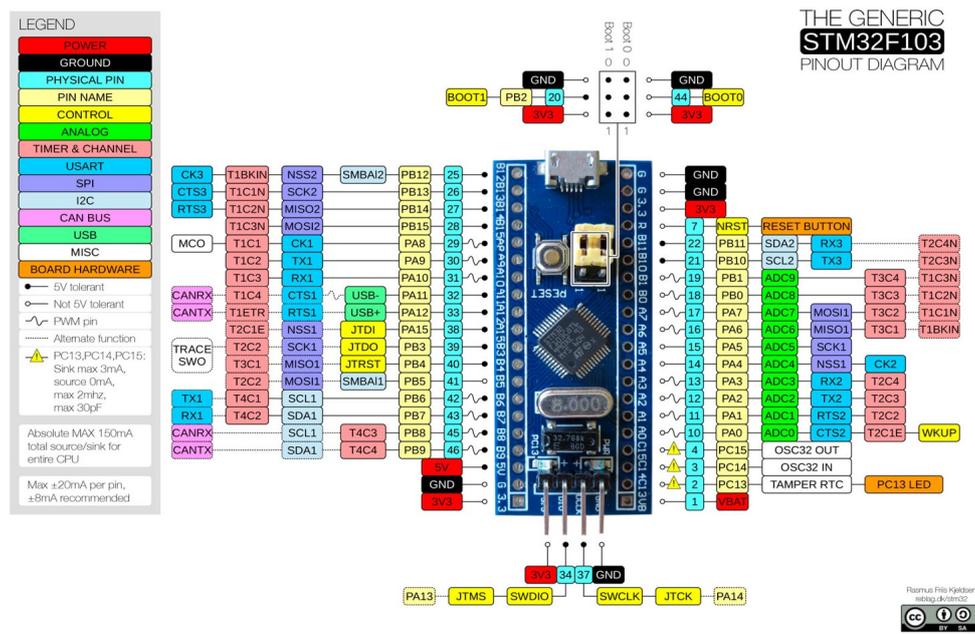


Figura 55: Diagrama de pines de una placa Bluepill con un procesador STM32F103.

Existen otras opciones como las placas ESP32 con módulos de comunicación Wifi o Bluetooth integrados. Puede ser una opción interesante si se quiere eliminar el Módulo de computación y mantener la comunicación a distancia.

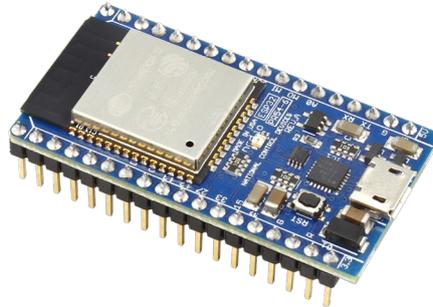


Figura 56: Placa ESP32 con módulo Wifi y Bluetooth integrado.

Control de los actuadores lineales

El control de 2 actuadores lineales independientes se puede realizar usando varias salidas digitales del microcontrolador y un puente H cuádruple. Por ejemplo, un circuito integrado como el L293D. Se presenta en encapsulados DIP-16 o SOIC-16.

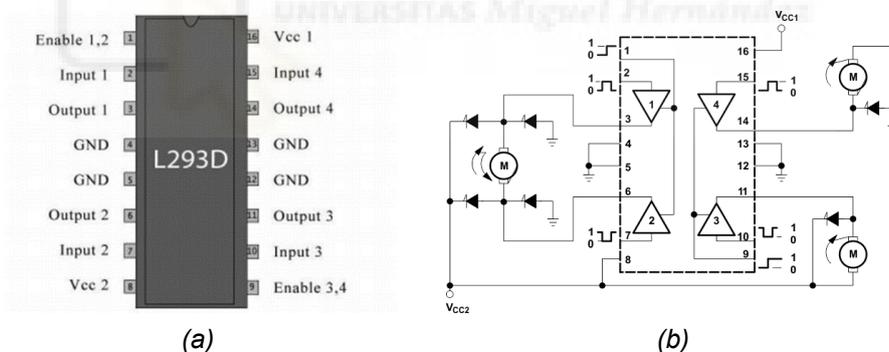


Figura 57: Pines L293D (a) y Esquemático L293D (b).

Cada canal del L293D tiene un límite de corriente de 600mA. Casi 3 veces por encima de la corriente máxima que los actuadores lineales pueden consumir 246 mA. Esto permitiría conectar 2 motores a un mismo canal, por ejemplo, los motores de elevación; sin embargo, es mejor mantener un control independiente.

6.1 Absolute Maximum Ratings

over operating free-air temperature range (unless otherwise noted)⁽¹⁾

	MIN	MAX	UNIT
Supply voltage, V_{CC1} ⁽²⁾		36	V
Output supply voltage, V_{CC2}		36	V
Input voltage, V_I		7	V
Output voltage, V_O	-3	$V_{CC2} + 3$	V
Peak output current, I_O (nonrepetitive, $t \leq 5$ ms): L293	-2	2	A
Peak output current, I_O (nonrepetitive, $t \leq 100$ μ s): L293D	-1.2	1.2	A
Continuous output current, I_O : L293	-1	1	A
Continuous output current, I_O : L293D	-600	600	mA
Maximum junction temperature, T_J		150	$^{\circ}$ C
Storage temperature, T_{stg}	-65	150	$^{\circ}$ C

Figura 58: Extracto del datasheet del L293D.

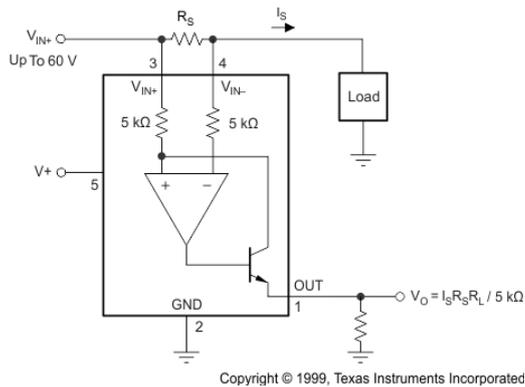
Finales de carrera

Para detectar el contacto de los pies del robot durante los movimientos de elevación, es necesario detectar cuándo se produce el contacto entre las superficies de contacto.

La pregunta es cómo asegurar de forma robusta que el robot está realizando el contacto. Es posible detectar el contacto usando finales de carrera normales y ópticos; sin embargo, esto implica tener dos componentes adicionales, cables, adaptaciones en las piezas para montarlos y elementos de ajuste. En el caso de los finales de carrera normales, es necesario que el sensor se dispare antes del contacto del pie. En los ópticos, puede haber problemas con la reflectividad de la superficie o la suciedad.

La alternativa sería medir la corriente consumida por los actuadores lineales con una resistencia Shunt. El robot debe estar pegado mientras el robot realiza un movimiento de elevación. Esto implica que cuando el pie móvil contacte con la superficie, intentará hacer fuerza para despegar el robot. Esa fuerza debe traducirse en un aumento del consumo de corriente, que puede ser detectado con una resistencia Shunt.

Se puede usar el INA138 (SOT-23-5), un amplificador de detección de corriente de ganancia variable de 2.7 a 36 V y 800 kHz.



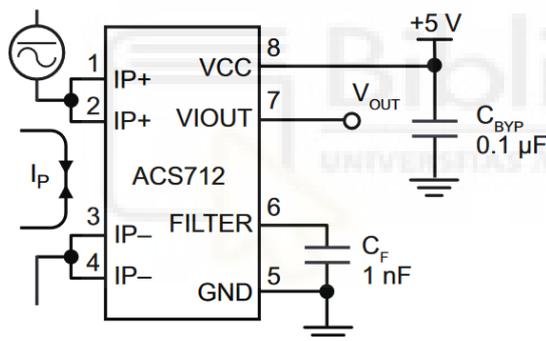
(a)



(b)

Figura 59: Aplicación típica (a) y encapsulado (b) INA138.

También sería posible usar un sensor de corriente por efecto hall como el ACS712. La variante más adecuada para el robot sería el ACS712ELCTR-05B-T. Este IC funciona a 5 Vcc y puede medir una corriente entre ± 5 A aunque su resolución quizá sea demasiado baja para esta aplicación.



(a)



(b)

Figura 60: Aplicación típica (a) y encapsulado (b) ACS712TELC-05B.

Dado que se pretende que el robot accione los actuadores de forma secuencial, sería posible colocar uno de estos sensores a la cabecera de la alimentación de todos los actuadores y ajustar los algoritmos de medida y control en función del actuador que se esté activando. De esta forma, se pueden ahorrar componentes y entradas del microcontrolador.

3.2.4 MÓDULO DE COMPUTACIÓN Y COMUNICACIONES

Este modulo consiste en el uso de un microprocesador como el que incorporan productos como las Raspberry Pi Pico o un Raspberry Pi Compute Module.

El microprocesador tendrá conexión directa con el microcontrolador de la electrónica de control, el cual se encargará de gestionar el hardware y suministrar información de bajo nivel.

La comunicación se puede llevar a cabo mediante varios protocolos. UART, USB, CAN, entre otros.

Respecto al apartado de comunicaciones, el robot debe tener la opción de poder conectarse a él de forma física o remota.

La conexión física debe realizarme mediante un conector USB-C.

Entre las opciones de conexión remota, se contemplan:

- Módulo Bluetooth o Wifi para realizar una conexión directa a corta distancia con el robot. Puede venir integrado en el Módulo de computación o de control.
- Radioenlace (digital o analógico) para la transmisión de imágenes, video e instrucciones a larga distancia.

3.2.5 SEÑALES

Tabla con las señales. Se excluyen las señales de comunicación como la UART, I2C, etc.

Tabla 3: Señales de control.

Nombre	Rango	Tipo	Descripción
V_CRG	3.5 – 24 V	DC	Voltaje de carga.
GND	0 V	DC	Voltaje de tierra.
+12V	12 V \pm 5%	DC	Alimentación para los componentes a 12V (actuadores lineales).
+5V	5 V \pm 5%	DC	Alimentación para los componentes a 5V (servomotores, L293D, etc.)
+3.3V	3.3 V \pm 5%	DC	Alimentación para los componentes a 3.3V (microcontrolador, FC, etc.)
V_SHUNT_E1 V_SHUNT_E2 V_SHUNT_T1 V_SHUNT_T2	0 – 3.3V	Analógica	Señal para indicar al controlador la dirección de movimiento del actuador.
FC_0	0 – 3.3 V	Digital NA	Final de carrera que indica la posición inicial.
PWM_1 PWM_2	0 – 3.3 V	50 Hz	Señal para los servomotores, exteriores o interiores. Duty 0.5 ms (Desactivado). Duty 2.5 ms (Activado).
M_E1_ENABLE M_E2_ENABLE M_T1_ENABLE M_T2_ENABLE	0 – 3,3 V	Digital NA	Señal para activar el controlador del motor seleccionado.
M_E1_DIR+ M_E2_DIR+ M_T1_DIR+ M_T2_DIR+	0 – 3,3 V	Digital NA	Señal para indicar al controlador la dirección de movimiento del actuador.
M_E1_DIR- M_E2_DIR- M_T1_DIR- M_T2_DIR-	0 – 3,3 V	Digital NA	Señal para indicar al controlador la dirección de movimiento del actuador.

3.2.6 PCB

Respecto al diseño de la PCB, este trabajo se limita a representar un modelo conceptual basado en el espacio provisto para la placa y los componentes que se proponen utilizar.

En la Figura 61, se puede ver que los componentes principales colocados en la parte superior son los portabaterías y algunos componentes de electrónica de potencia como los convertidores conmutados (naranja) y los puentes H (Encapsulado THT de 16 patas).

La parte inferior deberá estar provista de conectores rápidos para el RP compute module y el sensor de misión.

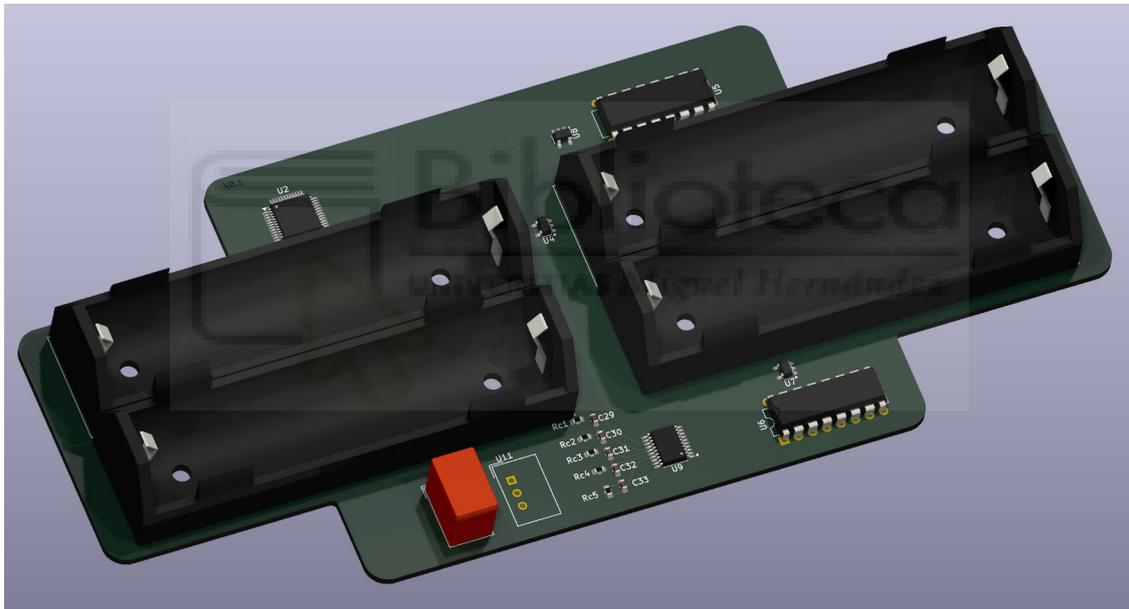


Figura 61: Modelo conceptual de la PCB del Xrobin.

3.3 SOFTWARE

Respecto a la arquitectura de software del robot, esta vendría inspirada por el firmware Klipper, ampliamente empleado en impresoras 3D. A continuación, se explica brevemente la arquitectura de Klipper, que sería análoga a la del robot:

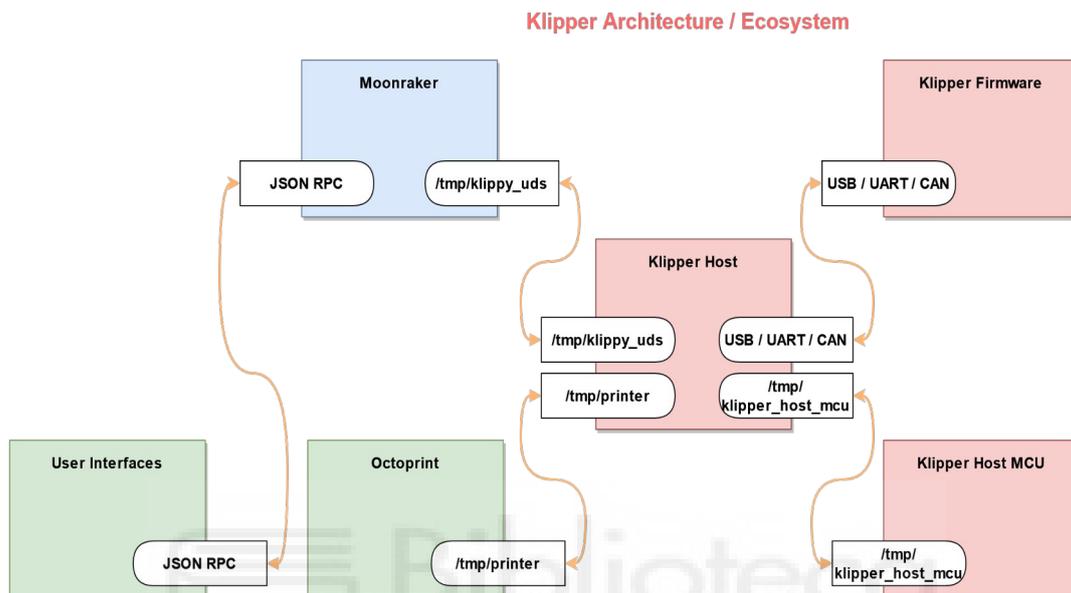


Figura 62: Diagrama de bloques del software de Klipper.

