



Universidad de Málaga

Escuela de Ingenierías Industriales

Departamento de Ingeniería de Sistemas y Automática

Trabajo Fin de Grado

Diseño y control de una muñeca robótica de 3 grados de libertad para un manipulador delta ligero

Grado en Ingeniería Electrónica, Robótica y Mecatrónica

Autor: David Rodríguez Onieva

Tutor: Juan Manuel Gandarias Palacios

Cotutor: Francisco Pastor Martín

30 de noviembre de 2024

Acrónimos y Notación Matemática

GDL	Grados de Libertad, refiriéndose al número de parámetros independientes que definen la configuración o el estado de un sistema mecánico.
IMU	Inertial Measurement Unit, es un sensor utilizado para detectar cambios en la aceleración y la rotación en un sistema de coordenadas.
PLA	Ácido Poliláctico, un material plástico biodegradable utilizado comúnmente en impresión 3D.
ABS	Acrilonitrilo Butadieno Estireno, un termoplástico utilizado en diversas aplicaciones de fabricación.
PETG	Glicol de Tereftalato de Polietileno, un material termoplástico que combina la facilidad de impresión del PLA con la durabilidad del ABS.
ATEX	Atmosphères Explosibles, normativa europea que regula los requisitos para equipos y sistemas de protección destinados a ser utilizados en atmósferas potencialmente explosivas.
RPAS	Remotely Piloted Aircraft Systems, sistemas de aeronaves no tripuladas controladas remotamente por un piloto desde una estación en tierra. Los RPAS se utilizan en diversas aplicaciones como vigilancia, seguridad, agricultura de precisión, inspección industrial y logística.

Índice

	Página
Índice de Figuras	vii
1 Introducción	1
1.1. Motivación y trabajos relacionados	2
1.1.1. Manipuladores Aéreos	2
1.1.2. Manipuladores paralelos y brazos delta	4
1.1.3. Avances recientes y soluciones propuestas	5
1.2. Objetivos y contribución	6
1.3. Estructura de la memoria	6
2 Metodología	9
2.1. Hardware	9
2.1.1. Diseño 3D	10
2.1.2. Impresión 3D	12
2.1.3. Guía de ensamblaje	13
2.1.4. Arduino + Sensor	16
2.2. Software	17
2.2.1. ROS	17
2.2.2. Arduino	19
3 Experimentos y Resultados	21
3.1. Protocolo de experimentación	22
3.2. Experimento 1. Control por Topic	24
3.2.1. Experimento 1.1 Rotación de Base Plate	24
3.2.2. Experimento 1.2 Rotación de Linkage	26
3.2.3. Experimento 1.3 Rotación de End Effector Mounting Plate	27
3.2.4. Experimento 1.4 Rotación conjunta de Linkage y End Effector Mounting Plate	28
3.2.5. Experimento 1.5 Rotación conjunta de Base Plate y Linkage	30
3.2.6. Experimento 1.6 Rotación de muñeca completa	31

3.3. Experimento 2. Control por IMU	32
3.4. Resultados y discusión	34
3.4.1. Control de la Orientación	34
3.4.2. Limitaciones Observadas	34
3.4.3. Discusión de los Resultados	34
4 Conclusión	37
Bibliografía	39

Índice de Figuras

Figura	Página
1.1. Dron equipado con un brazo manipulador flexible. (Fuente: RobotShop)	2
1.2. Herramientas más utilizadas como efectores finales. (Fuente: Theory of Applied Robotics)	3
1.3. Robot delta desarrollado en la Universidad de Málaga.	4
2.1. Diseño de la muñeca esférica	10
2.2. Piezas que conforman la muñeca esférica	11
2.3. Cambios efectuados en la pieza efector final del brazo delta	11
2.4. Motores Dynamixel serie X (Fuente: ROBOTIS)	12
2.5. Paso 1 del montaje de la muñeca	13
2.6. Paso 2 del montaje de la muñeca	13
2.7. Paso 3 del montaje de la muñeca	14
2.8. Paso 4 del montaje de la muñeca	14
2.9. Paso 5 del montaje de la muñeca	15
2.10. Paso 6 del montaje de la muñeca	15
2.11. Sensor inercial de medida MP6050 (Fuente: NAYLAMP MECHATRONICS)	16
2.12. Conexión realizada entre el sensor IMU y la placa Arduino (Fuente: Pinterest)	16
3.1. Muñeca esférica montada en el brazo delta	22
3.2. Conexión en serie de los motores Dynamixel (Fuente: ROBOTIS)	22
3.3. Conexión de los motores con el PC (Fuente: ROBOTIS)	23
3.4. Nodos básicos para la ejecución	23
3.5. Ejecución del nodo "ControlTopic" donde los motores quedan en espera de recibir la orientación objetivo.	24
3.6. Comando que hay que utilizar para ver qué topics están activos.	24
3.7. Rotación de Base Plate	25
3.8. Rotación de Base Plate en onshape	25
3.9. Posición angular marcada por los motores a lo largo del experimento 1.1	25
3.10. Rotación de Linkage a 310º	26
3.11. Rotación de Linkage a 310º en Onshape	26

3.12. Posición angular marcada por los motores a lo largo del experimento 1.2	27
3.13. Rotación de End Effector Mounting Plate a 300°	27
3.14. Rotación de End Effector Mounting Plate a 300° en onshape	28
3.15. Posición angular marcada por los motores a lo largo del experimento 1.3	28
3.16. Rotación de Linkage y End Effector Mounting Plate	29
3.17. Rotación de Linkage y End Effector Mounting Plate en Onshape	29
3.18. Posición angular marcada por los motores a lo largo del experimento 1.4	29
3.19. Rotación de Base Plate y Linkage	30
3.20. Rotación de Base Plate y Linkage en Onshape	30
3.21. Posición angular marcada por los motores a lo largo del experimento 1.5	31
3.22. Rotación de la muñeca esférica de manera conjunta	31
3.23. Rotación de la muñeca esférica de forma simultánea en Onshape	32
3.24. Posición angular marcada por los motores a lo largo del experimento 1.6	32
3.25. Datos tomados por el sensor publicados en sus respectivos topics.	33
3.26. Orientación de los motores vs orientación del sensor inercial.	33

Introducción

Contenido

1.1. Motivación y trabajos relacionados	2
1.1.1. Manipuladores Aéreos	2
1.1.2. Manipuladores paralelos y brazos delta	4
1.1.3. Avances recientes y soluciones propuestas	5
1.2. Objetivos y contribución	6
1.3. Estructura de la memoria	6

En este capítulo se exploran los avances recientes en el campo de la robótica aérea, centrándose particularmente en los manipuladores aéreos. Se contextualiza el desarrollo de tecnologías que permiten a estos sistemas operar de manera efectiva en entornos desafiantes, abordando los problemas de estabilidad y control de orientación que presentan. Además, se revisan investigaciones y proyectos, como el proyecto AEROARMS, que han contribuido significativamente al avance de la robótica aérea, estableciendo un marco para las soluciones propuestas en este ámbito.

A medida que se avanza en el capítulo, se presentan soluciones innovadoras diseñadas para mejorar la funcionalidad de los manipuladores aéreos. Entre estas, destaca el desarrollo de una muñeca robótica que añade grados de libertad al manipulador, así como la integración de tecnologías de estabilización inspiradas en estabilizadores de cámaras. Estos enfoques no solo buscan optimizar la estabilidad y precisión en las tareas de manipulación, sino que también ofrecen una visión clara

de las direcciones futuras para la investigación y el desarrollo en la robótica aérea, asegurando que los sistemas sean más eficientes y versátiles en su aplicación práctica.

1.1. Motivación y trabajos relacionados

En la actualidad, la robótica se ha convertido en un elemento indispensable para mejorar la eficiencia y el rendimiento en una amplia variedad de ámbitos. La automatización de procesos a través de sistemas robóticos ha permitido optimizar tareas repetitivas, reducir errores humanos y aumentar la productividad. Este progreso ha llevado a la integración de manipuladores robóticos en múltiples sectores [1].

1.1.1. Manipuladores Aéreos

El avance de la robótica ha dado lugar a un área que presenta desafíos únicos: los manipuladores aéreos. Estos sistemas, montados en plataformas voladoras como drones o multi-rotores, se utilizan para realizar tareas en el aire, tales como la inspección de infraestructuras, la manipulación de objetos en lugares de difícil acceso y aplicaciones de rescate. A pesar de su potencial, los manipuladores aéreos enfrentan retos significativos derivados de la inestabilidad inherente de los dispositivos y las fuerzas dinámicas que actúan sobre ellos durante sus operaciones. Un ejemplo de manipulador aéreo es el de la figura 1.1



Figura 1.1: Dron equipado con un brazo manipulador flexible. (Fuente: [RobotShop](#))

A lo largo de los años, se han realizado investigaciones y pruebas sobre los manipuladores aéreos. Un ejemplo notable fue el proyecto AEROARMS, liderado por Aníbal Ollero, profesor emérito en la Escuela Técnica Superior de Ingeniería de la Universidad de Sevilla [2]. Este proyecto desarrolló drones equipados con brazos robóticos para la inspección y mantenimiento de plantas industriales. Estos robots aéreos fueron diseñados para operar en entornos complejos, permitiendo una interacción activa y manipulación de objetos en espacios desorganizados. Los principales desafíos del proyecto incluyeron la eficiencia de los drones para su uso industrial, la compatibilidad y funcionalidad de los brazos robóticos, así como la gestión de energía y control en tiempo real. El sistema fue validado en refinerías y fábricas de cemento, mostrando mejoras significativas en eficiencia y costos de mantenimiento [3].

Además, los resultados del proyecto AEROARMS tuvieron un impacto significativo en empresas que ofrecen soluciones de alto costo para la inspección y mantenimiento en la industria del petróleo y gas. El sistema fue diseñado y desarrollado considerando regulaciones internacionales como ATEX y RPAS, y fue validado en aplicaciones piloto en refinerías y fábricas de cemento, demostrando mejoras en eficiencia y reducción de costos de mantenimiento.

Uno de los aspectos más críticos en el diseño de manipuladores aéreos es el control de la orientación del efector final. La naturaleza dinámica de los drones implica que su base está en constante movimiento, lo que puede generar perturbaciones no deseadas en la orientación del efector final. Este problema es especialmente crítico en sistemas que disponen de un número limitado de grados de libertad, restringiendo su capacidad para corregir estas perturbaciones y mantener la estabilidad del efector final. La implementación de un control preciso de la orientación es esencial para asegurar la precisión y estabilidad en las tareas de manipulación aérea [4].

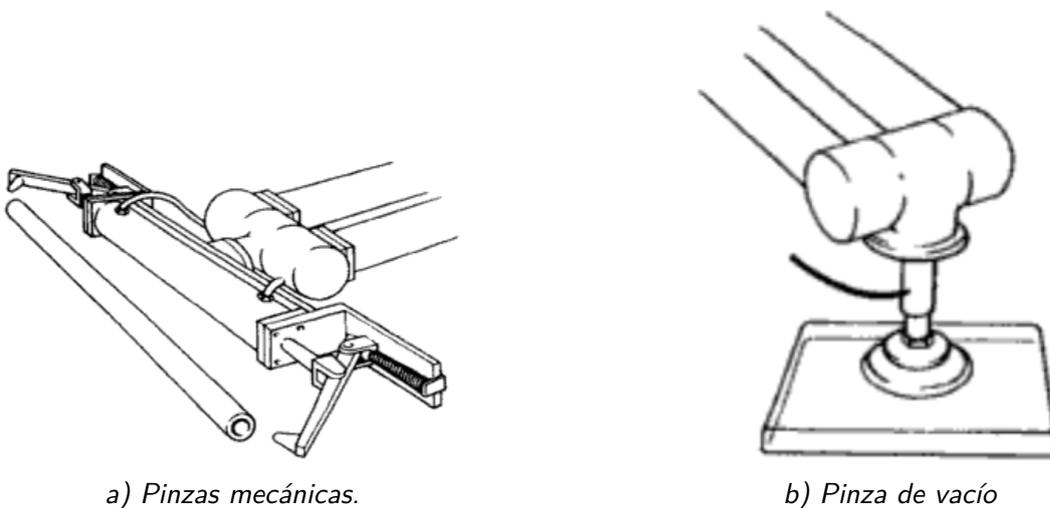


Figura 1.2: Herramientas más utilizadas como efectores finales. (Fuente: [Theory of Applied Robotics](#))

El efector final es el componente que se sitúa en el extremo del último enlace del manipulador y proporciona la capacidad de agarre del objeto a manipular o permite la colocación de herramientas

adecuadas para la tarea, como pintura o soldadura. Los efectores finales más simples son las pinzas mecánicas, que generalmente cuentan con dos dedos y utilizan un sistema de accionamiento neumático, como se muestra en la figura 1.2. En algunas aplicaciones, se emplean también accionamientos eléctricos que permiten un control proporcional sobre la fuerza de agarre [4].

1.1.2. Manipuladores paralelos y brazos delta

En este proyecto nos vamos a centrar en los manipuladores paralelos que van acoplados en los drones, es decir, en los manipuladores paralelos aéreos. Un robot de cinemática paralela, también llamado robot de cadena cerrada o manipulador paralelo, consiste básicamente en una plataforma móvil unida a una base fija por medio de varios brazos. Típicamente, cada brazo está controlado por un actuador. En general, estos robots paralelos pueden manipular una carga mayor que los robots de cadena abierta, ya que comparten la carga entre varios brazos paralelos.

Un ejemplo de manipulador paralelo es el robot delta. Se caracteriza por tener varios brazos móviles conectados a una estructura central, formando una configuración en forma de paralelogramo. Esta configuración permite que los brazos del robot se muevan de manera sincronizada, proporcionando una amplia gama de movimientos en tres dimensiones. Por ello, consta de 3 GDL con los que se controla la posición [5].



Figura 1.3: Robot delta desarrollado en la Universidad de Málaga.

El robot delta de la figura 1.3 ha sido diseñado en el Departamento de Sistemas y Automática de la Universidad de Málaga. Como se puede observar, está compuesto por varios eslabones y articulaciones, lo que le permite manipular objetos dentro de un espacio de trabajo tridimensional.

En el brazo delta utilizado en este proyecto, se emplean tres motores Dynamixel para controlar los eslabones del brazo. Estos motores contribuyen a la capacidad del manipulador para realizar movimientos precisos.

1.1.3. Avances recientes y soluciones propuestas

En los últimos años, el campo de la robótica ha experimentado avances significativos, especialmente en la optimización de los manipuladores aéreos y su capacidad para realizar tareas complejas en entornos desafiantes. Una de las áreas más destacadas es el desarrollo de tecnologías que mejoran el control de la orientación del efector final, esencial para lograr precisión y estabilidad en la manipulación de objetos.

Un enfoque innovador ha sido la implementación de muñecas robóticas que proporcionan grados de libertad adicionales, permitiendo a los manipuladores aéreos ajustar la orientación de sus efectores finales en tiempo real. Esto se traduce en una mejora notable en la capacidad de respuesta ante perturbaciones durante el vuelo, lo que es crítico para realizar tareas delicadas como la colocación de objetos o la manipulación de herramientas en condiciones variables.

Un ejemplo de este avance se encuentra en el diseño de efectores finales articulados para aplicaciones quirúrgicas. Estos dispositivos, utilizados en procedimientos como la cirugía endoscópica [6], permiten a los cirujanos acceder a áreas difíciles con una mayor precisión que las herramientas rígidas tradicionales. Recientes investigaciones han mostrado el desarrollo de un efector terminal robótico de 3 grados de libertad, que ha demostrado una repetibilidad prometedora y un espacio de trabajo efectivo, lo que resalta la aplicabilidad de estas tecnologías en entornos que requieren un alto grado de control.

Además, se han explorado mecanismos de estabilización inspirados en tecnologías de estabilizadores de cámaras, los cuales son cruciales para mantener la estabilidad en la grabación de imágenes. Al aplicar principios similares en la robótica aérea, se busca mejorar la estabilidad de los manipuladores montados en drones, utilizando sensores y algoritmos de control para contrarrestar las perturbaciones generadas durante el vuelo. Esta integración no solo optimiza el rendimiento general del manipulador, sino que también aumenta su eficacia en tareas complejas.

Una alternativa a las muñecas robóticas tradicionales es el mecanismo de junta esférica activa (Active Ball Joint Mechanism) [7], que ofrece un rango de movimiento superior y mejora la adaptabilidad en la manipulación de objetos. La investigación sobre este mecanismo ha mostrado su efectividad en la manipulación precisa en entornos quirúrgicos, resaltando su capacidad para mejorar la interacción con el entorno y reducir el riesgo de daño a los objetos manipulados.

En conclusión, los recientes avances en el diseño y control de manipuladores aéreos, junto con la integración de tecnologías innovadoras, prometen un futuro brillante para la robótica aérea,

ampliando sus aplicaciones en diversos campos y mejorando la capacidad de estos sistemas para operar en entornos complejos y dinámicos.

1.2. Objetivos y contribución

Este proyecto se centra en resolver el desafío del control de la orientación del efector final de un brazo delta. Actualmente, este manipulador aéreo cuenta con un sistema de control de posición que, al estar limitado a solo tres grados de libertad (GDL), restringe su capacidad para reaccionar a las perturbaciones causadas por el movimiento del dron además de limitar las tareas que puede realizar. Para solucionar esta situación, se propone el diseño y control de una muñeca robótica que añada grados de libertad adicionales al manipulador, permitiendo así un control de la orientación del efector final.

Para lograr este objetivo, se abordan los siguientes tres puntos clave:

Diseño mecánico

- Se buscará una solución que integre tres grados de libertad adicionales al manipulador existente. Esto es fundamental para ampliar la capacidad del sistema y permitir un control de la orientación del efector final. Es crucial que el diseño mantenga un equilibrio adecuado entre funcionalidad y peso, ya que el manipulador estará montado en un dron que tiene restricciones significativas en cuanto a la carga útil y la estabilidad durante el vuelo.

Sistema de control

- El desarrollo de un sistema de control eficaz es esencial para integrar los nuevos grados de libertad de la muñeca. Se requerirá la implementación de algoritmos avanzados que faciliten el control en tiempo real, lo que asegurará que el manipulador pueda adaptarse a las condiciones cambiantes durante las tareas de manipulación.

Integración con el Manipulador Existente

- Por último, será necesario llevar a cabo la integración de la muñeca robótica con el brazo delta.

1.3. Estructura de la memoria

Tras dar una introducción, se detalla la estructura seguida en la memoria que presenta el desarrollo del proyecto.