- Las cajas verdes representan diferentes interfaces de usuario, lo que se muestra en un navegador web. Este se aloja en una computadora monoplaca (p. ej. Raspberry Pi).
- La caja azul es el conector entre el firmware de bajo nivel y la interfaz de usuario. Este se aloja en una computadora monoplaca.
- Las cajas rojas representan el *firmware* en sí mismo:
 - El Host, se encarga de gestionar la lógica del robot, tareas complejas, interfaces, etc. Se suele alojar en una computadora monoplaca.
 - El firmware, es la parte del código alojada en el microcontrolador. Esta parte se encarga de gestionar el hardware de la placa. En nuestro caso el módulo de control. Un Host puede controlar varios firmware, en otras palabras, varios dispositivos.
 - El Host MCU, cuando el Host se comporta también como MCU. Por ejemplo para conectar un acelerómetro. En nuestro caso, se conectará directamente al sensor de misión.

Un ejemplo de interfaz de usuario serían MainSail, Fluidt, Octoprint entre otros.

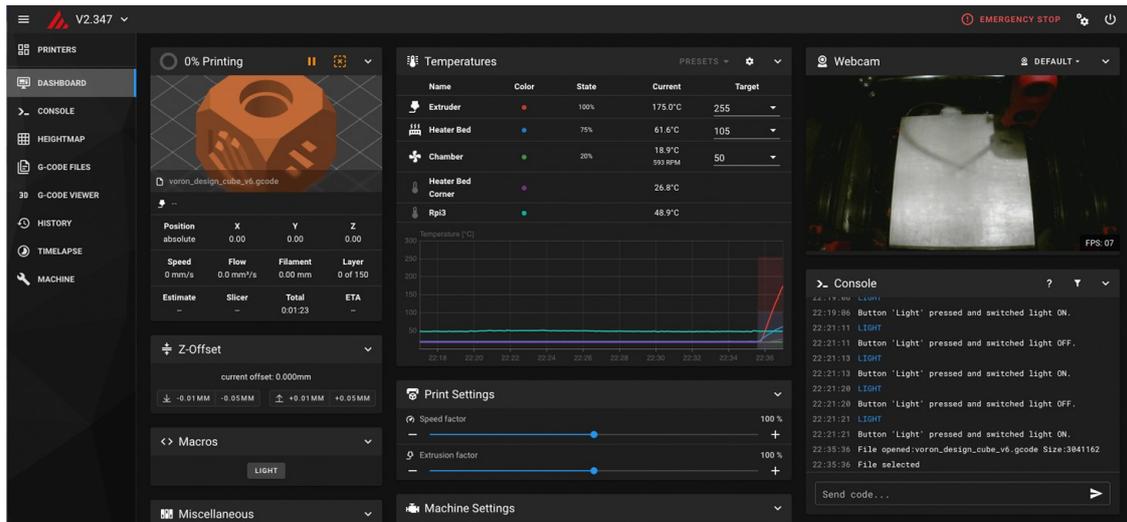


Figura 63: Interfaz de usuario de Mainsail.

Tomando como ejemplo Mainsail, la tecnología base empleada para la interfaz de usuario son:

- Vue.js, un framework progresivo para desarrollar interfaces de usuario en JavaScript.
- Vuetify, una biblioteca de componentes de interfaces de usuario de código abierto basada en Vue.js. Esta tecnología es muy interesante ya que permite un desarrollo rápido y estandarizado, lo que simplifica el trabajo enormemente.

Para más detalles, se recomienda visitar la documentación oficial del proyecto.

- <https://www.klipper3d.org/>
- <https://docs.mainsail.xyz/>
- <https://vuetifyjs.com>
- <https://vuejs.org/>
- <https://moonraker.readthedocs.io>



El software necesario para las comunicaciones e interfaces web escapa de los conocimientos del autor. Sin embargo, se hará una mención a las tecnologías usadas en Klipper que pueden servir como punto de partida para desarrollos futuros.

Respecto al software que controle el hardware del robot, sensores y actuadores, existen diferentes formas de estructurar el software dependiendo de la complejidad del proyecto. La más común es realizar un bucle infinito, en la cual el programa se ejecuta de forma secuencial. Esta suele ser la solución más sencilla para proyectos de baja complejidad. Sin embargo, su implementación esconde varios inconvenientes:

- Capacidad para recordar el contexto.
- Código espagueti (cuando el código se vuelve desorganizado y difícil de entender, normalmente porque se deben realizar muchas correcciones para mantener el contexto adecuadamente).
- Escalabilidad.
- Desaprovechamiento de recursos del microcontrolador.
- Concurrencia.

Una solución para esto sería el uso de máquinas de estado.

Una máquina de estado o SM (State Machine) se definen como un conjunto de estados que sirven de intermediarios en esta relación de entradas y salidas, haciendo que el historial de señales de entrada determine para cada instante, un estado para la máquina de forma tal que la salida depende únicamente del estado y las entradas actuales.

Las máquinas de estado presentan la ventaja fundamental de que son capaces de retener el contexto, permitiendo realizar diferentes acciones para una misma entrada en función del contexto.

Además, se pueden representar de forma visual con diagramas UML (Universal Machine Language). Este diagrama representa las funciones de un teclado representado como máquina de estados.

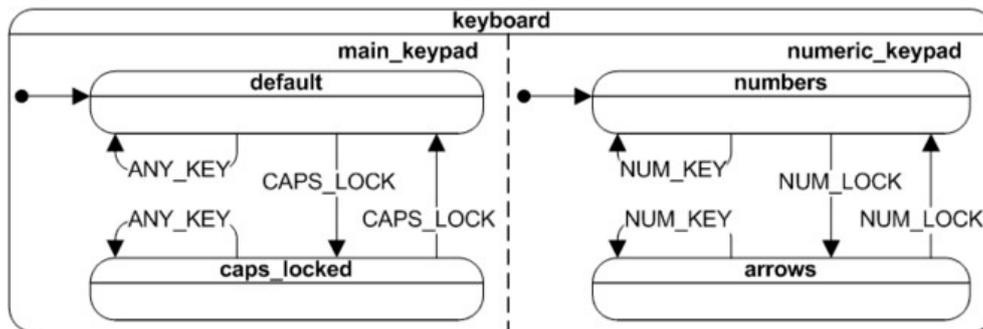


Figura 64: Diagrama de una máquina de estados para un teclado.

Sin embargo, la máquina de estados por sí sola no es capaz de resolver los problemas del desaprovechamiento de recursos del microcontrolador y la concurrencia. Para esto último, un sistema operativo en tiempo real (RTOS, de sus siglas en inglés, Real Time Operative System) sí que podría sacarle el máximo partido al microcontrolador. Un ejemplo de RTOS es FREERTOS, el cual es FOSS y fácil de implementar usando STM32Cube IDE. El inconveniente principal es que poner en marcha estos sistemas requiere de conocimientos más avanzados.

A continuación, se describirán las herramientas que se han probado en este trabajo para la implementación de la máquina de estados.

3.3.1 STATESMITH

StateSmith es una herramienta multiplataforma, libre y de código abierto para generar máquinas de estados en múltiples lenguajes de programación. El código generado es legible, tiene cero dependencias y es adecuado para su uso con pequeños microcontroladores, videojuegos, aplicaciones, web, ordenadores... Evita las asignaciones dinámicas de memoria por motivos de seguridad o rendimiento.

En esta imagen del repositorio de Github, se muestra el proceso a seguir para generar el código [6].

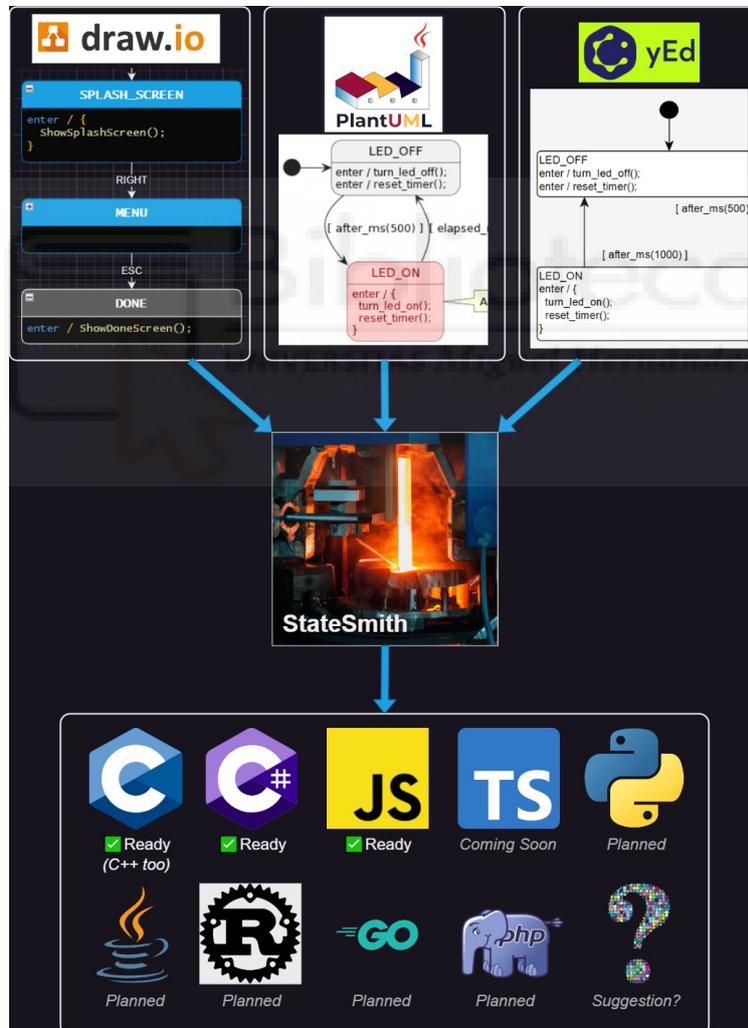


Figura 65: Flujo de trabajo de Statesmith.

Se llegó a realizar un ejemplo con dos botones, uno para cambiar de estado con la señal PB_ACTIVE y otro para para el proceso con la señal PB_KILL. Los estados hacen que un LED se apague y se encienda con diferente frecuencia.

Se pueden ver en el interior de los estados las transiciones a partir de las señales y las funciones típicas de la librería HAL de STM32.

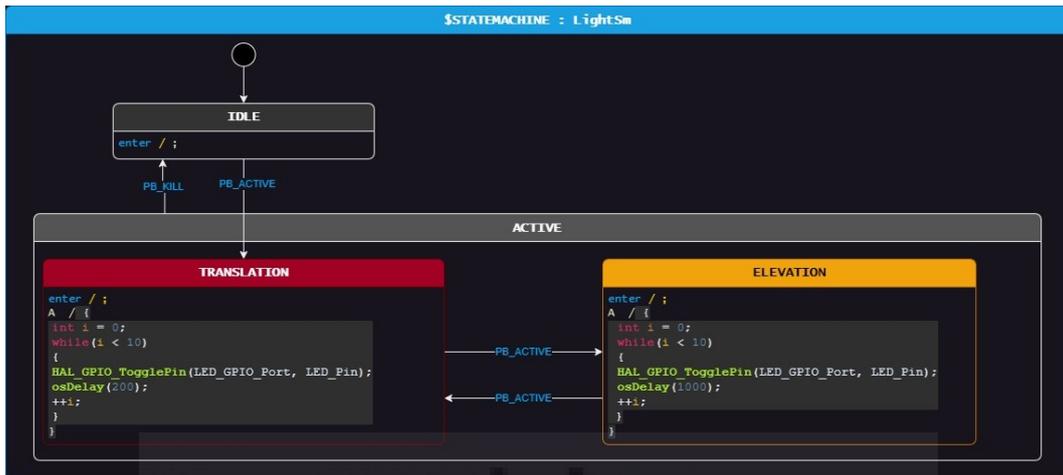


Figura 66: Máquina de estados en draw.io como entrada para Statesmith.

El objetivo de este test era comprobar cómo de sencillo era usar esta herramienta. En la opinión del autor de este trabajo, una vez superada la puesta en marcha y entender como generar el código, fue sencillo de utilizar. En manos de una persona con conocimientos de software embebido esta herramienta puede ser muy eficaz para el prototipado.

3.3.2 QP (QUANTUM PLATFORM)

Como dice el autor, Miro Samek, en su web: “QP™ (Quantum Platform) es una familia de frameworks embebidos en tiempo real (RTEFs) de código abierto y entornos de ejecución basados en Objetos Activos y Máquinas de Estado Jerárquicas. La familia QP se compone de los frameworks QP™/C y QP™/C++, cuya calidad está estrictamente controlada, exhaustivamente documentada y disponible bajo el flexible modelo de licencia dual.”

En esta imagen se muestra a alto nivel la función de QP.

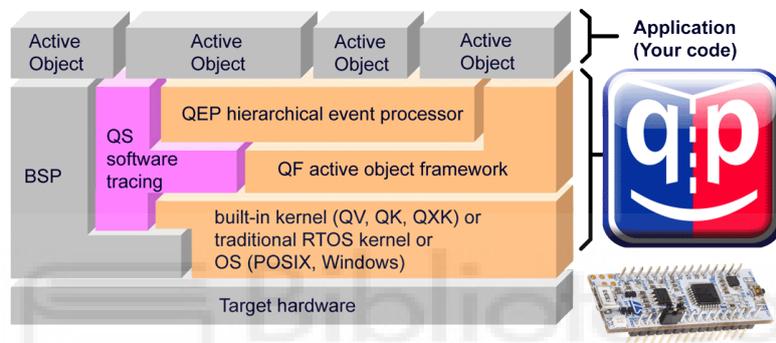


Figura 67: Diagrama de bloques de QP.

Lo más interesante de este ecosistema es la herramienta llamada QM, una interfaz gráfica para el dibujo y la generación de las máquinas de estados, junto al resto de elementos necesarios.

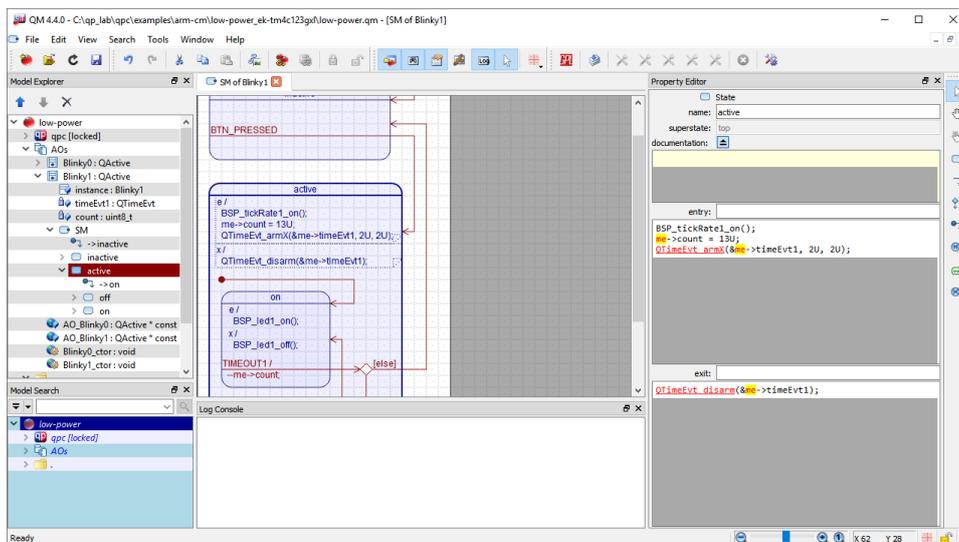


Figura 68: Interfaz de usuario de QM.

Los RTEF de QP™ también pueden funcionar con muchos RTOS tradicionales de terceros. Se proporcionan puertos QP y ejemplos listos para usar para varios RTOS: embOS, FreeRTOS, ThreadX, uC/OS-II, Zephyr.

Además, recientemente se ha integrado en el STM32CubeIDE [7].

En resumen, esta es una herramienta muy potente y de uso profesional. De hecho, se ha usado como máquina de estados jerárquica (HSM, Hierarchical State Machine) en varios proyectos de la NASA. Steve Scandore, líder técnico del software de vuelo en el proyecto Perseverance, reconoció el uso de esta herramienta en la “Embedded Online Conference 2021”. Este hace hincapié en el apartado de generación automática del código a partir de un archivo de texto, lo cual ahorra tiempo y errores a los desarrolladores de software [8].