En primer lugar, se presenta la **Metodología**. En esta sección se detallan el enfoque y los procedimientos seguidos para el diseño, desarrollo y control de la muñeca robótica, incluyendo las técnicas y herramientas utilizadas durante el proceso.

Posteriormente, se presentan los **Experimentos y Resultados**. Aquí se detallan las comprobaciones experimentales realizadas para evaluar el desempeño del sistema desarrollado, analizando los resultados obtenidos y comparándolos con los objetivos establecidos inicialmente.

Finalmente, se exponen las **Conclusiones**. Esta sección destaca las contribuciones del proyecto, discute las implicaciones de los resultados obtenidos y sugiere posibles direcciones para futuras investigaciones.

Con esta estructura, se pretende ofrecer una visión clara y detallada de cada etapa del proyecto, desde su concepción hasta la evaluación final, asegurando que el lector pueda seguir y entender el desarrollo completo del trabajo.

Metodología

Contenido

2.1. Hardware	9
2.1.1. Diseño 3D	10
2.1.2. Impresión 3D	12
2.1.3. Guía de ensamblaje	13
2.1.4. Arduino + Sensor	16
2.2. Software	17
2.2.1. ROS	17
2.2.2. Arduino	19

En este capítulo se detalla el enfoque metodológico adoptado para el desarrollo del proyecto estableciendo las bases para comprender el proceso de diseño, implementación y control del sistema. La metodología abarca dos aspectos fundamentales: Hardware y Software, los cuales están interrelacionados y son cruciales para el éxito del sistema.

2.1. Hardware

La sección de Hardware presenta los componentes físicos utilizados en el sistema, destacando su selección e integración. Se detallan los motores Dynamixel y el sensor inercial (IMU) modelo MP6050, así como el proceso de montaje y configuración, garantizando un funcionamiento óptimo y eficiente del sistema.

2.1.1. Diseño 3D

Para el desarrollo del diseño 3D, se utilizó Onshape, una plataforma de diseño 3D basada en la web que permite la accesibilidad desde cualquier dispositivo y ofrece un entorno colaborativo gratuito.

El proceso de diseño fue iterativo; tras crear el modelo, se realizaron simulaciones de movimiento para evaluar el rango de movimiento de la muñeca esférica. Estas simulaciones optimizaron la posición de los motores y ayudaron a identificar las áreas críticas que necesitaban refuerzos, mejorando así la estabilidad del sistema.

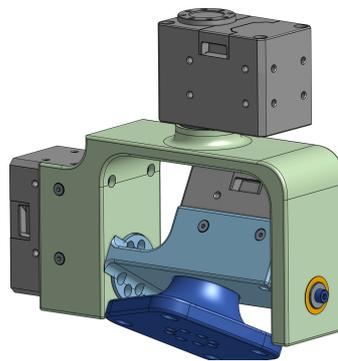


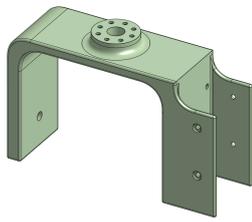
Figura 2.1: Diseño de la muñeca esférica

La muñeca se basa en el diseño presentado en el curso de cinemática directa para manipuladores [8].

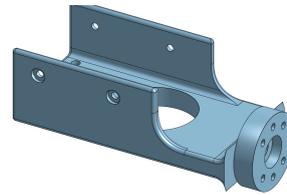
La muñeca esférica se compone de tres piezas, cada una conectada a un motor, y se fija al efector final del brazo delta. La configuración de los motores permite que los ejes de giro intersecten en un único punto, conocido como punto de rotación o punto muñeca. Esta disposición facilita el movimiento coordinado de las piezas, permitiendo a cada motor contribuir sinérgicamente a los movimientos del efector final y alcanzar posiciones angulares complejas.

Cada pieza proporciona un grado de libertad adicional (GDL) al sistema, sumando un total de tres GDL que contribuyen al movimiento del efector final. A continuación, se describen las funciones de cada pieza:

- **Base de la Muñeca (Base Plate):** Asegura el montaje correcto del motor y la pieza de conexión, evitando desplazamientos no deseados.
- **Eslabón de Conexión (Linkage):** Actúa como enlace entre la base y la placa de montaje del efector, permitiendo el movimiento en el eje de giro.
- **Placa de Montaje del Efector (End Effector Mounting Plate):** Soporta el efector final; su diseño incluye orificios para un montaje firme.



a) Base de la Muñeca (Base Plate)



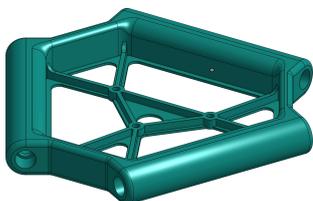
b) Eslabón de Conexión (Linkage)



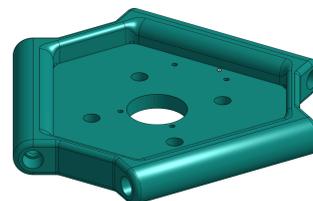
c) Placa de Montaje del Efecto (End Effector Mounting Plate)

Figura 2.2: Piezas que conforman la muñeca esférica

Las dimensiones de las piezas que se ven en la figura 2.2 están disponibles en los planos 3, 4 y 5 del apéndice. Para fijar la muñeca al brazo delta, se rediseñó la pieza "efector final" original, donde se acoplaban las pinzas o herramientas según la tarea a realizar. La modificación incluye la incorporación de material para tener una base estable donde colocar el motor 1, que se atornilla en esta pieza. ¹



a) Estado inicial



b) Versión actualizada

Figura 2.3: Cambios efectuados en la pieza efector final del brazo delta

Después de realizar el nuevo diseño, se imprimió nuevamente y se colocó en el brazo delta, sustituyendo la versión anterior. La diferencia entre la pieza original y la modificada se puede

¹Para más detalles sobre la pieza efector del brazo delta, se puede consultar [en este enlace](#).

observar en la figura 2.3. Los principales cambios incluyen la adición de material para crear una base estable donde apoyar el motor, así como la incorporación de orificios para permitir el paso del eje de giro del motor y huecos para atornillar el motor 1 a esta pieza.

Para el funcionamiento de la muñeca, se seleccionaron motores Dynamixel modelo XM430-W210. En la figura 2.4 se puede ver cómo son los motores de la serie X. Estos motores son actuadores inteligentes desarrollados por la empresa ROBOTIS y se caracterizan por sus elevadas especificaciones técnicas.



Figura 2.4: Motores Dynamixel serie X (Fuente: [ROBOTIS](#))

Los motores Dynamixel XM430-W210 ofrecen un alto par, lo que significa que pueden ejercer una fuerza considerable en su eje de rotación, permitiendo mover cargas más pesadas o realizar movimientos más complejos. Están fabricados con materiales de alta calidad, lo que garantiza una larga vida útil. Para más información sobre estos motores, incluyendo su software, montaje y especificaciones, se puede consultar su manual en línea en [XM430-W210 e-Manual](#).

Finalmente, si se desea replicar este trabajo, se puede acceder al espacio de trabajo que contiene los archivos ".STL" de las piezas a través del siguiente enlace: [Onshape](#).

2.1.2. Impresión 3D

El peso de la muñeca esférica es de 360 gramos, lo que se encuentra dentro del payload del manipulador aéreo, que es de 5 kg. Tras el diseño, las piezas fueron obtenidas mediante una impresora 3D. Existen una amplia variedad de materiales, entre los que se encuentran el Acrilonitrilo Butadieno Estireno (ABS), el Glicol de Tereftalato de Polietileno (PETG) y el ácido poliláctico (PLA) [9]. Se optó por utilizar PLA por varias razones, incluyendo su facilidad de impresión, bajo costo y buena resistencia mecánica.

Aunque se pueden considerar alternativas como ABS o PETG, que ofrecen diferentes propiedades mecánicas y térmicas, el PLA es adecuado para prototipos y piezas no estructurales debido a su rigidez y facilidad de uso, lo que lo convierte en una opción ideal para este tipo de aplicaciones.

El uso de la impresión 3D no solo permite la creación de geometrías complejas y personalizadas, sino que también facilita la iteración rápida en el diseño, permitiendo ajustes según sea necesario durante el proceso de desarrollo. La impresora utilizada es la que se encuentra en el taller 27 de la Escuela de Ingenierías Industriales en Málaga. Es el modelo Prusa MK3S+.

2.1.3. Guía de ensamblaje

444

Para poder montar la muñeca de forma correcta, vamos a seguir los siguientes pasos:

1. Colocar el motor 1 sobre la pieza efector tal y como muestra la figura 2.5 y atornillar el motor 1 con 4 tornillos M2 5X12.

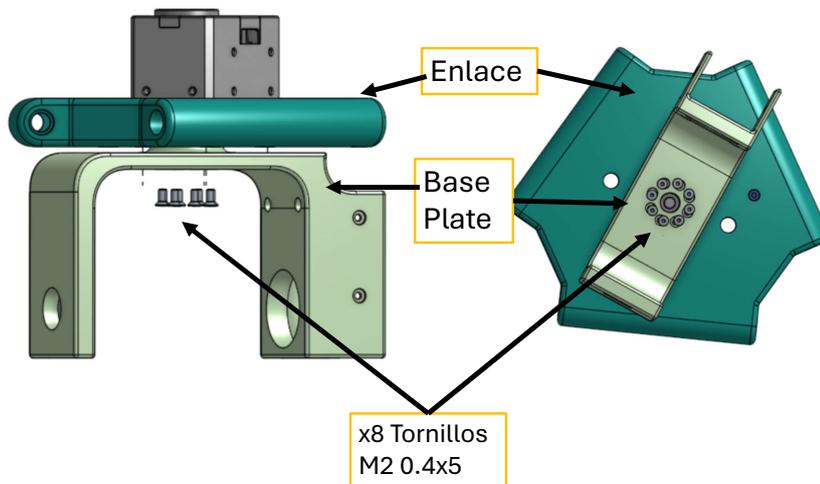


Figura 2.5: Paso 1 del montaje de la muñeca

2. Colocar la pieza Base Plate en Efector tal y como muestra la figura 2.6 y fijarla mediante 8 tornillos M2 0.4X5.

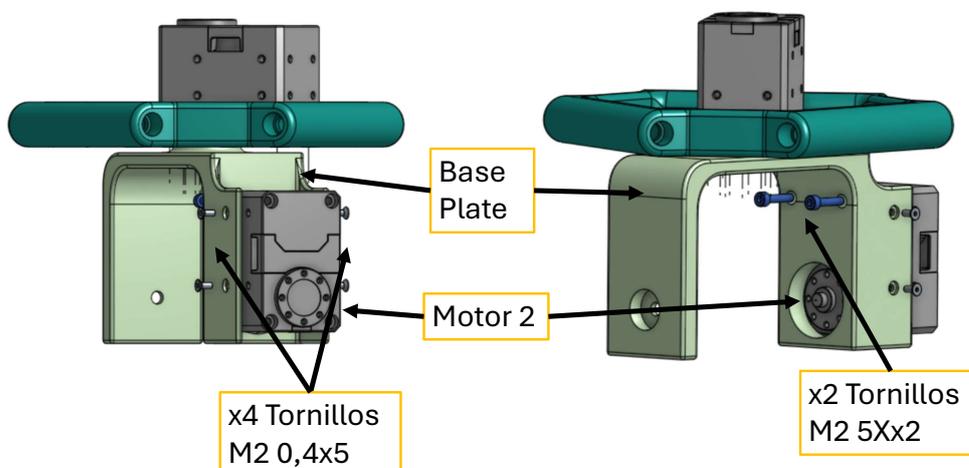


Figura 2.6: Paso 2 del montaje de la muñeca

3. Colocar el motor 2 en Base Plate tal y como muestra la figura 2.7 y fijarlo mediante 4 tornillos M2 0.4X5 y 2 tornillos M2 5X12.

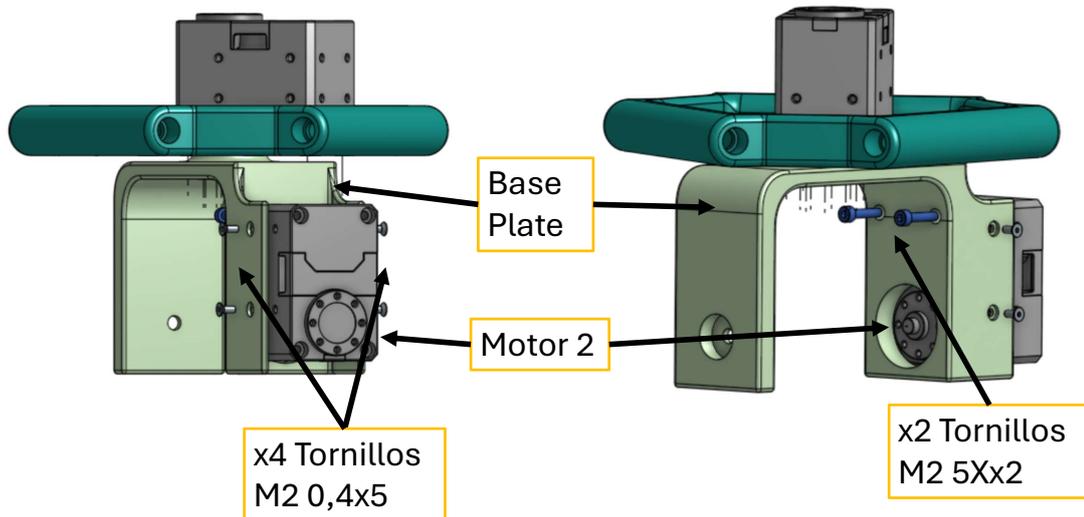


Figura 2.7: Paso 3 del montaje de la muñeca

4. Colocar la pieza Linkage y un rodamiento de dimensión 8x22x7 mm en Base Plate tal y como se muestra en la figura 2.8 y fijarlas mediante un tornillo M2 5X18, una arandela M4 y 6 tornillos M2 0.4X5.

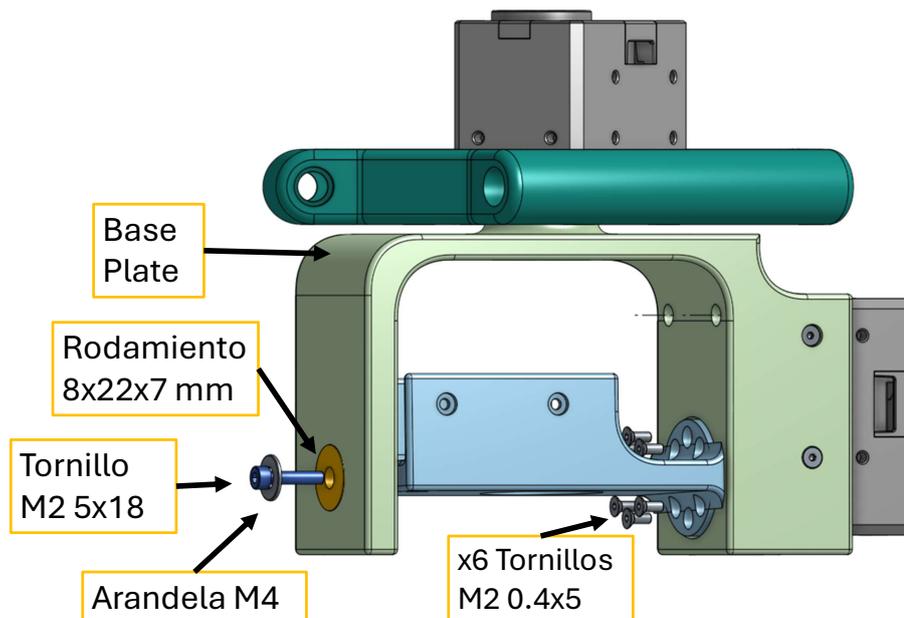


Figura 2.8: Paso 4 del montaje de la muñeca

- Colocar el motor 3 en Linkage tal y como se muestra en la figura 2.9 y fijarlo mediante 2 tornillos M2 0,4x5 y 4 tornillos M2 5x12.

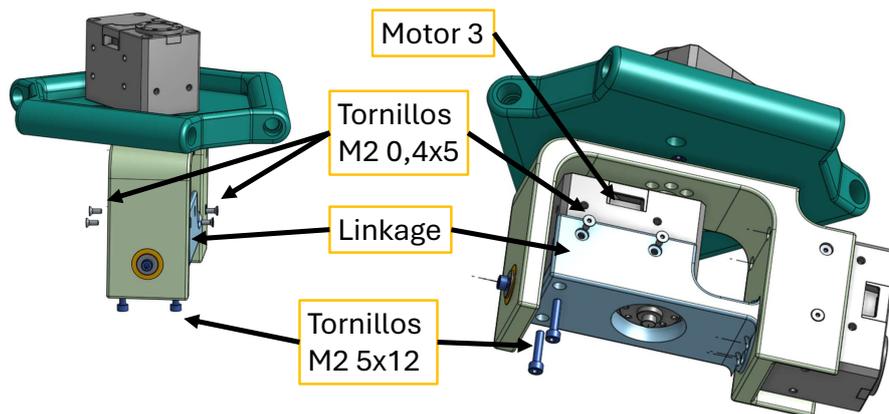


Figura 2.9: Paso 5 del montaje de la muñeca

- Colocar la pieza End Effector Mounting Plate en Linkage tal y como se muestra en la figura 2.10 y fijarla mediante 8 tornillos M2 0.4X5.

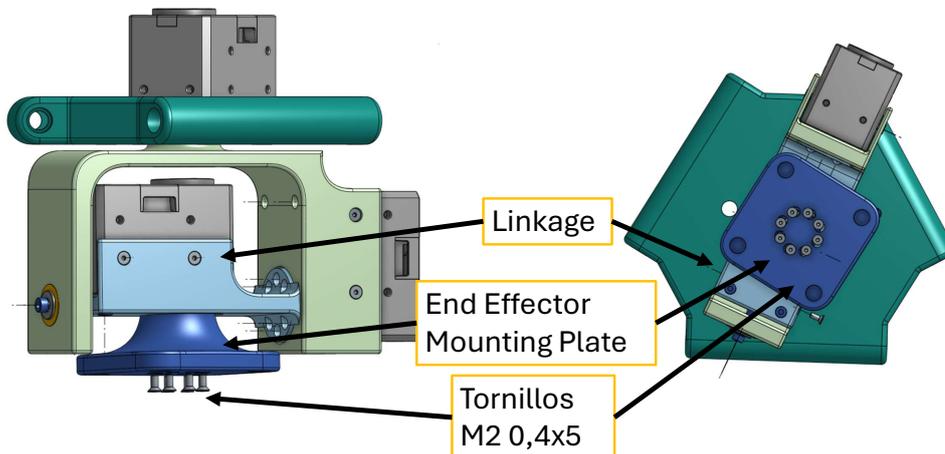


Figura 2.10: Paso 6 del montaje de la muñeca

Una vez ensamblados los motores, las piezas y los tornillos, la muñeca adopta la forma mostrada en el plano 1 del apéndice. Además, el plano 2 proporciona una vista desglosada de todas las piezas necesarias para el montaje de la muñeca esférica, junto con sus ubicaciones en el ensamblaje. En

este plano, no se considera la pieza efector del brazo delta, ya que solo se muestran las piezas de la muñeca esférica; el efector es donde se enlaza.