A continuación, se enumeran algunos recursos útiles sobre esta herramienta y donde encontrarlos:

- Sitio web: <https://www.state-machine.com/>
- Github: <https://github.com/QuantumLeaps/>
- Manual online de QM: <https://www.state-machine.com/qm/>
- [Book: Practical UML Statecharts in C/C++, 2nd Ed](#)
- [Book: Practical Statecharts in C/C++](#)
- Uso de licencias <https://www.state-machine.com/licensing>

Respecto al apartado del software y electrónica, tratar de hacer funcionar este framework ha sido la parte que más tiempo ha llevado aunque finalmente solo fue posible compilar el código y ejecutarlo en un microcontrolador STM32F103C8T6. Sin embargo, la máquina de estados nunca llegó a funcionar correctamente ya que al entrar en ella, nunca se terminaba de ejecutar la primera acción.

A pesar de lo mencionado anteriormente, ésta destaca como la herramienta más profesional, completa, mejor documentada y con más uso profesional de todas las que se han probado en este trabajo.

Por desgracia, esta implementación queda actualmente fuera del alcance de este TFM, dejándola como trabajo futuro para otro estudiante que disponga de suficientes conocimientos y experiencia en software embebido para ser capaz de hacer funcionar esta herramienta con facilidad.

3.3.3 BOOST-EXT/SML

Esta librería permite la escritura de máquinas de estados de forma declarativa, es decir, se escribe el comportamiento en el código y la librería se encarga de generar el código subyacente.

La librería forma parte del conjunto de librerías Boost::ext para C++. La librería no tiene dependencias (ni si quiera de STL o Boost) y es ligera.

Esta herramienta no ha sido probada en este TFM pero parece una opción prometedora. Para conocer más detalles, se recomienda ver los enlaces del proyecto y la charla del autor de la librería, Kris Jusiak, en “C++Now 2019”:

- <https://boost-ext.github.io/sml/index.html>
- <https://github.com/boost-ext/sml>
- <https://www.youtube.com/watch?v=Zb6xcd2as6o>

3.3.4 MÁQUINA DE ESTADOS PARA EL XROBIN

Los estados se representan con una X , de Xrobin, acompañada de un código:

$$X_{rst}^{ES} \rightarrow X_{+00}^{HE}$$

Tabla 4: Descripción de los símbolos usados en los estados.

Ítem	Descripción	Estados	Significado
E	Estado de los actuadores de elevación.	H	Pie exterior por debajo del interior.
		M	Pie exterior e interior al mismo nivel.
		L	Pie exterior por encima del interior.
S	Estados de los servomotores exterior e interior, en ese orden.	O	Ambos desactivados.
		E	Exterior activado.
		I	Ambos activados.
		X	Interior activado.
r	Estado de traslación	+	h, a, b, c (Figura 6)
		-	d, e, f, g (Figura 6)
s	Estado del actuador de traslación 1.	0	Extendido.
		1	Contraído.
t	Estado del actuador de traslación 2.	0	Extendido.
		1	Contraído.

Estas son las acciones permitidas:

Tabla 5: Acciones permitidas en cada estado.

Transición	Mensaje	Descripción
Elevación	+E	Elevar los pies exteriores
	-E	Contraer los pies exteriores
Traslación	+T1	Extender el actuador de traslación 1
	-T1	Contraer el actuador de traslación 1
	+T2	Extender el actuador de traslación 2
	-T2	Contraer el actuador de traslación 2

Servomotor	+Se	Activar los servomotores del pie exterior
	-Se	Desactivar los servomotores del pie exterior
	+Si	Activar los servomotores del pie interior
	-Si	Desactivar los servomotores del pie interior

Reglas para las transiciones:

- Las transiciones deben realizarse secuencialmente.
- El robot solo puede trasladarse si el estado de elevación es completamente extendido o contraído.
- Solo se puede cambiar el estado de los servomotores cuando el robot se encuentre en el punto medio de elevación.
- Al menos un juego de servomotores debe estar activado en todo momento salvo que se le ordene al robot con un comando especial.

Con estas simples reglas, se pueden determinar las acciones permitidas en cada estado.

Tabla 6: Transiciones permitidas en cada estado.

Estado		Transiciones		
		Elevación	Traslación	Servomotor
Hi	X_{rst}^{HE}	-E	$\pm T1$ $\pm T2$	No permitido
Mid	X_{rst}^{ME}	+E	No permitido	+Si
	X_{rst}^{MX}	No permitido	No permitido	-Se / -Si
	X_{rst}^{MI}	-E	No permitido	+Se
Low	X_{rst}^{LI}	+E	$\pm T1$ $\pm T2$	No permitido

Ahora podemos desarrollar una tabla de transiciones más detallada en la que se pueden ver las acciones necesarias para pasar de un estado a otro. Se representan solo 4 posibles estados de traslación de los 8 posibles para facilitar la visualización.

Tabla 7: Tabla de transiciones.

X_{+00}^{HE}	\rightarrow +T2 \leftarrow -T2	X_{+01}^{HE}	\rightarrow +T1 \leftarrow -T1	X_{+11}^{HE}	\rightarrow -T2 \leftarrow +T2	X_{+10}^{HE}	\rightarrow -T1 \leftarrow +T1
\uparrow +E \downarrow -E	NP						
X_{+00}^{ME}		X_{+01}^{ME}		X_{+11}^{ME}		X_{+10}^{ME}	
\uparrow -Si \downarrow +Si		\uparrow -Si \downarrow +Si		\uparrow -Si \downarrow +Si		\uparrow -Si \downarrow +Si	
X_{+00}^{MX}		X_{+01}^{MX}		X_{+11}^{MX}		X_{+10}^{MX}	
\uparrow +Se \downarrow -Se		\uparrow +Se \downarrow -Se		\uparrow +Se \downarrow -Se		\uparrow +Se \downarrow -Se	
X_{+00}^{MI}		X_{+01}^{MI}		X_{+11}^{MI}		X_{+10}^{MI}	
\uparrow +E \downarrow -E		\uparrow +E \downarrow -E		\uparrow +E \downarrow -E		\uparrow +E \downarrow -E	
X_{+00}^{LI}	\rightarrow +T2 \leftarrow -T2	X_{+01}^{LI}	\rightarrow +T1 \leftarrow -T1	X_{+11}^{LI}	\rightarrow -T2 \leftarrow +T2	X_{+10}^{LI}	\rightarrow -T1 \leftarrow +T1

Con esta tabla se pueden determinar los pasos necesarios para pasar de un estado a otro. Pero se debe tener en cuenta que la posición final del actuador depende del camino tomado. El resultado de las acciones $X_{+01}^{HE} \rightarrow X_{+11}^{HE} \rightarrow X_{+11}^{LI}$ es diferente a $X_{+01}^{HE} \rightarrow X_{+01}^{LI} \rightarrow X_{+11}^{LI}$, ya que, en el primer caso se mueve la plataforma, mientras que en el segundo caso, se mueve el bastidor.

Las secuencias deben ser calculadas inicialmente o en tiempo real para ir de las coordenadas iniciales a las finales. El algoritmo para realizar este cálculo, no entra en el alcance de este TFM.

En el caso de haber sido implementado como máquina de estados, este sería una pseudo-representación del comportamiento deseado. Nótese que el uso de las submáquinas de estado y las transiciones no sigue el estándar de QM; solo se trata de representar de forma intuitiva el comportamiento a seguir.

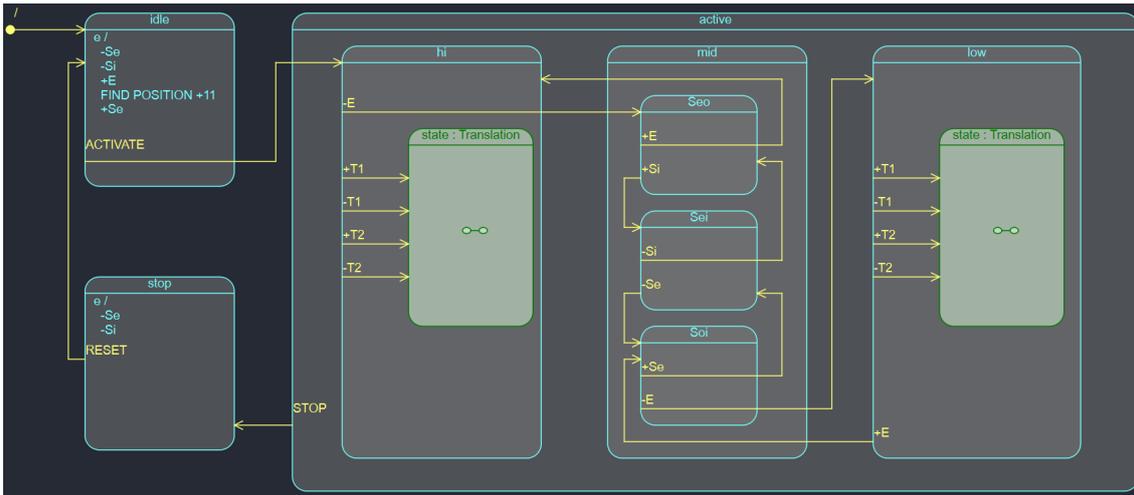


Figura 69: Propuesta de la máquina de estados principal para el Xrobin.

La máquina siempre empieza en la transición inicial, que apunta al estado “idle”. Al entrar, se ejecutan las acciones de entrada (e/), en este caso, desactivar los servomotores, extender el pie exterior, encontrar la posición inicial y activar el servomotor exterior.

Para pasar al estado “active” se debe recibir el mensaje “ACTIVE”, en este caso se entra directamente al estado “active.hi”. A partir de aquí, los diferentes mensajes guiarán el comportamiento del robot a través del diagrama.

Es posible realizar acciones comunes a varios sub-estados. Por ejemplo, el mensaje “STOP” permite pasar del estado “active” a “stop”, justo al entrar se desactivan los servomotores. Si envía el mensaje “RESET”, se vuelve al estado “idle” y se repiten las acciones de entrada al estado.

Las acciones de traslación se llevarán a cabo dentro de una sub-máquina de estados encargada de realizar las transiciones y retener el contexto.

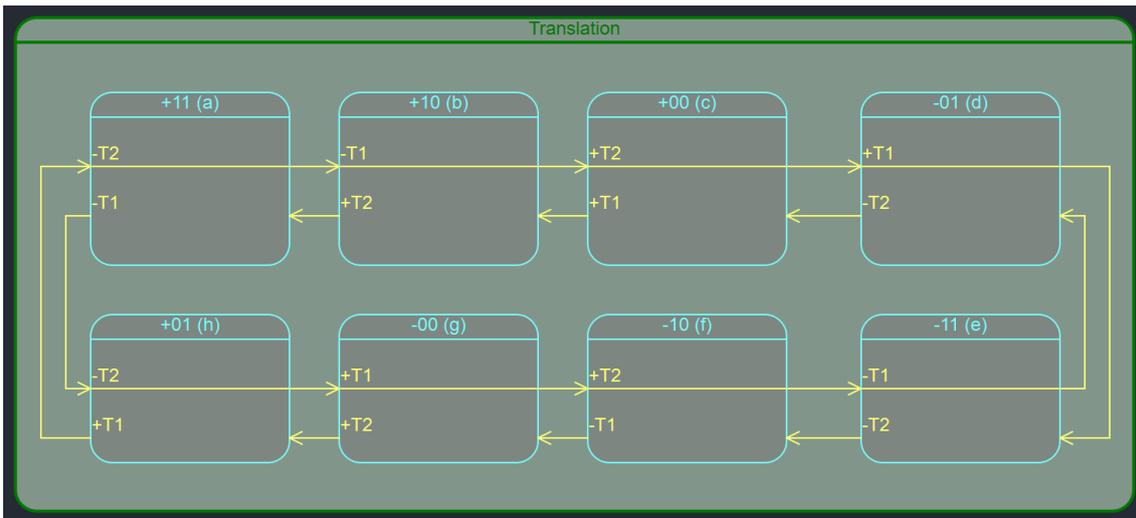


Figura 70: Propuesta de una sub-máquina para el Xrobin donde se controlan las traslaciones.

Para entender y sacar el máximo rendimiento a QM, se debe consultar la documentación oficial, mencionada en el apartado de QM.



4 RESULTADOS

En esta sección se ha hecho una recopilación de los resultados de aplicar los puntos mencionados a lo largo del trabajo a la construcción de este segundo prototipo del Xrobin.

4.1 CONSTRUCCIÓN DE LOS SERVOMÓDULOS

4.1.1 MECANIZADO DE LOS COLECTORES MAGNÉTICOS

Los colectores magnéticos fueron mecanizados a partir de unas barras de acero de sección rectangular de 8x20x25 mm. En un primer amarre, se mecanizó el contorno y el rebaje de la parte inferior. Posteriormente, se añadieron chaflanes y se cortó la pieza usando un disco circular.

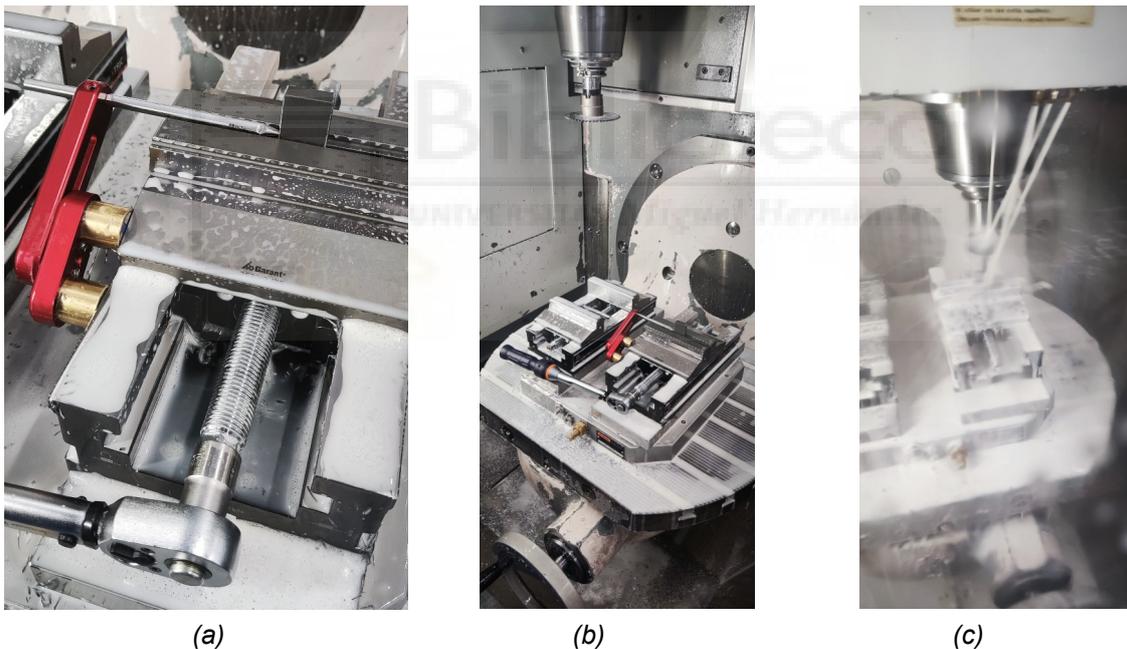
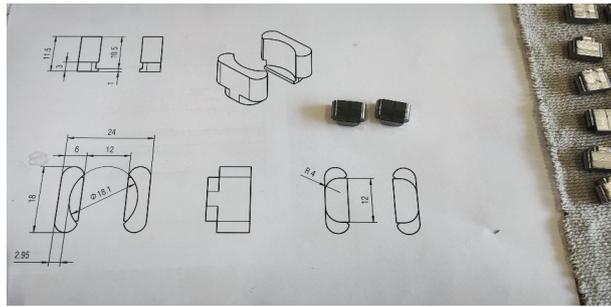


Figura 71: Amarre del bruto con un localizador magnético, mordaza (a) y máquina (b), e imagen durante el proceso de mecanizado (c).

El siguiente amarre consiste en una mordaza de aluminio con la forma de las ranuras donde irán finalmente alojadas para poder mecanizar el alojamiento del imán en dos piezas a la vez.



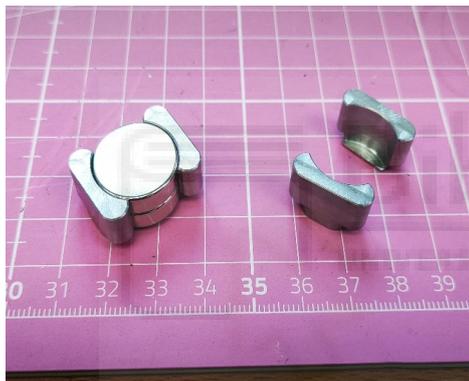
(a)



(b)

Figura 72: Piezas mecanizadas del primer amarre sobre el plano de referencia para el mecanizado (a) y amarre de los colectores magnéticos en una mordaza de aluminio para la segunda operación de mecanizado (b).