2.1.4. Arduino + Sensor

Para uno de los experimentos, más concretamente el segundo, se ha utilizado un sensor de medición inercial (IMU) modelo MP6050, junto con una placa Arduino Nano conectados cómo viene en la figura 2.12. La elección de la placa Arduino Nano se debe a su tamaño compacto, lo que facilita su manipulación y conexión durante los experimentos. La Arduino Nano es una pequeña placa de desarrollo basada en un microcontrolador ATmega328P, que proporciona múltiples pines de entrada/salida (I/O) y es compatible con diversos sensores y módulos.



Figura 2.11: Sensor inercial de medida MP6050 (Fuente: [NAYLAMP MECHATRONICS](#))

En la figura 2.11 podemos ver el sensor utilizado. Este se caracteriza porque combina un acelerómetro y un giroscopio, permitiendo medir tanto la aceleración como la velocidad angular en tres ejes (X, Y, Z). Este sensor es crucial para el experimento, ya que es el encargado de proporcionar información sobre la orientación y el movimiento, que se utilizan para ajustar la posición de los motores Dynamixel en tiempo real.

La aceleración, que es la variación de la velocidad por unidad de tiempo, se puede utilizar para medir cambios en la orientación. Además, los giroscopios, que miden la velocidad angular, permiten calcular la posición angular del sistema. La comunicación del módulo se realiza a través del protocolo I2C [10], lo que le permite trabajar con la mayoría de los microcontroladores, como la Arduino Nano.

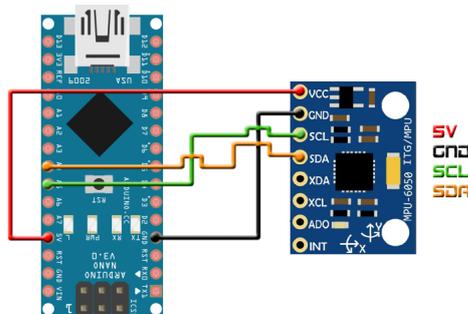


Figura 2.12: Conexión realizada entre el sensor IMU y la placa Arduino (Fuente: [Pinterest](#))

2.2. Software

Finalmente, la sección de Software describe el entorno de programación utilizado, destacando el papel fundamental de ROS (Robot Operating System) en la comunicación entre los diferentes componentes del sistema. Se explican las librerías específicas empleadas y los programas desarrollados para el control de los motores, enfatizando la suscripción a topics y el ajuste de la orientación en tiempo real basado en los datos del IMU.

Los datos y el código fuente de los programas desarrollados están disponibles en [Github](#).

2.2.1. ROS

El presente proyecto utiliza ROS (Robot Operating System) como marco fundamental para la implementación de la comunicación y el control de los motores Dynamixel. La elección de ROS permite integrar múltiples componentes de hardware y software, facilitando el desarrollo de aplicaciones robóticas complejas.

Antes de continuar, es importante explicar un término que aparecerá con frecuencia a lo largo de este documento: en el contexto de ROS, un topic es un canal de comunicación fundamental que permite la transmisión de datos entre diferentes nodos del sistema. Este mecanismo se basa en el patrón de publicador/suscriptor, donde un nodo puede publicar mensajes en un topic y otros nodos pueden suscribirse a ese mismo topic para recibir esos mensajes de manera asíncrona. Esto permite que los nodos se comuniquen de forma eficiente, sin necesidad de establecer conexiones directas entre ellos. Por ejemplo, un sensor puede publicar datos en un topic específico, mientras que otros nodos, como sistemas de monitoreo o control, pueden suscribirse a esos datos para realizar análisis o acciones en tiempo real. Este modelo promueve la modularidad y escalabilidad del sistema robótico, facilitando la integración de múltiples componentes. [11]

Con este concepto aclarado, se procede a la implementación del sistema. Para este propósito, se han empleado diversas librerías, cada una diseñada para cumplir funciones específicas que optimizan la interacción con los motores y la comunicación en el sistema ROS:

- **Dynamixel ROS Library:** Esta es una librería desarrollada por un compañero que proporciona una interfaz simplificada para el uso de los motores Dynamixel. Su principal objetivo es facilitar la programación y el control de estos motores, permitiendo que los usuarios accedan a funciones complejas con comandos simples. Esto resulta especialmente útil para aquellos que no están familiarizados con la programación a bajo nivel. Se encuentra disponible en [github](#)².
- **Dynamixel SDK:** Esta es la librería oficial proporcionada por ROBOTIS para el manejo de los motores Dynamixel. El SDK ofrece una amplia gama de funciones que permiten el control y la configuración de los motores, incluyendo la gestión de parámetros, el control de posición

²Puede acceder a github, donde se encuentra la librería, a través del siguiente [enlace](#)

y velocidad, y la comunicación con el hardware.³ No se ha usado directamente la librería, aunque es necesaria tenerla instalada para utilizar la librería Dynamixel Ros Library

- **rosserial:** Esta librería permite establecer una comunicación efectiva entre una placa Arduino y el sistema ROS. A través de rosserial, se pueden publicar los datos recibidos por los sensores conectados a la placa en diferentes topics de ROS. En este proyecto, se utilizan varios topics específicos para transmitir información sobre la orientación, como el `roll_angle_topic`, `pitch_angle_topic` y `yaw_angle_topic`. Esto permite que el sistema reaccione dinámicamente a los cambios en la orientación capturados por los sensores.

El sistema de control diseñado permite a los motores alcanzar la orientación objetivo, sin importar la posición inicial. Se han desarrollado dos programas, cada uno de los cuales implementa un nodo específico:

- **ControlTopic:** se encarga de gestionar la orientación de la muñeca esférica mediante la interacción con los motores Dynamixel y la recepción de datos de orientación desde un topic específico. Este nodo recibe la información de ángulos de inclinación (roll, yaw y pitch) que son enviados por un usuario a través del topic `user_position_input` y ajusta la posición de los motores en función de esos datos.

El nodo se suscribe al topic `user_position_input` donde espera recibir un mensaje de tipo `Int32MultiArray`. Este mensaje debe contener exactamente tres valores enteros que representan las posiciones deseadas de los tres motores. Si un motor no está en modo de control de posición, se cambia al modo correspondiente, asegurando así que los motores respondan adecuadamente a las órdenes de orientación.

Ahora se le pasan los datos de la orientación deseada, ya sea de uno, dos o los tres motores, mediante el siguiente comando:

```
rostopic pub -1 /user_position_input std_msgs/Int32MultiArray "data: [_,_,]"
```

En este comando, "data: "[_,_,]" corresponde con la posición del primer, segundo y tercer motor, respectivamente. En caso de que solo se desee modificar la posición de uno de los motores, se debe enviar el dato correspondiente a dicho motor, mientras que el resto se establecerá en -1. Por ejemplo, si se envía `[-1, -1, 50]`, solo se actualizará el valor del tercer motor, que girará hasta alcanzar 50°. Tras enviar el comando, los motores girarán hasta alcanzar la orientación objetivo desde una posición inicial.

- **ControlImu:** Tiene como objetivo principal determinar la orientación de los motores a través de los valores proporcionados por un sensor. El IMU publica constantemente los datos de su orientación en un topic, lo que permite que los motores se ajusten de manera continua para

³Para obtener más información sobre cómo utilizar el SDK y sus capacidades, se puede consultar la página oficial de [ROBOTIS](#)

alinearse con estas orientaciones. Esto garantiza que el sistema se mantenga dinámicamente adaptado a los cambios en la posición del sensor.

El nodo se suscribe a tres topics distintos:

`/yaw_angle_topic`, `/roll_angle_topic` y `/pitch_angle_topic`

Cada uno de los cuales envía información sobre el ángulo correspondiente. Cada vez que el IMU publica un nuevo valor de orientación, se invocan las funciones de callback asociadas, que actualizan las posiciones deseadas de los motores en función de la información recibida.

La frecuencia de publicación de los ajustes de posición de los motores se establece en 10 Hz mediante la instrucción `ros::Rate loop_rate(10)`. Esto significa que cada segundo, el nodo procesa los datos de los topics y publica las posiciones de los motores diez veces, lo que permite una respuesta rápida y continua a los cambios en la orientación del IMU.

2.2.2. Arduino

El programa para la placa Arduino Nano se llama `Accelerometro.ino`. Se encarga de leer los datos de un sensor MPU6050, calcular los ángulos de inclinación (roll, yaw y pitch) y publicarlos a través de tres topics: `roll_angle_topic`, `pitch_angle_topic` y `yaw_angle_topic`.

El código realiza las siguientes funciones:

- **Configuración de Librerías y Variables:** Se incluyen las librerías necesarias y se declaran las variables para el sensor MPU6050 y los nodos ROS.
- **Inicialización:** En la función `setup()`, se configura la comunicación serial, se inicializa el MPU6050 y se calibran los offsets. También se inicializan los nodos y publicadores de ROS.
- **Bucle Principal:** En la función `loop()`, se actualizan los datos del sensor, se calculan y normalizan los ángulos, y se publican los nuevos valores si la variación es mayor a 2 grados.
- **Normalización de Ángulos:** La función `normalizeAngle()` asegura que los ángulos estén dentro del rango de 0 a 360 grados.

Este enfoque permite una transmisión de datos eficiente, evitando la saturación al enviar datos únicamente cuando hay cambios significativos en los valores de los ángulos. Además, la frecuencia de ejecución del bucle es de 100 ms.

Experimentos y Resultados

Contenido

3.1. Protocolo de experimentación	22
3.2. Experimento 1. Control por Topic	24
3.2.1. Experimento 1.1 Rotación de Base Plate	24
3.2.2. Experimento 1.2 Rotación de Linkage	26
3.2.3. Experimento 1.3 Rotación de End Effector Mounting Plate	27
3.2.4. Experimento 1.4 Rotación conjunta de Linkage y End Effector Mounting Plate	28
3.2.5. Experimento 1.5 Rotación conjunta de Base Plate y Linkage	30
3.2.6. Experimento 1.6 Rotación de muñeca completa	31
3.3. Experimento 2. Control por IMU	32
3.4. Resultados y discusión	34
3.4.1. Control de la Orientación	34
3.4.2. Limitaciones Observadas	34
3.4.3. Discusión de los Resultados	34

En este capítulo se presentan las pruebas realizadas en el taller 27 de la Universidad de Málaga, con el objetivo de comprobar el correcto funcionamiento de la muñeca desarrollada y documentar los inconvenientes encontrados durante la experimentación. A lo largo de este capítulo, se describirán los protocolos de experimentación, los resultados obtenidos en cada experimento y las limitaciones observadas.

3.1. Protocolo de experimentación

Para llevar a cabo los experimentos, se parte del supuesto de que la muñeca está correctamente instalada en el brazo delta, siguiendo los pasos detallados anteriormente, como se muestra en la imagen 3.1:



Figura 3.1: Muñeca esférica montada en el brazo delta

A continuación, conectaremos los motores entre sí, formando una conexión en serie, como indica la siguiente imagen 3.2:

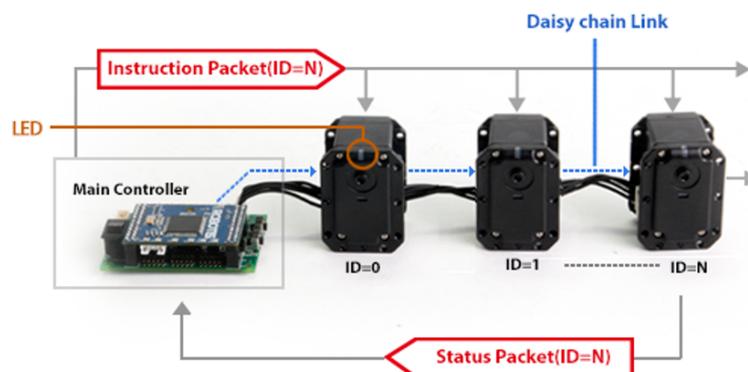


Figura 3.2: Conexión en serie de los motores Dynamixel (Fuente: [ROBOTIS](#))

Seguidamente, los conectaremos al PC, resultando en un montaje como el del esquema 3.3:

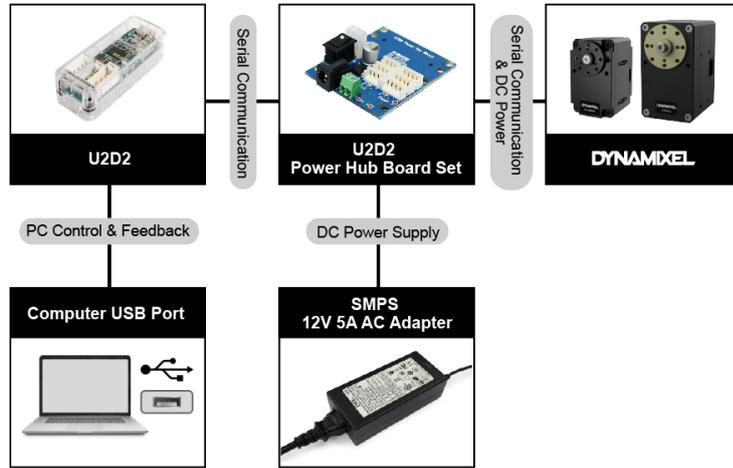


Figura 3.3: Conexión de los motores con el PC (Fuente: [ROBOTIS](#))

Una vez realizado todo el conexionado, deberemos ejecutar un par de comandos en el terminal antes de empezar la fase de experimentación:

```
ordavidro@ordavidro:~$ roscore
... logging to /home/ordavidro/.ros/log/dc3ae77a-815f-11ef-bdab-35ef62361d3e/roslaunch-ordavidro-27859.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://ordavidro:35343/
ros_comm version 1.16.0
```

SUMMARY

```
PARAMETERS
* /roscore: noetic
* /rosversion: 1.16.0
```

NODES

```
auto-starting new master
process[master]: started with pid [27873]
ROS_MASTER_URI=http://ordavidro:11311/

setting /run_id to dc3ae77a-815f-11ef-bdab-35ef62361d3e
process[rosout-1]: started with pid [27883]
started core service [/rosout]
```

Nodo Roscore

```
ordavidro@ordavidro:~$ roslaunch rosserial_python serial_node.py /dev/ttyUSB1
[INFO] [1727944222.839989]: ROS Serial Python Node
[INFO] [1727944222.857325]: Connecting to /dev/ttyUSB1 at 57600 baud
[INFO] [1727944224.973745]: Requesting topics...
[INFO] [1727944226.899678]: Note: publish buffer size is 280 bytes
[INFO] [1727944226.902175]: Setup publisher on roll_angle_topic [std_msgs/Float32]
[INFO] [1727944226.915111]: Setup publisher on yaw_angle_topic [std_msgs/Float32]
[INFO] [1727944226.929304]: Setup publisher on pitch_angle_topic [std_msgs/Float32]
```

Nodo de enlace de conexión entre Arduino y ROS

Figura 3.4: Nodos básicos para la ejecución

Para empezar, ejecutaremos el comando de 'roscore' y el nodo de enlace de conexión entre Arduino y ROS, obteniendo lo que vemos en la imagen 3.4.

3.2. Experimento 1. Control por Topic

En este primer experimento, la orientación objetivo a la que se moverán los tres motores se enviará a través de un topic. Para ello, utilizaremos el nodo "ControlTopic" creado.

```
ordavidro@ordavidro:~$ rosrn wrist_control ControlTopic
[ INFO] [1727942350.128644993]: Initialization success
[ INFO] [1727942350.137344195]: DMXL 1: Control table set for: XM430-W210
[ INFO] [1727942350.139700595]: Initialization success
[ INFO] [1727942350.146322697]: DMXL 2: Control table set for: XM430-W210
[ INFO] [1727942350.148794097]: Initialization success
[ INFO] [1727942350.155343999]: DMXL 3: Control table set for: XM430-W210
```

Figura 3.5: Ejecución del nodo "ControlTopic" donde los motores quedan en espera de recibir la orientación objetivo.

Una vez lanzado el nodo, se establece comunicación con los motores. Prueba de ello se puede ver en la imagen 3.5. El nodo debe estar en ejecución durante el experimento; por ello, abrimos otro terminal desde donde publicaremos los datos de la orientación.

```
ordavidro@ordavidro:~$ rostopic list
/rosout
/rosout_agg
/user_position_input
```

Figura 3.6: Comando que hay que utilizar para ver qué topics están activos.

Si ejecutamos el comando de la figura 3.6, podremos observar los topics que se encuentran activos. De la lista, publicaremos en el topic "*user_position_input*".

Durante el experimento, se observó un problema de diseño relacionado con la rotación del primer motor. Cuando se intenta girar la pieza "Base Plate" a ángulos fuera del rango de 210-320°, el cable que une el motor 1 al motor 2 se engancha con las varillas del brazo delta, provocando que el motor se desconecte. Esto resulta en un error, la terminación del programa y, por consiguiente, el experimento. Este problema se debe a que no se consideraron adecuadamente las varillas del brazo delta durante el diseño de la muñeca. Para evitar este inconveniente, se ha limitado el rango de rotación del primer motor entre 210 y 320° en los experimentos.

Vamos a realizar este experimento para diferentes casos enviando distintas orientaciones. Además, en todos los casos se recogerá una gráfica utilizando el comando rosbag donde se verá cómo se mantiene la posición angular del motor hasta que se manda el valor y acto seguido comienza a rotar hasta alcanzar la posición deseada.

3.2.1. Experimento 1.1 Rotación de Base Plate

Si se le pasa solo un dato de orientación al motor 1, se consigue hacer rotar la pieza "Base Plate". Este caso se puede apreciar en el *video*¹. Se han recogido un par de frames de ese video, los cuales se pueden ver en la figura 3.7, donde se muestra la orientación inicial y final de la pieza. En este caso, se le ha mandado rotar hasta 230°.