Este es el resultado obtenido. Un par de colectores magnéticos pequeños, ligeros y sencillos de fabricar. Además, encajan a la perfección con los imanes seleccionados.



(a)



(b)

Figura 73: Vista superior (a) e inferior (b) de los colectores magnéticos con imanes.

4.1.2 MONTAJE DE LOS COLECTORES E IMANES

El siguiente paso consiste en montar los colectores magnéticos en sus alojamientos. Estos encajan con un ajuste deslizante y son fijados usando unas gotas de súper-pegamento. Son de fácil instalación y se pueden retirar con un pequeño golpe de martillo. Para limpiarlos se puede usar acetona.



Figura 74: Proceso de ensamblaje de los colectores magnéticos (a) y ensamblaje completado (b).

El siguiente paso es marcar los polos del imán de acurdo con la Figura 16. Para ello es necesario usar un buscador de polos magnéticos o un imán del cual se sepa su polaridad. Con ayuda del visor de campo magnético es más fácil trazar la línea media. Esto es porque un error en el ángulo de montaje tiene implicaciones en la forma final del campo magnético.

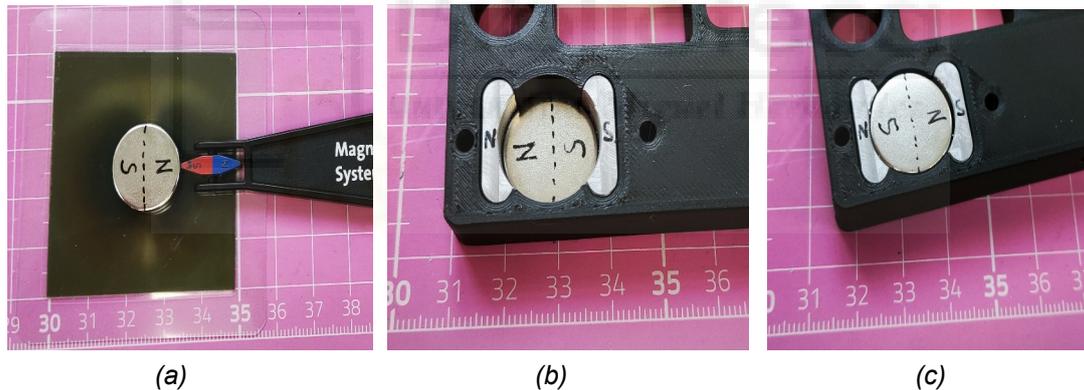


Ilustración 4.1: Identificación del norte magnético en un imán (a), montaje de los imanes fijo (b) y móvil (c) del servomódulo(c)

A continuación, se puede proceder a pegar el acoplador al imán superior de acuerdo con las marcas de norte y sur.

4.1.3 ENSAMBLAJE DEL SERVOMOTOR

Después, se embute lentamente el acoplador del servomotor en el eje usando una prensa. El siguiente paso es, preparar el soporte con pin liso de 3mm y un

inserto de latón de M3. Finalmente, solo queda ensamblar el servomotor en su soporte con dos tornillos y asegurar el conjunto con un único tornillo de M3.

Cabe recordar que el servomotor debe encontrarse en 0° y el acoplador debe montarse según se ve en la Figura 13.

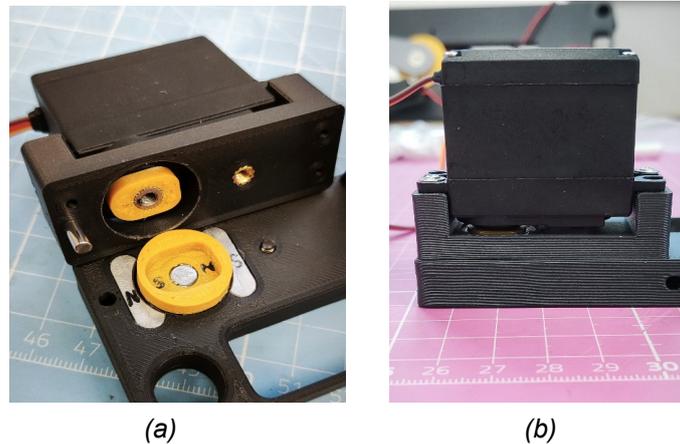


Figura 75: Servomódulo antes (a) y después (b) de ser ensamblado.

4.1.4 CAMPO MAGNÉTICO DE LOS IMANES CONMUTABLES

Se ha empleado un visor de campo magnético en forma de lámina fina multicolor, el cual indica mayor intensidad de flujo con tonos azules y menor con tonos amarillos. Nótese que solo se puede detectar flujo perpendicular a la lámina.

En primer lugar, se muestran el campo magnético generado por el imán usado en el prototipo inicial, el diseño de esta se muestra en la Figura 1. A la izquierda, el imán desactivado, mientras que a la derecha, el imán activado. Las observaciones concuerdan con la figura mencionada anteriormente.

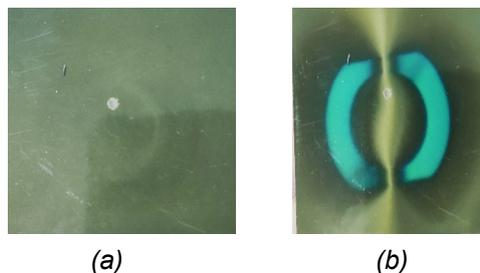


Figura 76: Observación del campo magnético en los estados desactivado $[0^\circ]$ (a) y activado $[180^\circ]$ (b) del imán conmutable original.

En segundo lugar, se muestran los nuevos imanes conmutables en diferentes posiciones. Desde estar desactivado a estar completamente activado. En la posición desactivada, se puede ver que hay parte del flujo magnético que escapa al exterior. Esto puede deberse a varias causas:

- Que no haya suficiente metal como para capturar completamente el campo magnético, debido a que el hueco de aire sea demasiado grande.
- Que el metal existente se haya saturado.
- Que la polarización del imán N35 sea más débil y no pueda mantener la dirección diametral de la misma forma que el N48.

En cualquier modo, al activarse, el imán presenta el comportamiento deseado.

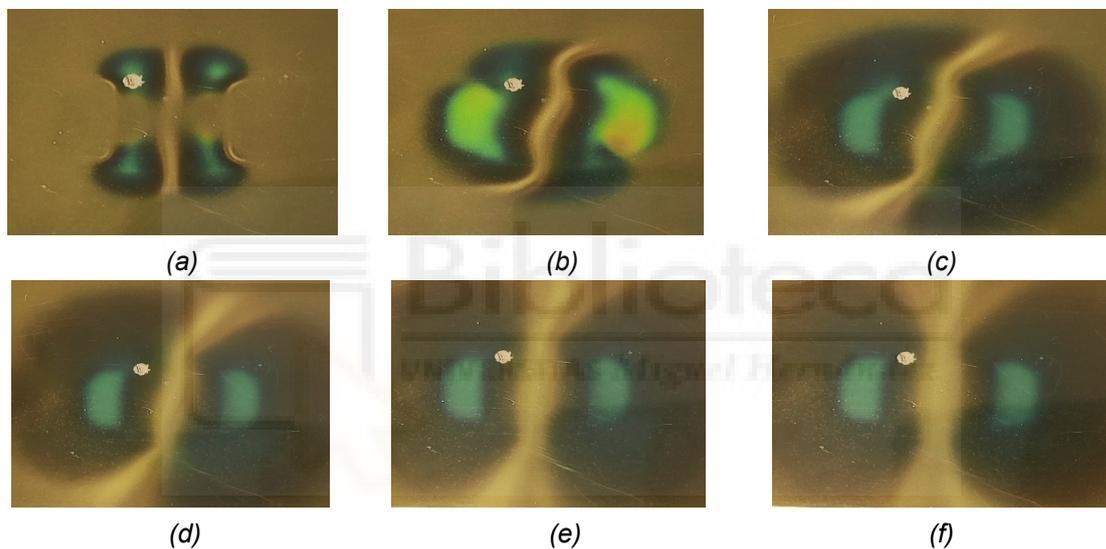


Figura 77: Observación del campo magnético en los estados desactivado $[0^\circ]$ (a), $[36^\circ]$ (b), $[72^\circ]$ (c), $[108^\circ]$ (d), $[144^\circ]$ (e) y desactivado $[180^\circ]$ (f) del imán conmutable nuevo.

4.2 MONTAJE DEL PIE EXTERIOR

El ensamblaje es muy sencillo, solo hace falta seguir estos pasos:

- Montar los servomódulos como se indica en el apartado 4.1.
- Introducir las guías en la base de pie exterior y asegurarlas con el tornillo sin cabeza de M4.
- Colocar los pasadores del actuador lineal.
- Acoplar el pie exterior al bastidor con el tornillo de M4.

Estas imágenes muestran el resultado final tras haber ensamblado todos los componentes del servomódulo.

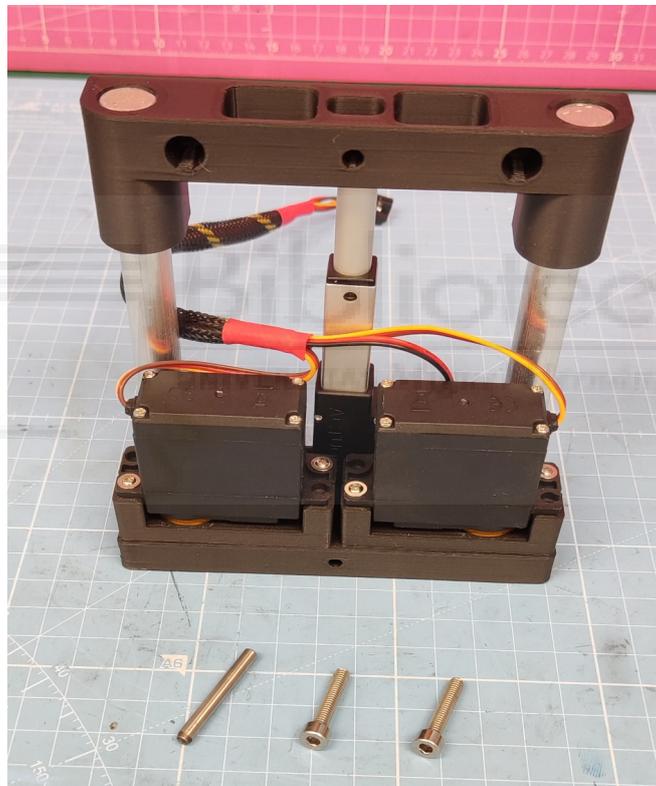


Figura 78: Pie exterior ensamblado y elementos de fijación.

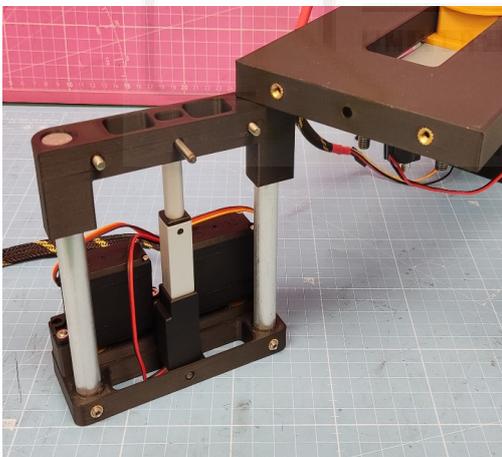
En la vista inferior, se pueden ver los colectores, las guías y el rebaje en la pieza impresa en 3D para evitar la colisión con el cable.



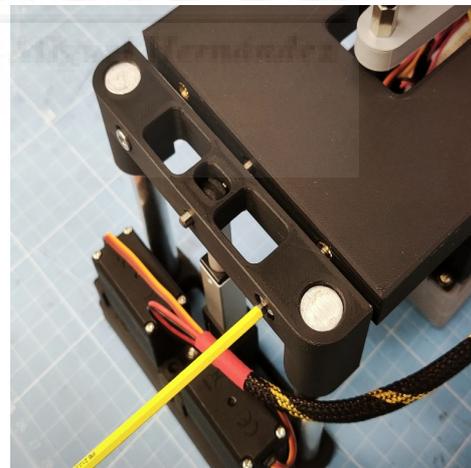
Figura 79: Vista inferior del pie exterior ensamblado.

En estas imágenes, se puede ver la facilidad de ensamblado de los pies exteriores gracias al concepto modular.

1. Se introduce el pasador en el pie exterior.
2. Se une con el bastidor.
3. Se unen las dos piezas con tornillos de M4.



(a)



(b)

Figura 80: Interfaces del pie exterior y el bastidor (a) y ensamblaje de ambas partes (b).

4.3 MONTAJE DEL PIE INTERIOR

El ensamblaje es sencillo, solo hace falta seguir estos pasos:

- Montar los servomódulos como se indica en el apartado 4.1, pero no ensamblar los servos hasta haber colocado la PCB (pieza amarilla).
- Atornillar los pilares usando los tornillos avellanados de M6.
- En el caso del prototipo, ensamblar la electrónica en una placa prototipo y placas auxiliares.

Estas imágenes muestran el resultado final.

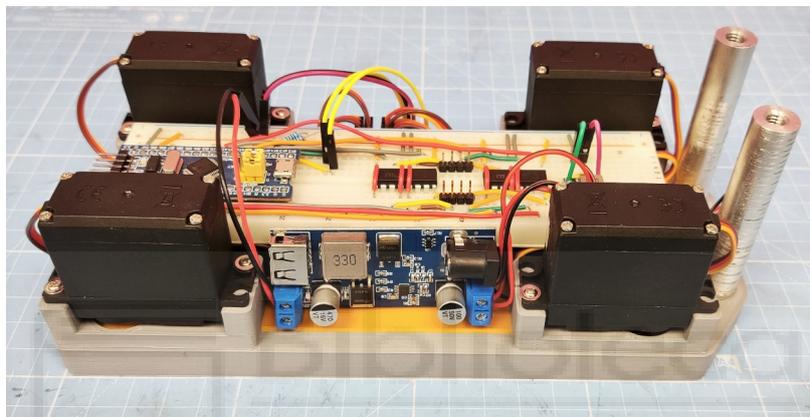


Figura 81: Pie interior completamente ensamblado incluyendo componentes electrónicos.

En la vista inferior, se puede ver el espacio reservado para alojar más componentes electrónicos.



Figura 82: Vista inferior del pie interior ensamblado.

4.4 MONTAJE DEL BALANCÍN Y LA DESLIZADERA

Este apartado pretende mostrar como quedaría ensamblado el balancín sobre la deslizadera. Nótese que estas piezas deben ser montadas sobre el bastidor y no por separado.

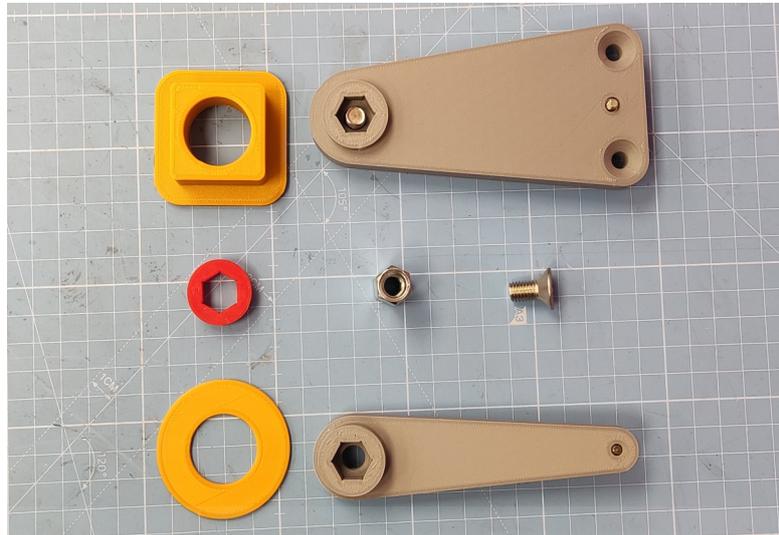
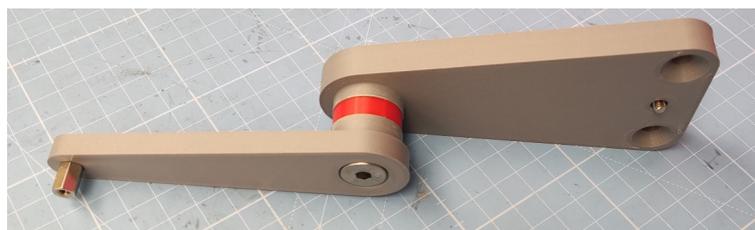


Figura 83: Subcomponentes del balancín: bieletas, deslizadera, arandela, separador central, tuerca larga M8 y tornillos M8 avellanados.