¹[Enlace al video del experimento 1.1](#)

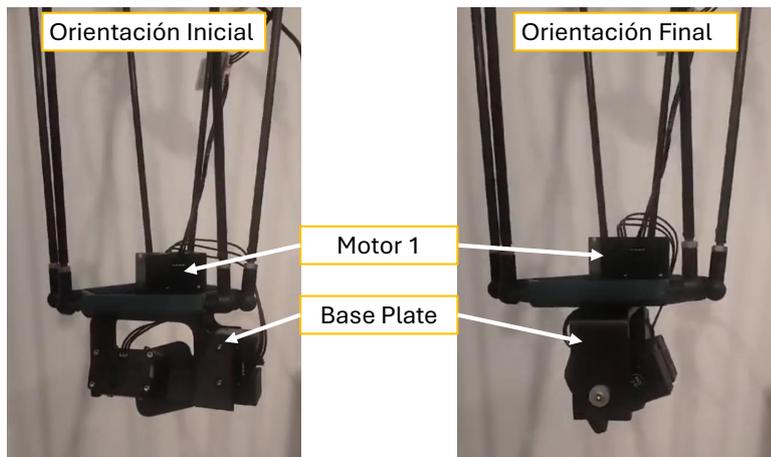


Figura 3.7: Rotación de Base Plate

Dado que en la figura 3.7 no se aprecian bien las piezas, también hemos incluido la imagen 3.8 generada en Onshape. Esta imagen muestra la muñeca en la misma orientación que en los frames tomados, permitiendo apreciar claramente las distintas piezas.

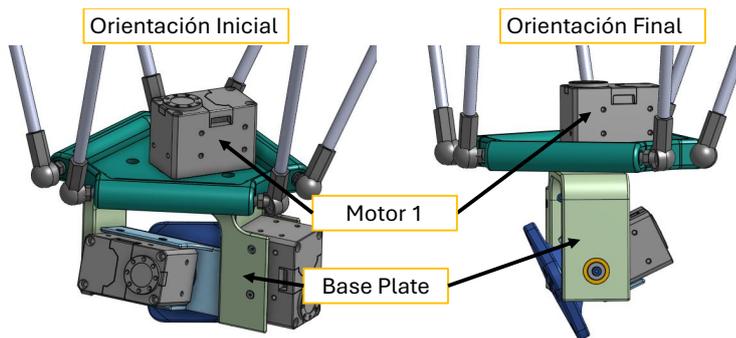


Figura 3.8: Rotación de Base Plate en onshape

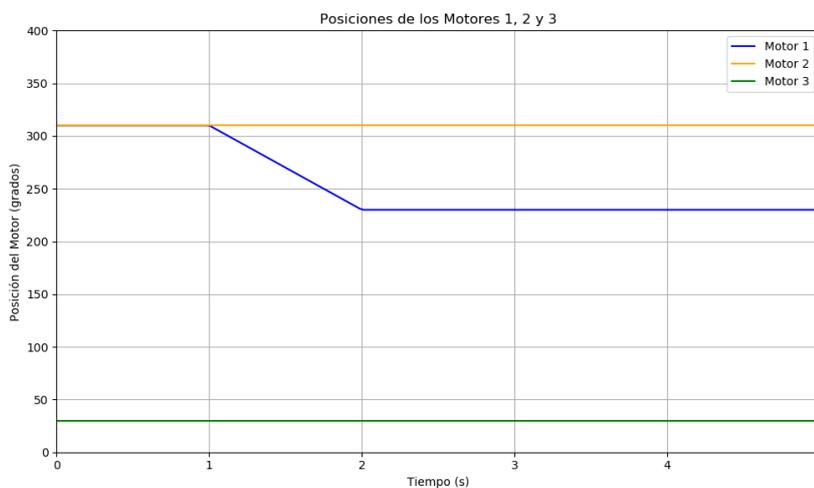


Figura 3.9: Posición angular marcada por los motores a lo largo del experimento 1.1

Podemos apreciar en la gráfica 3.12 que el motor 1 se encuentra en 310° hasta que cambia a 230° , siendo esta la orientación buscada para la muñeca.

3.2.2. Experimento 1.2 Rotación de Linkage

Si se le pasa solo un dato de orientación al motor 2, se consigue hacer rotar la pieza "Linkage". Este caso se puede apreciar en el *video*². Se han recogido un par de frames de ese video, los cuales se pueden ver en la figura 3.10, donde se muestra la orientación inicial y final de la pieza. En este caso, se le ha mandado rotar hasta 310° .

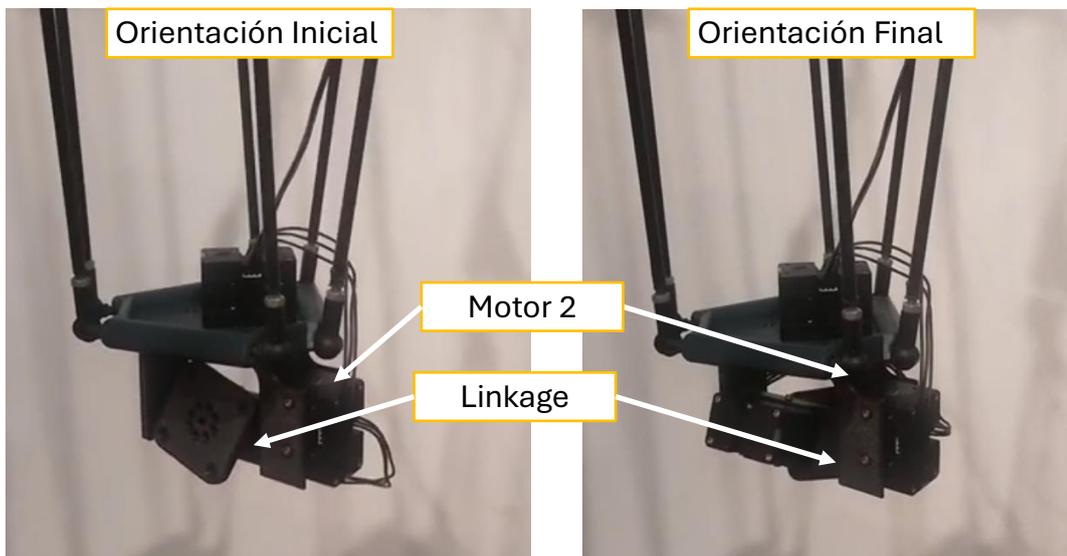


Figura 3.10: Rotación de Linkage a 310°

Dado que en la figura 3.10 no se aprecian bien las piezas, también hemos incluido la figura 3.11 generada en Onshape. Esta imagen muestra la muñeca en la misma orientación que en los frames tomados, permitiendo apreciar claramente las distintas piezas.

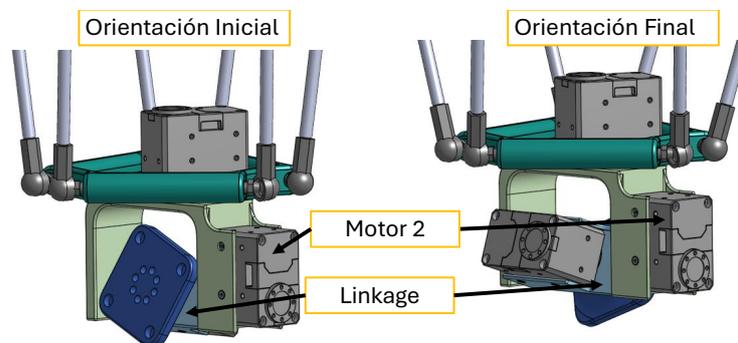


Figura 3.11: Rotación de Linkage a 310° en Onshape

²[Enlace al video del experimento 1.2](#)

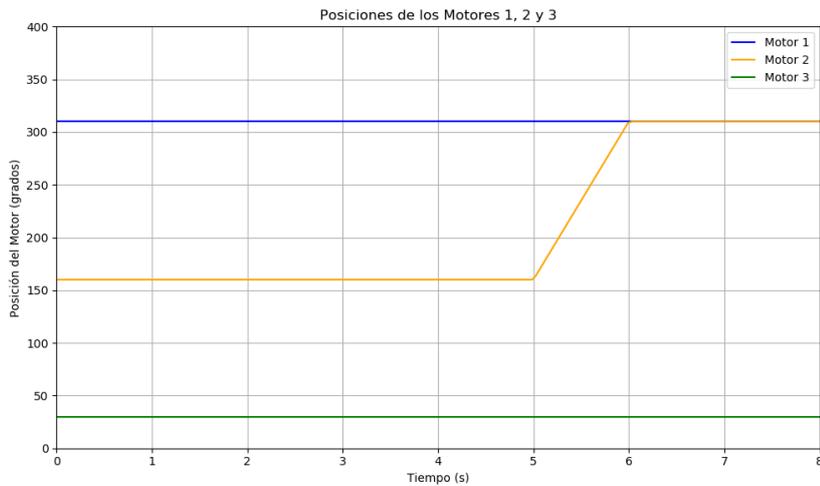


Figura 3.12: Posición angular marcada por los motores a lo largo del experimento 1.2

Podemos apreciar en la gráfica 3.12 que el motor 2 se encuentra en 160° hasta que cambia a 310°, siendo esta la orientación buscada para la muñeca.

3.2.3. Experimento 1.3 Rotación de End Effector Mounting Plate

Si se le pasa solo un dato de orientación al motor 3, se consigue hacer rotar la pieza "End Effector Mounting Plate". Este caso se puede apreciar en el *video*³. Se han recogido un par de frames de ese video, los cuales se pueden ver en la figura 3.13, donde se muestra la orientación inicial y final de la pieza. En este caso, se le ha mandado rotar hasta 300°.