Tras el montaje se pudo comprobar que el movimiento de rotación apenas tiene juego y que la interfaz hexagonal evita perfectamente el movimiento relativo entre las dos bieletas.



(a)



(b)

Figura 84: Balancín ensamblado (a) y ensamblado con la deslizadera (b).

4.5 MONTAJE DEL MECANISMO TIJERA

Una vez ensamblados los pies exteriores, el balancín y la deslizadera se puede proceder a montar los actuadores lineales. Es recomendable montarlos completamente extendidos porque así se facilita el atornillado del pie interior.

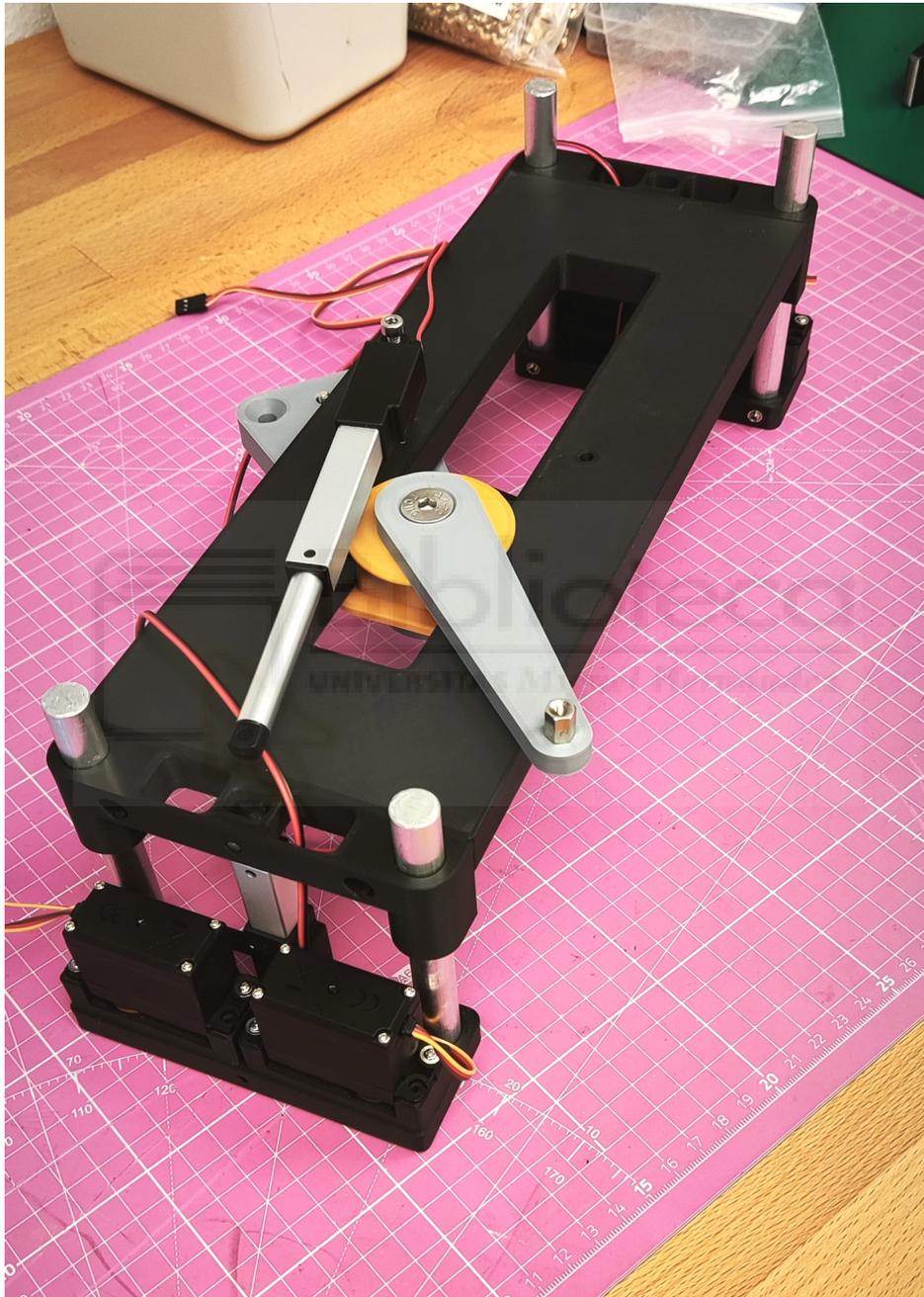


Figura 85: Mecanismo tijera antes de atornillar el actuador de traslación superior (extendido) a la bieleta.

En esta imagen se pueden ver los apoyos de los actuadores, contruidos a partir de separadores hexagonales y tornillos de M4. Se debe mencionar que el tornillo de M4 tiene bastante juego y eso perjudica al comportamiento del robot. Se propone usar algún tipo de casquillo fino para eliminar el juego y actuar como cojinete entre el plástico y el metal.

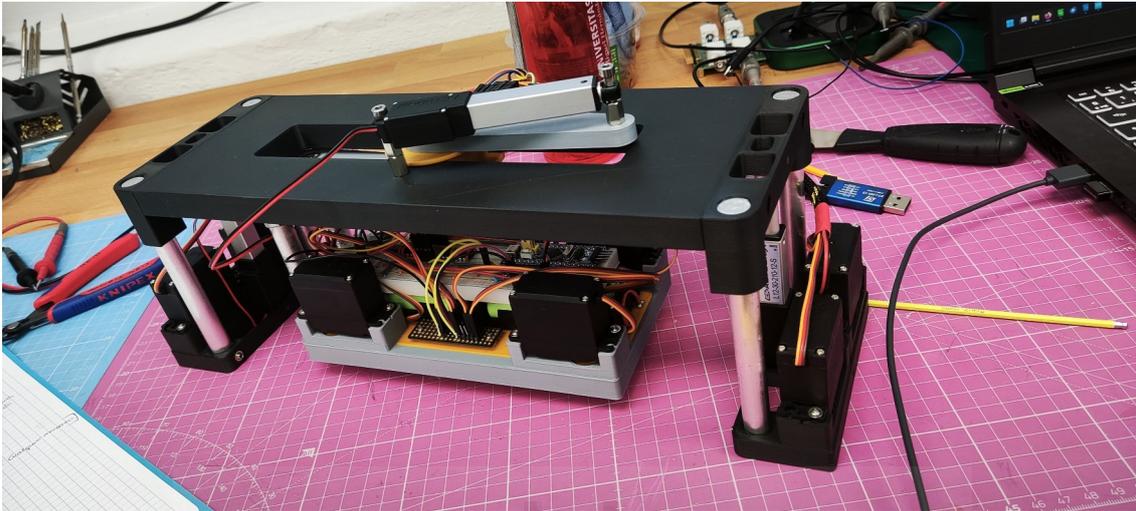


Figura 86: Mecanismo tijera después de atornillar ambos actuadores lineales (contraídos).

Esta imagen muestra un ejemplo de lo que ocurre cuando los actuadores no están extendidos completamente. Uno de los tornillos del pie interior no es accesible.

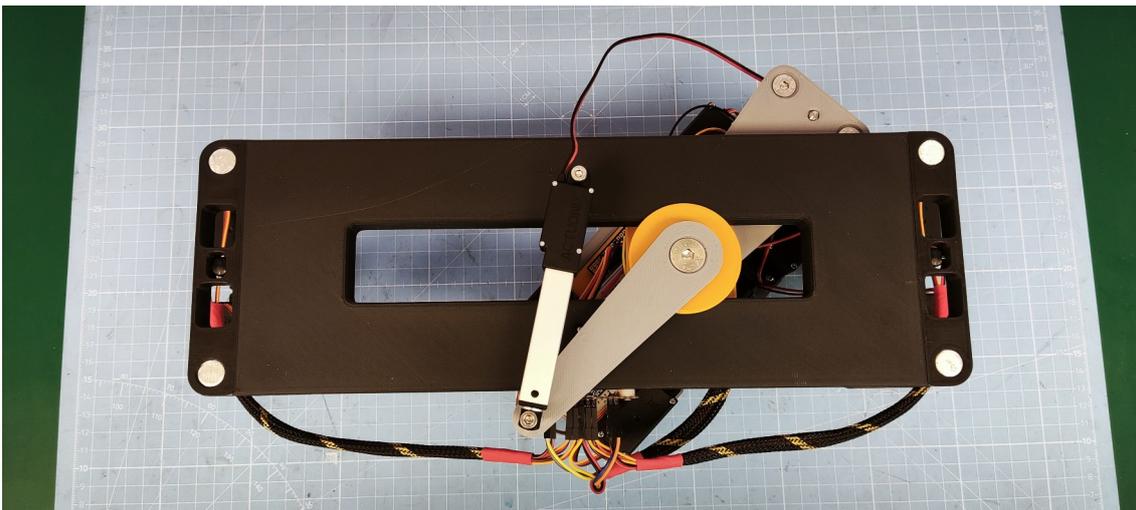


Figura 87: Mecanismo tijera después de atornillar ambos actuadores lineales, superior contraído e inferior extendido.

4.6 ELECTRÓNICA PARA EL PROTOTIPO

La electrónica del robot se ha simplificado respecto a la propuesta del TFM, ya que, la fabricación de la PCB escapa del alcance de este proyecto. Finalmente, los componentes empleados han sido:

- 1 placa de pruebas.
- 2 subplacas para realizar conexiones auxiliares.
- 2 soportes dobles de baterías 18650.
- 4 baterías de litio 18650 conectadas en serie.
- 1 convertor DC a 12V, para los actuadores lineales.
- 1 convertor DC a 5V, para los servos y electrónica de control.
- 1 placa de desarrollo bluepill con un STM32F103C8T6.
- 2 puentes H dobles, L293D para controlar los actuadores lineales.
- 2 resistencias de 1 k Ω para poner a tierra los pines del microcontrolador que generan el PWM que controla los servomotores.
- Cables, conectores y pines para hacer las conexiones.
- Una cubierta tranzada y termorretractil para hacer arneses.

En esta imagen se ven todos los módulos conectados entre si.

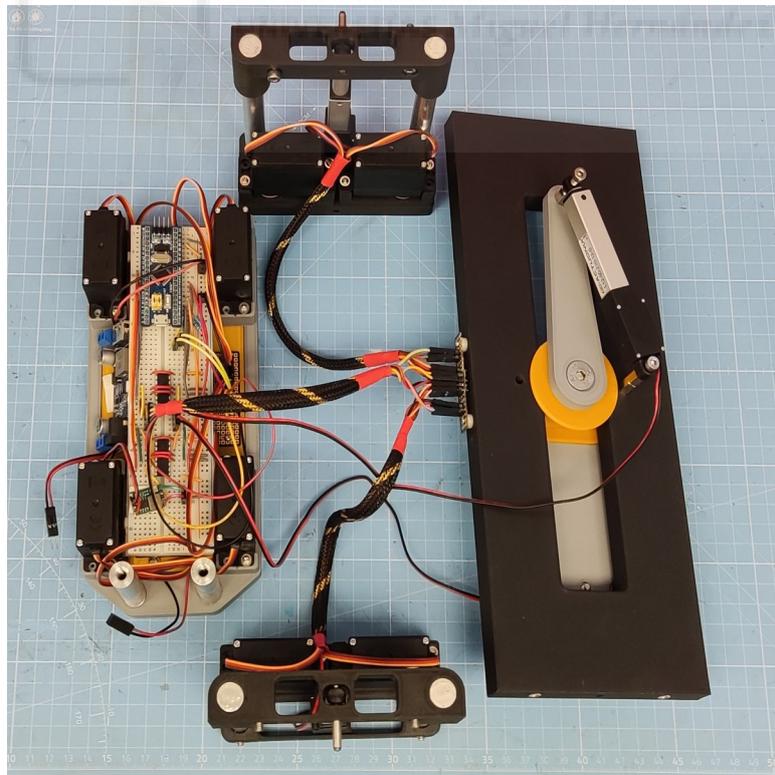


Figura 88: Módulos del robot separados mecánicamente pero conectados eléctricamente.

Aquí se pueden ver todos los componentes alojados en el pie interior, desmontados y montados.

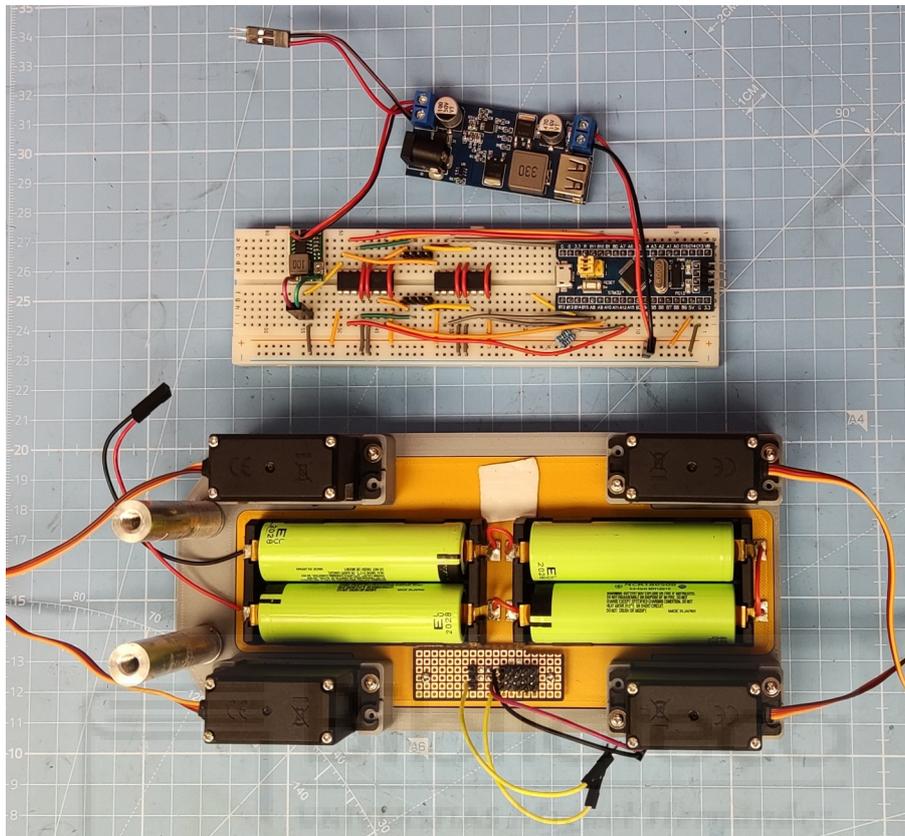


Figura 89: Componentes electrónicos alojados en el pie interior, no ensamblados.

Como se puede ver, el resultado es bastante limpio a pesar de la cantidad de cables que son necesarios para realizar las conexiones.

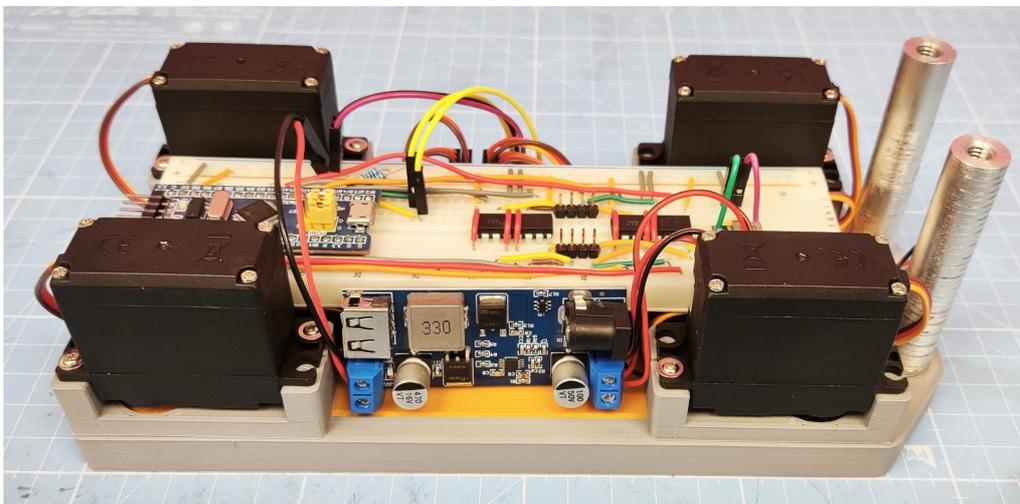
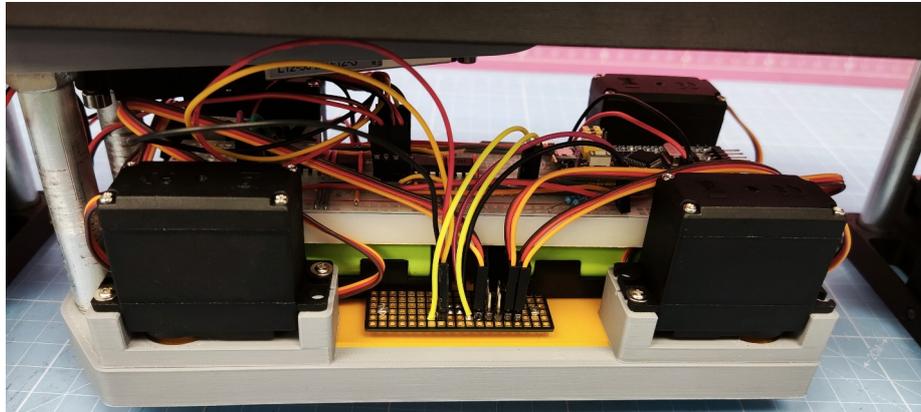
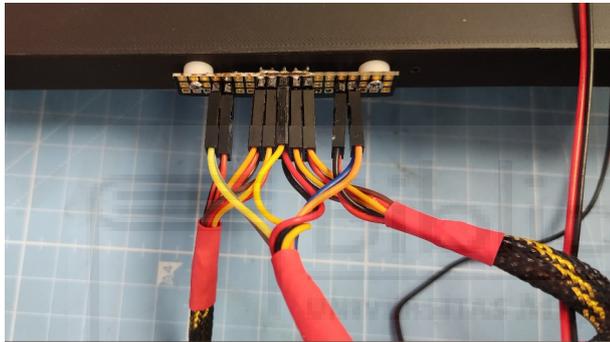


Figura 90: Componentes electrónicos alojados en el pie interior, no ensamblados.

Estas son las placas auxiliares para realizar la conexiones de los servomotores y los actuadores lineales. También se puede ver la terminación de un arnés que acaba en un pie exterior.



(a)



(b)



(c)

Figura 91: Placas de conexiones auxiliar del pie interior (a) y bastidor (b), y un extremo de un arnés de cables.

4.7 SOFTWARE PARA EL PROTOTIPO

El programa no será la versión definitiva con todas las funcionalidades sino una versión de demostración donde se probarán varios movimientos del robot de forma aislada, centrándose únicamente en el software embebido del microcontrolador.

Las secuencias consistirán en funciones que incluirán instrucciones necesarias para pasar de una configuración del robot a otra.

Esta es la configuración del microcontrolador en el STM32CubeIDE:

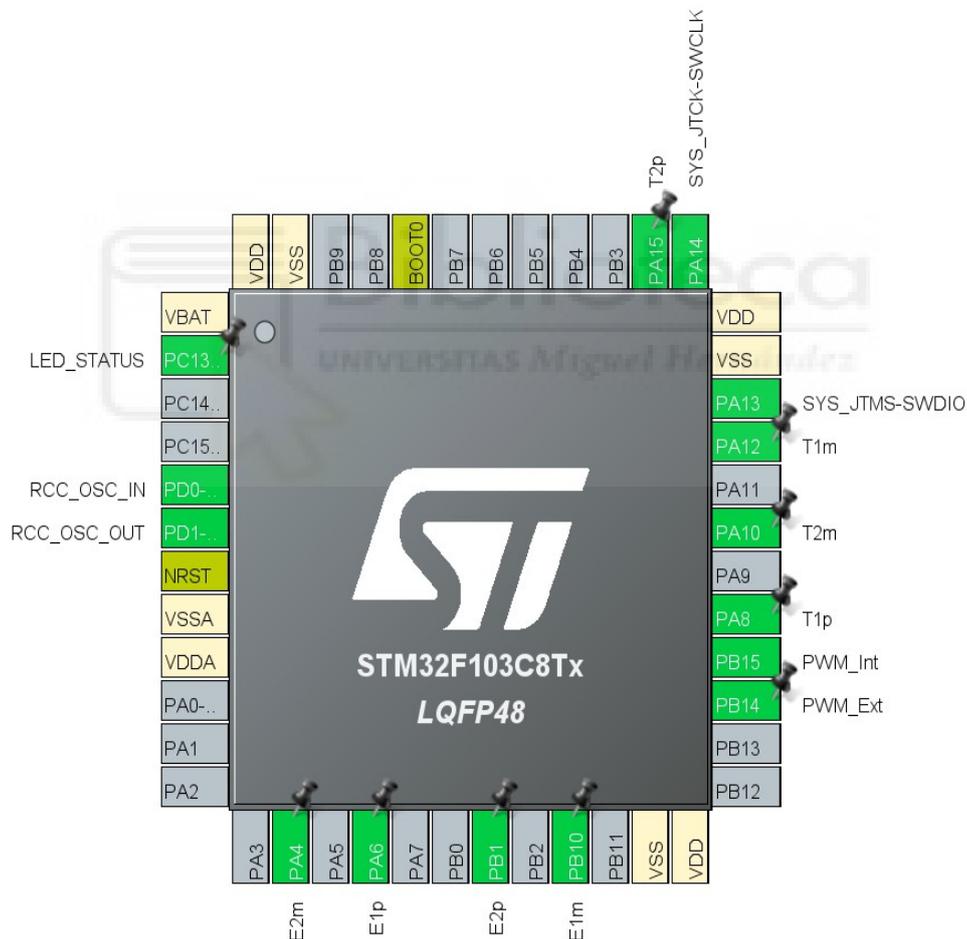


Figura 92: STM32CubeIDE, representación visual de la configuración de pines del microcontrolador.

En esta imagen, la configuración de los GPIO.

Pin Name	Signal on Pin	GPIO output level	GPIO mode	GPIO Pull-up/Pull-down	Maximum ...	User Label	Modified
PA4	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	E2m	✓
PA6	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	E1p	✓
PA8	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	T1p	✓
PA10	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	T2m	✓
PA12	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	T1m	✓
PA15	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	T2p	✓
PB1	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	E2p	✓
PB10	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	E1m	✓
PC13-TAMPER-RTC	n/a	Low	Output Push Pull	No pull-up and no pull-down	Low	LED_ST...	✓

Figura 93: STM32CubeIDE, tabla de configuración de pines del microcontrolador.

En esta otra imagen, la configuración de los temporizadores que generan el PWM.

Mode

- Channel1: Disable
- Channel2: PWM Generation CH2N
- Channel3: PWM Generation CH3N
- Channel4: Disable

Configuration

Reset Configuration

Parameter Settings | User Constants | NVIC Settings | DMA Settings | GPIO Settings

Configure the below parameters:

- Counter Settings
 - Prescaler (PSC - 16 bits value): 8-1
 - Counter Mode: Up
 - Counter Period (AutoReload Register - 16 bits value): 20000-1
 - Internal Clock Division (CKD): No Division
 - Repetition Counter (RCR - 8 bits value): 0
 - auto-reload preload: Enable
- Trigger Output (TRGO) Parameters
 - Master/Slave Mode (MSM bit): Disable (Trigger input effect not delayed)
 - Trigger Event Selection: Reset (UG bit from TIMx_EGR)
- Break And Dead Time management - BRK Configuration
 - BRK State: Disable
 - BRK Polarity: High
- Break And Dead Time management - Output Configuration
 - Automatic Output State: Disable
 - Off State Selection for Run Mode (OSSR): Disable
 - Off State Selection for Idle Mode (OSSI): Disable
 - Lock Configuration: Off
- PWM Generation Channel 2N
 - Mode: PWM mode 1
 - Pulse (16 bits value): 0
 - Output compare preload: Enable
 - Fast Mode: Disable
 - CHN Polarity: High
 - CHN Idle State: Reset
- PWM Generation Channel 3N
 - Mode: PWM mode 1
 - Pulse (16 bits value): 0
 - Output compare preload: Enable
 - Fast Mode: Disable
 - CHN Polarity: High
 - CHN Idle State: Reset

Figura 94: STM32CubeIDE, menú de configuración de los timers del microcontrolador.

Finalmente este es el código empleado en la secuencia de demostración. Esta consiste en:

1. Desactivar los servomódulos y extender los pies exteriores.
2. Activar los servomódulos exteriores.
3. Realizar movimientos de extensión y contracción con ambos actuadores de traslación.
4. Contraer los pies exteriores hasta llegar a la posición media.
5. Hacer el cambio de imantación de los pies exteriores a los interiores.
6. Terminar de contraer los pies exteriores para que el robot repose sobre el pie interior.
7. Realizar movimientos de extensión y contracción con ambos actuadores de traslación.
8. Volver al estado 1 pero sin desactivar los servomódulos, con el fin de evitar que el robot se desprege de la pared y caiga al suelo.

Nótese que el control de la extensión y contracción se ha hecho con el uso de temporizadores y existen tiempos de espera entre las instrucciones para que dé tiempo a que se completen todos los movimientos.

```
100
101 /* Infinite loop */
102 /* USER CODE BEGIN WHILE */
103 int counter = 0;
104
105 while (counter < 1)
106 {
107     /* USER CODE END WHILE */
108
109     /* USER CODE BEGIN 3 */
110
111     //-----DEMONSTRATION SEQUENCE-----//
112
113     //SET hi state
114     TIM1->CCR2 = 500-1; // Deactivate external servos
115     TIM1->CCR3 = 500-1; // Deactivate internal servos
116     HAL_GPIO_WritePin(E1p_GPIO_Port, E1p_Pin, SET); // Activate extension signal for elevation actuator 1
117     HAL_GPIO_WritePin(E2p_GPIO_Port, E2p_Pin, SET); // Activate extension signal for elevation actuator 2
118     HAL_Delay(10000); // Wait 10 seconds to ensure full extension. (THIS TIME CAN BE
119     HAL_GPIO_WritePin(E1p_GPIO_Port, E1p_Pin, RESET); // Stop extension signal for elevation actuator 1
120     HAL_GPIO_WritePin(E2p_GPIO_Port, E2p_Pin, RESET); // Stop extension signal for elevation actuator 2
121     HAL_Delay(5000); // Wait 5 seconds to put the robot in a metal surface
122     TIM1->CCR2 = 2500-1; // Activate external servos
123     HAL_Delay(5000); // Wait 5 seconds to Check that everything is okay.
124
125     //Translation 1
126     HAL_GPIO_WritePin(T1p_GPIO_Port, T1p_Pin, SET); // Activate extension signal for translation actuator 1
127     HAL_Delay(15000); // It takes around 15 s to perform a full extension.
128     HAL_GPIO_WritePin(T1p_GPIO_Port, T1p_Pin, RESET); // Stop extension signal for translation actuator 1
129     HAL_GPIO_WritePin(T1m_GPIO_Port, T1m_Pin, SET); // Activate contraction translation actuator 1
130     HAL_Delay(15000); // It takes around 15 s to perform a full contraction.
131     HAL_GPIO_WritePin(T1m_GPIO_Port, T1m_Pin, RESET); // Contraction extension signal for translation actuator 1
132
133     //Translation 2
134     HAL_GPIO_WritePin(T2p_GPIO_Port, T2p_Pin, SET); // Activate extension signal for translation actuator 2
135     HAL_Delay(15000); // It takes around 15 s to perform a full extension.
136     HAL_GPIO_WritePin(T2p_GPIO_Port, T2p_Pin, RESET); // Stop extension signal for translation actuator 2
137     HAL_GPIO_WritePin(T2m_GPIO_Port, T2m_Pin, SET); // Activate contraction translation actuator 2
138     HAL_Delay(15000); // It takes around 15 s to perform a full contraction.
139     HAL_GPIO_WritePin(T2m_GPIO_Port, T2m_Pin, RESET); // Contraction extension signal for translation actuator 2
140
```

```

141 //SET Mid state
142 HAL_GPIO_WritePin(E1m_GPIO_Port, E1m_Pin, SET); // Activate contraction signal for elevation actuator 1
143 HAL_GPIO_WritePin(E2m_GPIO_Port, E2m_Pin, SET); // Activate contraction signal for elevation actuator 2
144 HAL_Delay(3000); // Wait 4 seconds to reach the middle position
145 HAL_GPIO_WritePin(E1m_GPIO_Port, E1m_Pin, RESET); // Stop extension signal for elevation actuator 1
146 HAL_GPIO_WritePin(E2m_GPIO_Port, E2m_Pin, RESET); // Stop extension signal for elevation actuator 2
147 TIM1->CCR3 = 2500-1; // Activate internal servos
148 HAL_Delay(5000); // Wait 5 seconds to make sure the robot is secured
149
150 //SET Low state
151 TIM1->CCR2 = 500-1; // Deactivate external servos
152 HAL_Delay(5000); // Wait 5 seconds to make sure the robot is attached only by the external servos
153 HAL_GPIO_WritePin(E1m_GPIO_Port, E1m_Pin, SET); // Activate contraction signal for elevation actuator 1
154 HAL_GPIO_WritePin(E2m_GPIO_Port, E2m_Pin, SET); // Activate contraction signal for elevation actuator 2
155 HAL_Delay(8000); // Wait 8 seconds to reach the lower position
156 HAL_GPIO_WritePin(E1m_GPIO_Port, E1m_Pin, RESET); // Stop contraction signal for elevation actuator 1
157 HAL_GPIO_WritePin(E2m_GPIO_Port, E2m_Pin, RESET); // Stop contraction signal for elevation actuator 2
158
159 //Translation 1
160 HAL_GPIO_WritePin(T1p_GPIO_Port, T1p_Pin, SET); // Activate extension signal for translation actuator 1
161 HAL_Delay(15000); // It takes around 15 s to perform a full extension.
162 HAL_GPIO_WritePin(T1p_GPIO_Port, T1p_Pin, RESET); // Stop extension signal for translation actuator 1
163 HAL_GPIO_WritePin(T1m_GPIO_Port, T1m_Pin, SET); // Activate contraction translation actuator 1
164 HAL_Delay(15000); // It takes around 15 s to perform a full contraction.
165 HAL_GPIO_WritePin(T1m_GPIO_Port, T1m_Pin, RESET); // Contraction extension signal for translation actuator 1
166
167 //Translation 2
168 HAL_GPIO_WritePin(T2p_GPIO_Port, T2p_Pin, SET); // Activate extension signal for translation actuator 2
169 HAL_Delay(15000); // It takes around 15 s to perform a full extension.
170 HAL_GPIO_WritePin(T2p_GPIO_Port, T2p_Pin, RESET); // Stop extension signal for translation actuator 2
171 HAL_GPIO_WritePin(T2m_GPIO_Port, T2m_Pin, SET); // Activate contraction translation actuator 2
172 HAL_Delay(15000); // It takes around 15 s to perform a full contraction.
173 HAL_GPIO_WritePin(T2m_GPIO_Port, T2m_Pin, RESET); // Contraction extension signal for translation actuator 2
174
175 //SET Mid state
176 HAL_GPIO_WritePin(E1p_GPIO_Port, E1p_Pin, SET); // Activate extension signal for elevation actuator 1
177 HAL_GPIO_WritePin(E2p_GPIO_Port, E2p_Pin, SET); // Activate extension signal for elevation actuator 2
178 HAL_Delay(3000); // Wait 4 seconds to reach the middle position
179 HAL_GPIO_WritePin(E1p_GPIO_Port, E1p_Pin, RESET); // Stop extension signal for elevation actuator 1
180 HAL_GPIO_WritePin(E2p_GPIO_Port, E2p_Pin, RESET); // Stop extension signal for elevation actuator 2
181 TIM1->CCR2 = 2500-1; // Activate internal servos
182 HAL_Delay(5000); // Wait 5 seconds to make sure the robot is secured
183

```

```

184 //SET Hi State
185 TIM1->CCR3 = 500-1; // Deactivate internal servos
186 HAL_Delay(5000); // Wait 5 seconds to make sure the robot is attached only by the external servos
187 HAL_GPIO_WritePin(E1p_GPIO_Port, E1p_Pin, SET); // Activate extension signal for elevation actuator 1
188 HAL_GPIO_WritePin(E2p_GPIO_Port, E2p_Pin, SET); // Activate extension signal for elevation actuator 2
189 HAL_Delay(8000); // Wait 8 seconds to reach the middle position
190 HAL_GPIO_WritePin(E1p_GPIO_Port, E1p_Pin, RESET); // Stop extension signal for elevation actuator 1
191 HAL_GPIO_WritePin(E2p_GPIO_Port, E2p_Pin, RESET); // Stop extension signal for elevation actuator 2
192
193 counter +=1; //Increment counter so while loop stops. Reset is needed to start the sequence
194 }
195 /* USER CODE END 3 */
196
197

```

4.8 ROBOT COMPLETO

Finalmente, en este apartado se muestra el nuevo prototipo del Xrobin, Xrobin 2, ya completado. Las dimensiones finales son de 390 x 115 x 125 mm y el peso total es de 2018 g. El CAD indicaba 1812 g, sin contar la electrónica.

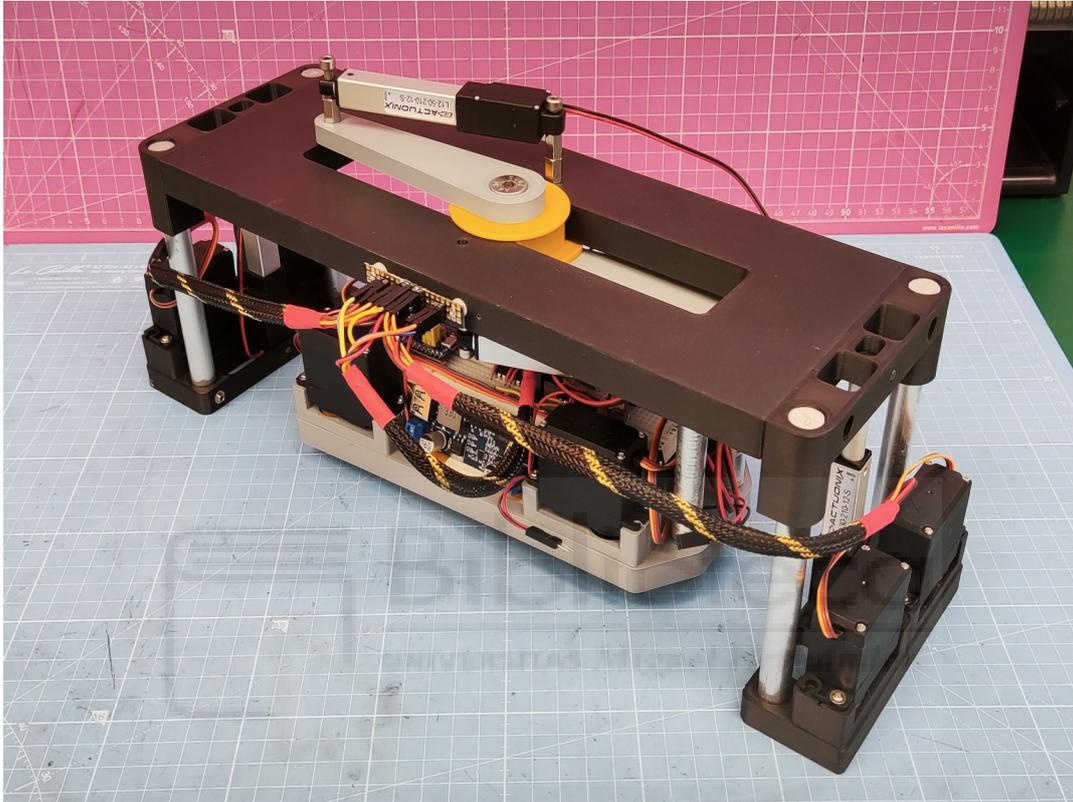


Figura 95: Xrobin 2.

En la siguiente tabla se hace un conteo de la cantidad de componentes mecánicos distintos que han sido necesarios y el conteo total.

Tabla 8: Resumen de la lista de materiales mecánicos

Tipo de pieza	Distintos	Usados
Fabricados	16	58
Comprados	19	114
Total	37	172

Esta es la vista frontal del robot en el estado elevado y la configuración de traslación (g) de la Figura 6.

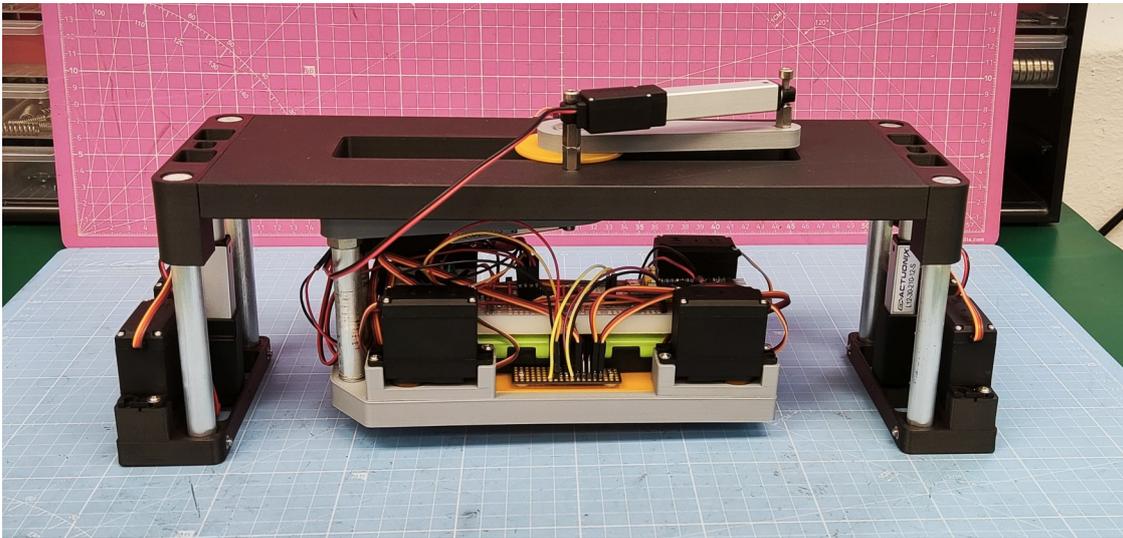


Figura 96: Vista frontal del Xrobin 2.

En la vista lateral, se puede ver que los cables están colocados de tal forma que no interfieren con el movimiento del balancín. Una propuesta de mejora sería incluir una PCB alargada en el lateral para evitar que los cables se puedan enganchar con objetos externos o con el propio robot durante el desplazamiento.

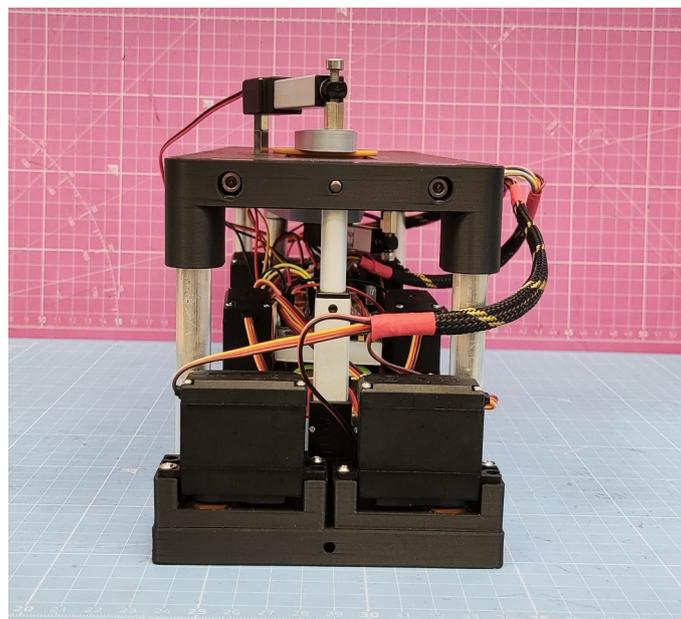


Figura 97: Vista lateral del Xrobin 2.

Para demostrar que el prototipo funciona como se esperaba, se han tomado fotografías del robot en las diferentes configuraciones programadas en la secuencia de demostración usando un carro de herramientas con chapa de acero de menos de 1mm de espesor.

En general, el comportamiento del robot es el que se esperaba para este segundo prototipo. Sin embargo, cabe mencionar varias limitaciones que se han detectado durante estas pruebas:

- El juego entre los actuadores lineales y los apoyos causa desviaciones en la posición apreciables y algunos movimientos bruscos especialmente cuando se cambia la imantación del pie interior al exterior y viceversa.
- En la transición entre estados de traslación, existen puntos donde la rigidez del robot es limitada. Especialmente cuando los actuadores pasan por encima de la junta prismática-rotativa. Se han observado vibraciones y un comportamiento errático, parcialmente amplificado por el juego anterior.
- Debido a la necesidad de proveer espacio al actuador lineal inferior, hay un voladizo considerable entre el pie interior y la junta prismática-rotativa. La flexión que se produce junto al peso del robot y la longitud de este hace que cuando el pie exterior está en movimiento pueda casi llegar a rozar la pared.

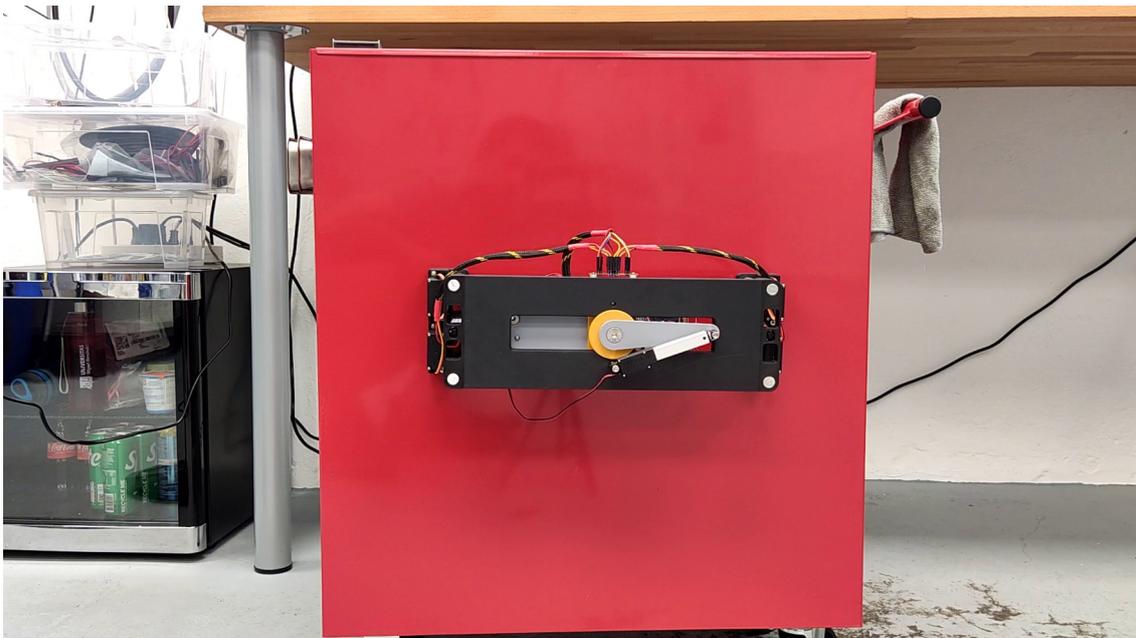


Figura 98: Xrobin 2 sobre la superficie metálica de un carro de herramientas.

En las siguientes imágenes, se observa el robot en diferentes configuraciones de traslación.

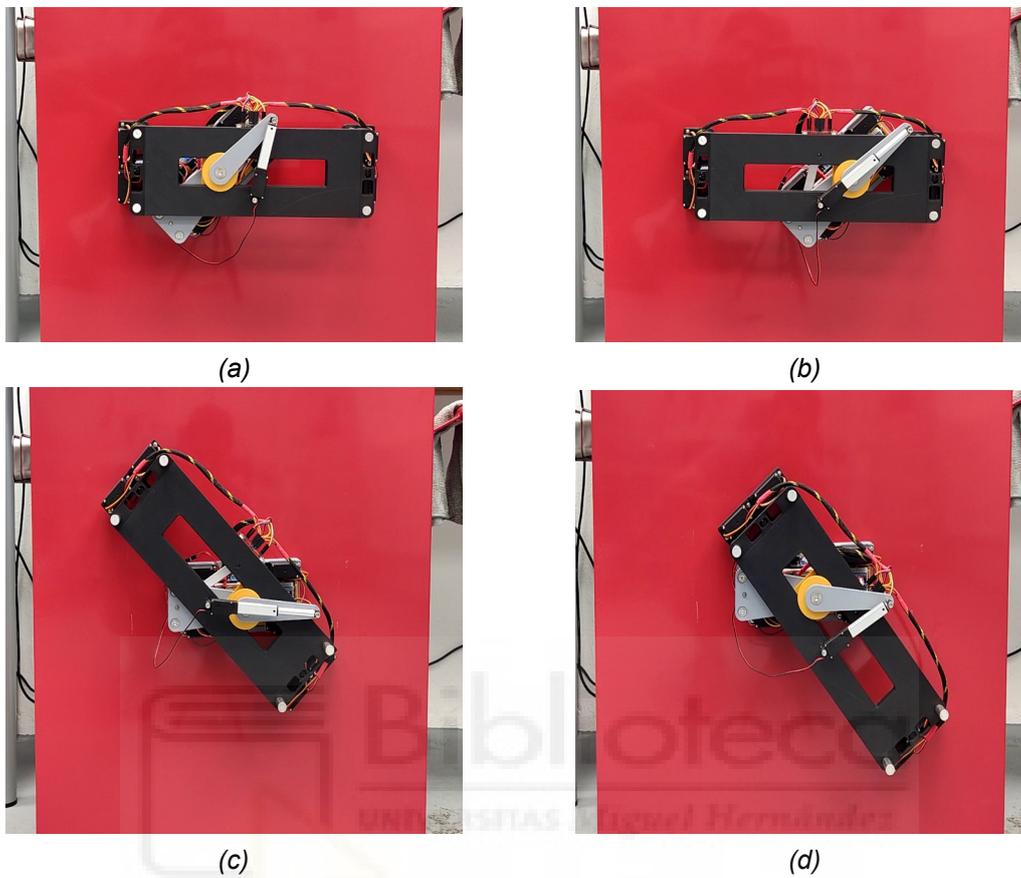


Figura 99: Xrobin 2 sobre la superficie metálica de un carro de herramientas, en varias configuraciones: X_{+01}^{HE} (a), X_{-10}^{HE} (b), X_{-10}^{LI} (c), X_{+01}^{LI} (d).

En las siguientes imágenes, se observa el robot en diferentes configuraciones de elevación.

En el estado intermedio con ambos grupos de servomódulos activados.



Figura 100: Xrobin 2 en el estado X_{-00}^{MX} .

En el estado inferior con únicamente los servomódulos interiores activados.

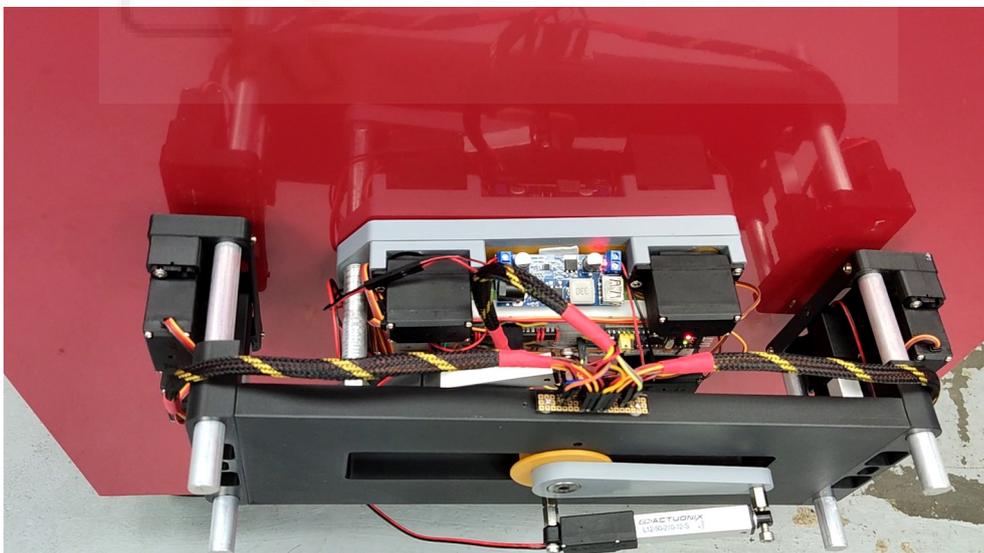


Figura 101: Xrobin 2 en el estado X_{-00}^{LI} .

5 CONCLUSIONES

En este apartado final se hace un resumen de los objetivos alcanzados en el proyecto y los trabajos futuros para continuar con el desarrollo del robot.

Respecto a los objetivos:

- Se han abordado la mayoría de las limitaciones del prototipo inicial.
- Se ha especificado minuciosamente el apartado mecánico.
- Se ha propuesto una base para el diseño de una plataforma de hardware/software escalable y transferible a otros proyectos.
- Se han fabricado las piezas necesarias para poder construir el prototipo.
- Se ha construido un prototipo funcional incluyendo mecánica, electrónica y software.
- El autor ha investigado y mejorado sus conocimientos de electrónica y software embebido.

Este trabajo se englobaría dentro del ámbito de la investigación y desarrollo de un producto en fase temprana. Debido al elevado grado de multidisciplinariedad requerida por este proyecto, en el que se requiere alcanzar soluciones en los ámbitos mecánico, electrónico e informático, no ha sido posible solucionar completamente las necesidades del prototipo en todos los ámbitos, habiéndose resuelto completamente en el ámbito mecánico pero quedando pendiente completar las dimensiones electrónica e informática/software.

En el futuro, como recomendación, sería conveniente formar un grupo de trabajo entre varios estudiantes, quizá en el contexto de los TFG (Trabajos de Fin de Grado interdisciplinares) de la UMH, para desarrollar de forma conjunta cada área del proyecto:

- Líder de proyecto: Tutor del TFG.
- Mecánica:
 - 2 personas dedicadas a la modularización y optimización de mecanismos y estructura.
- Electrónica: 2 personas.
 - 1 dedicada al hardware y la construcción del siguiente prototipo.
 - 1 persona para el desarrollo del software embebido.
- Software: 3 personas

- 1 dedicada a las interfaces de usuario.
- 1 dedicada al backend.
- 1 dedicada a las comunicaciones.

Lo más importante del desarrollo futuro de este proyecto no es solo el desarrollo de un robot, sino establecer un modelo de trabajo escalable y transferible a otro tipo de productos. Ya que una vez realizado el primer proyecto, la transferencia de conocimiento es muy rápida.

Finalmente, el autor se siente satisfecho con el trabajo realizado, los conocimientos adquiridos y espera que se le pueda dar continuidad al proyecto.



Bibliografía

- 1: Peidró, A., Tavakoli, M., Marín, J. M., & Reinoso, Ó., Design of compact switchable magnetic grippers for the HyReCRo structure-climbing robot. *Mechatronics*, 59, 199-212., 2019
- 2: Houxiang Zhang, Sky cleaner III, Consultado el 14/09/2024, <https://org.ntnu.no/intelligentsystemslab/robotics/skycleaner3.html>
- 3: Peidro, A., Garcia-Martinez, A., Marín, J. M., Paya, L., Gil, A., & Reinoso, O., Design of a mobile binary parallel robot that exploits nonsingular transitions. *Mechanism and Machine Theory*, 171, 104733, 2022
- 4: Sumitomo Heavy Industries, Successful development of magnetic wheeled robot capable of traveling over curved steel surfaces, Consultado el 14/09/2024, <https://www.shi.co.jp/english/info/2022/6kgpsq0000003be1.html>
- 5: Radialmagnets, Magnet calculator / Design, Consultado el 5/09/2024, <https://radialmagnet.com/MagCal.html>
- 6: Adam Fraser-Kruck, StateSmith Github repository, Consultado el 14/05/2024, <https://github.com/StateSmith/StateSmith>
- 7: Quantum Leaps, LLC, Using QP Framework with STM32 Cube IDE, Consultado el 22/07/2024, https://www.youtube.com/watch?v=M05jMX3_wJk&t=2s
- 8: Quantum Leaps, LLC, Hierarchical State Machines Persevere on Mars, Consultado el 14/09/2024, <https://www.youtube.com/watch?v=Gm5zoqFRp-M>

ANEXO A: Estructura del proyecto

Este proyecto está documentado de acuerdo a un sistema jerárquico que aísla cada tema del trabajo en una carpeta diferente para mejorar la organización. En estas carpetas se encuentran los archivos de proyecto necesarios para poder editar los diseños.

Este aspecto es clave para poder trabajar de forma centralizada y eficaz en proyectos multidisciplinarios.

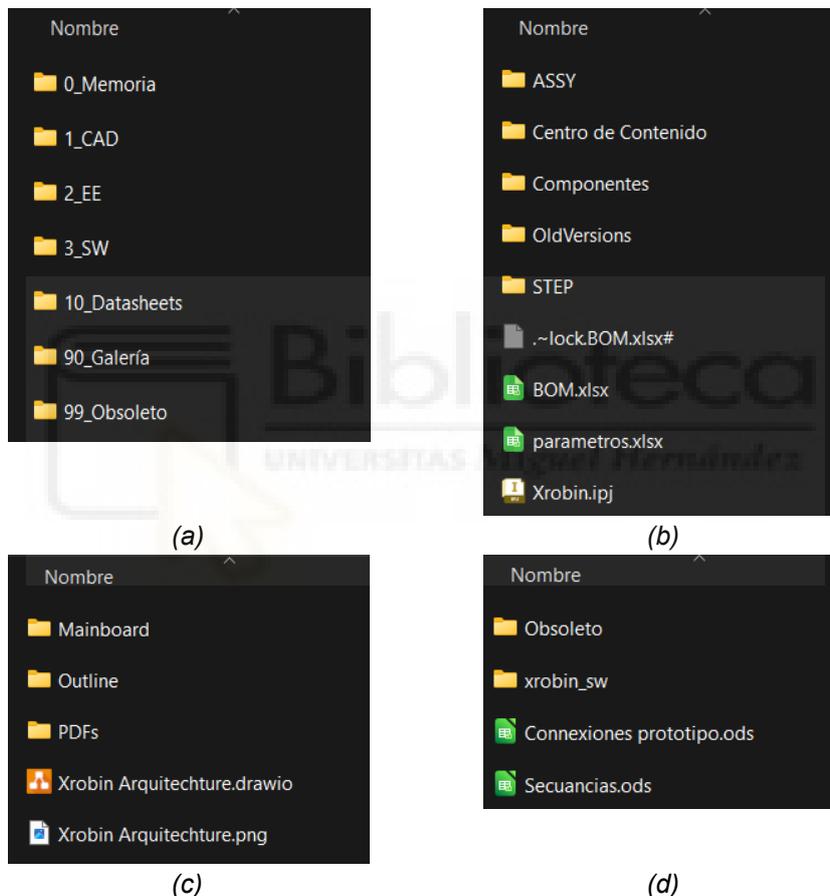


Figura 102: Carpetas del proyecto: General (a), 1_CAD (b), 2_EE (c), 3_SW (d).

ANEXO B: Lista de materiales mecánicos

Item	Part Number	Thumbnail	BOM Structure	QTY	Material	Appearance
 87	Acoplador iman-004-A		 Normal	8	ABS Plastic	Gold
 88	Acoplador servo-004-A		 Normal	8	ABS Plastic	Gold
 70	Arandela deslizadera-001-B		 Normal	1	ABS Plastic	Gold
 89	base_pie_exterior-005-A		 Normal	2	ABS Plastic	Smooth - Black
 24	Bastidor-001-B		 Normal	1	ABS Plastic	Smooth - White
 90	Cuerpo imanes 18mm-002-A		 Normal	8	Steel	Semi-Polished
 25	Deslizadera-001-B		 Normal	1	Plástico ABS	Gold
 18	Guia		 Normal	4	Aluminum 6061	Aluminum - Flat
 28	Manibela inferior-002-A		 Normal	1	ABS Plastic	Smooth - Medium Gray
 27	Manibela superior-002-A		 Normal	1	ABS Plastic	Smooth - Medium Gray
 66	Pie interior inferior-004-A		 Normal	1	ABS Plastic	Smooth - Medium Gray
 76	Powerboard-001PT-A		 Normal	1	Generic	Gold
 26	Separador Manibelas-001-B		 Normal	1	ABS Plastic	Smooth - Red

Item	Part Number	Thumbnail	BOM Structure	QTY	Material	Appearance
 78	Servo mount-001-A		 Normal	8	ABS Plastic	Smooth - Black
 19	Soporte de guias en el bastidor-002-C		 Normal	2	ABS Plastic	Smooth - Black
 84	(Unsaved)		 Purchased	1	Steel	Default
 73	Bateria 18650 Panasonic		 Purchased	4	Generic	Default
 94	Casquillo separador		 Purchased	2	Aluminum 6061	Aluminum - Flat
 98	Hold_18650-2p_smd		 Purchased	2	Generic	Default
 86	Iman Nd 18x5		 Purchased	16	Generic	Default
 80	Insert_M3		 Purchased	8	Brass, Soft Yellow	Brass - Satin
 95	Insert_M4_Short		 Purchased	14	Brass, Soft Yellow	Brass - Satin
 91	ISO 2338 - 3 h8 x 16		 Purchased	8	Stainless Steel, Austenitic	Semi-Polished
 53	ISO 2338 - 4 h8 x 32		 Purchased	4	Acero inoxidable, austenítico	Sempulido
 92	ISO 4027 - M4 x 4		 Purchased	4	Stainless Steel, 440C	Semi-Polished
 82	ISO 4762 - M3 x 10		 Purchased	8	Stainless Steel, 440C	Semi-Polished
 96	ISO 4762 - M4 x 16		 Purchased	2	Stainless Steel, 440C	Semi-Polished
 93	ISO 4762 - M4 x 20		 Purchased	6	Stainless Steel, 440C	Semi-Polished

Item	Part Number	Thumbnail	BOM Structure	QTY	Material	Appearance
 100	ISO 10642 - M6 x 20		 Purchased	4	Steel	Plate
 32	ISO 10642 - M8 x 16		 Purchased	2	Stainless Steel, 440C	Semi-Polished
 57	M8 Coupler		 Purchased	1	Steel	Galvanized
 69	MG995		 Purchased	8	Generic	Default
 74	Separador 10xM4		 Purchased	6	Steel	Semi-Polished
 68	Servo screw		 Purchased	16	Steel	Semi-Polished



ANEXO C: Esquemáticos y PCB layout

La función de este anexo es indicar que existe un proyecto de KiCad llamado “Mainboard.kicad_pro”, que contiene los esquemáticos y el editor de PCB.

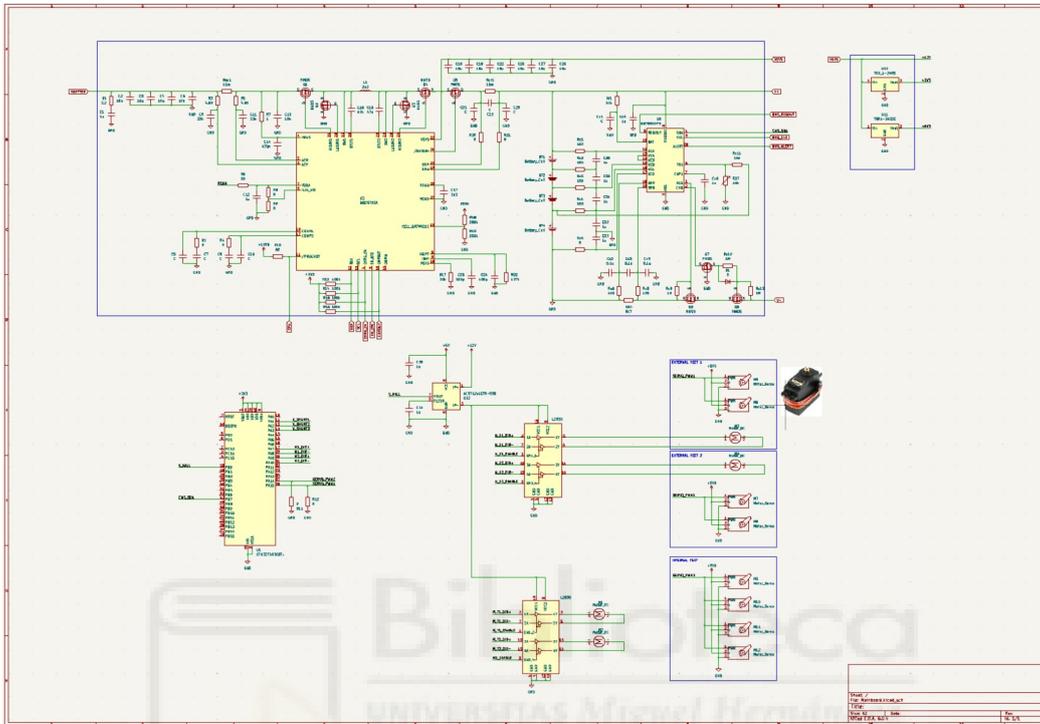


Figura 103: Imagen del esquemático de la PCB para el Xrobin 2.

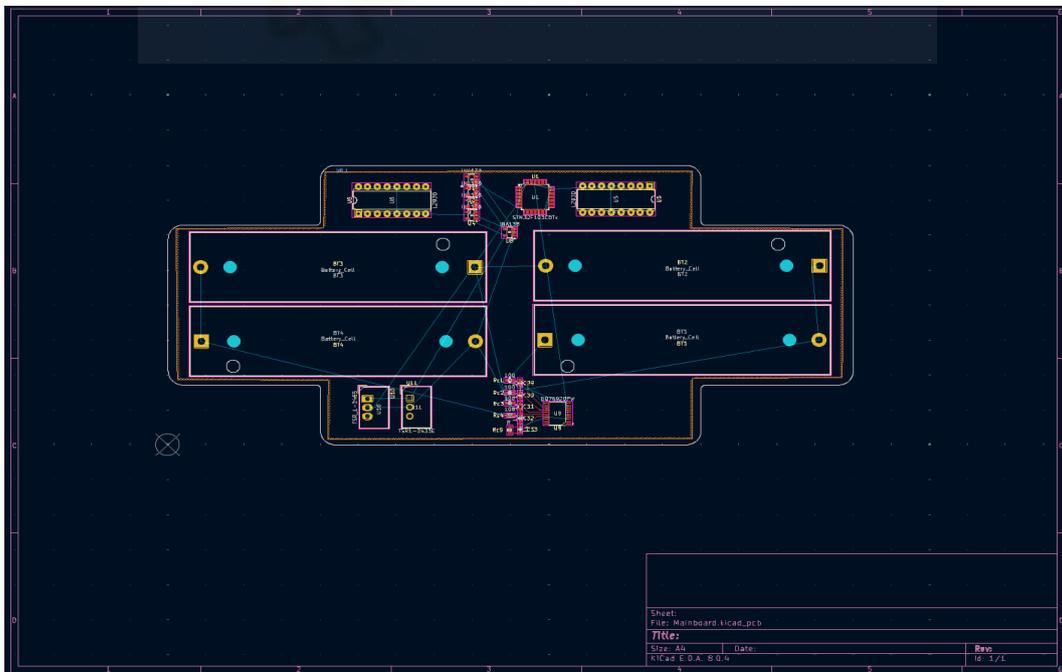


Figura 104: Imagen del editor de la PCB para el Xrobin 2.

ANEXO D: Glosario

Tabla 9: Acrónimos en inglés y español.

Siglas	Inglés	Español
ARVC	Automatización, Robótica y Visión por Computador	
BMS	Battery Management System	Sistema de gestión de baterías
DIP	Dual In-line Package	Un tipo de encapsulado
GPIO	General Purpose Input/Output	Entrada/Salida de Propósito General
HAL	Hardware abstraction layer A STM32 library	Capa de abstracción de hardware Una librería de STM32
IC	Integrated Circuit	Circuito integrado
MCU	Micro Controller Unit	Unidad microcontrolador
PD	Power delivery	Entrega de potencia
PMIC	Power Management Integrated Circuit	Circuito Integrado de Gestión de Energía
PWM	Pulse Width Modulation	Modulación por ancho de pulso
QM	Quantum Modeler	
QP	Quantum Platform	
RTEF	Real Time Embedded Framework	Marco de trabajo embebido en tiempo real
SIP	Single Inline Package	Un tipo de encapsulado
SM	State Machine	Máquina de estados
SOIC	Small Outline Integrated Circuit	Un tipo de encapsulado
TFG	Bachelor thesis	Trabajo Fin de Grado
TFM	Master thesis	Trabajo Fin de Master
TSSOP	Thin Shrink Small Outline Package	Un tipo de encapsulado
UART	Universal Asynchronous Receiver-Transmitter	Receptor-transmisor asíncrono universal
UMH	Universidad Miguel Hernandez de Elche	
USB	Universal Serial Bus	Bus serie universal
VQFM	Very Thin Quad Flat No-Lead	Un tipo de encapsulado