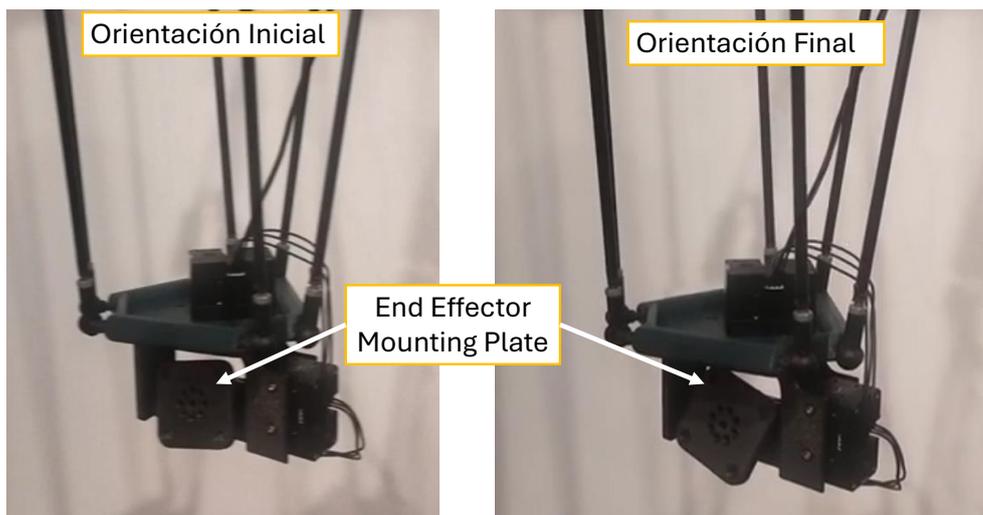


Figura 3.13: Rotación de End Effector Mounting Plate a 300°

³[Enlace al video del experimento 1.3](#)

Dado que en la figura 3.13 no se aprecian bien las piezas, también hemos incluido la figura 3.14 generada en Onshape. Esta imagen muestra la muñeca en la misma orientación que en los frames tomados, permitiendo apreciar claramente las distintas piezas

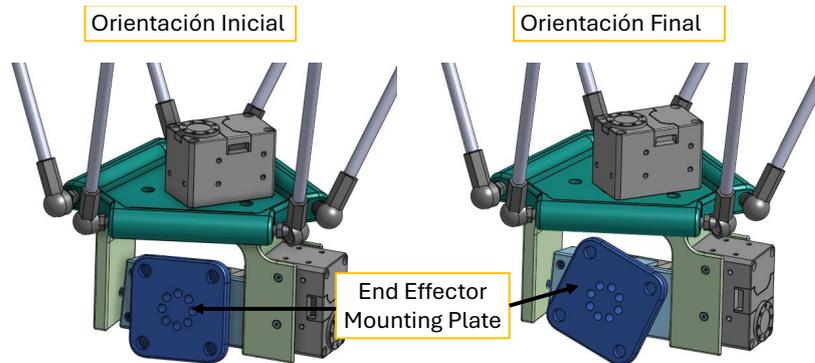


Figura 3.14: Rotación de End Effector Mounting Plate a 300° en onshape

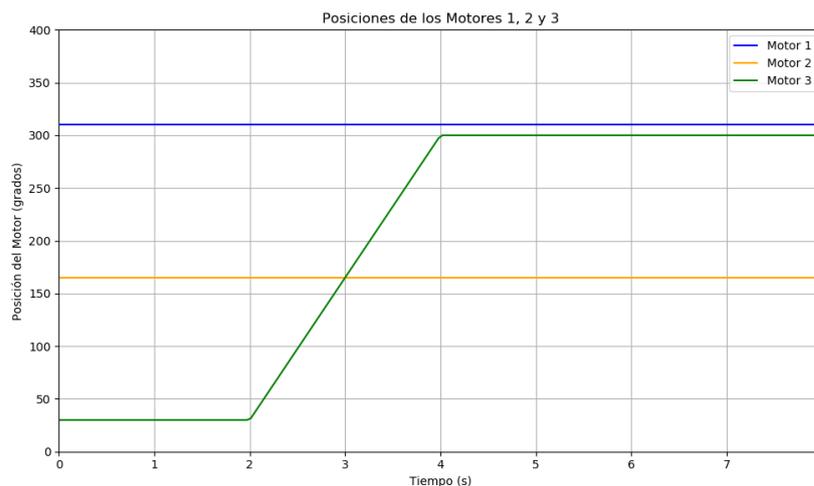


Figura 3.15: Posición angular marcada por los motores a lo largo del experimento 1.3

Podemos apreciar en la gráfica 3.15 que el motor 3 se encuentra en 30° hasta que cambia a 300°, siendo esta la orientación buscada para la muñeca.

3.2.4. Experimento 1.4 Rotación conjunta de Linkage y End Effector Mounting Plate

En el *video*⁴ se observa cómo las piezas Linkage y End Effector Mounting Plate giran conjuntamente. Este experimento demuestra la capacidad del sistema para manejar múltiples ejes de rotación de manera simultánea. La figura 3.16 ilustra las posiciones inicial y final.

⁴[Enlace al video del experimento 1.4](#)

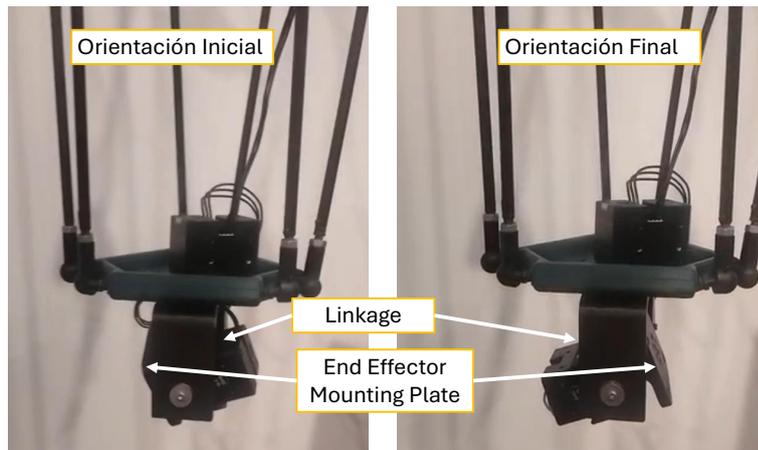


Figura 3.16: Rotación de Linkage y End Effector Mounting Plate

Dado que en la figura 3.16 no se aprecian bien las piezas, también hemos incluido la figura 3.17 generada en Onshape. Esta imagen muestra la muñeca en la misma orientación que en los frames tomados, permitiendo apreciar claramente las distintas piezas

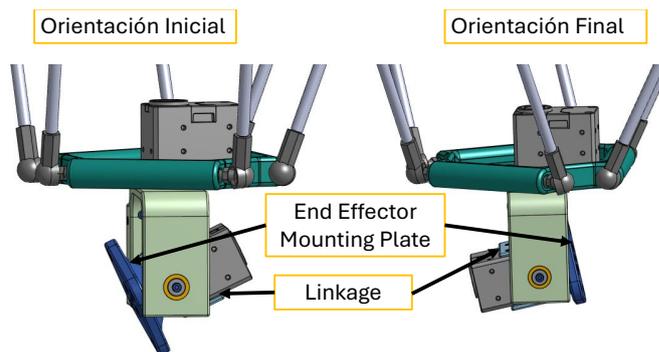


Figura 3.17: Rotación de Linkage y End Effector Mounting Plate en Onshape

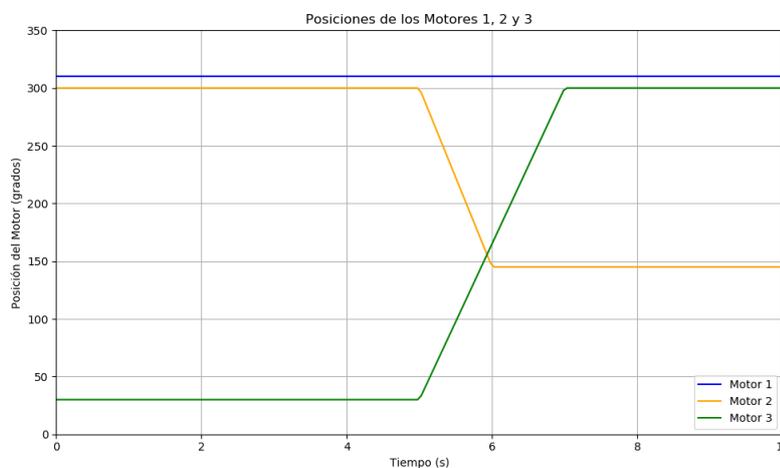


Figura 3.18: Posición angular marcada por los motores a lo largo del experimento 1.4

Podemos apreciar en la gráfica 3.18 que el motor 3 se mantiene a 30° hasta que cambia a 300° . En ese mismo instante, el motor 2 cambia de 300° a 145° , que es la orientación buscada para la muñeca..

3.2.5. Experimento 1.5 Rotación conjunta de Base Plate y Linkage

El *video*⁵ muestra cómo las piezas Base Plate y Linkage giran de manera conjunta. La figura 3.19 muestra las posiciones iniciales y finales.

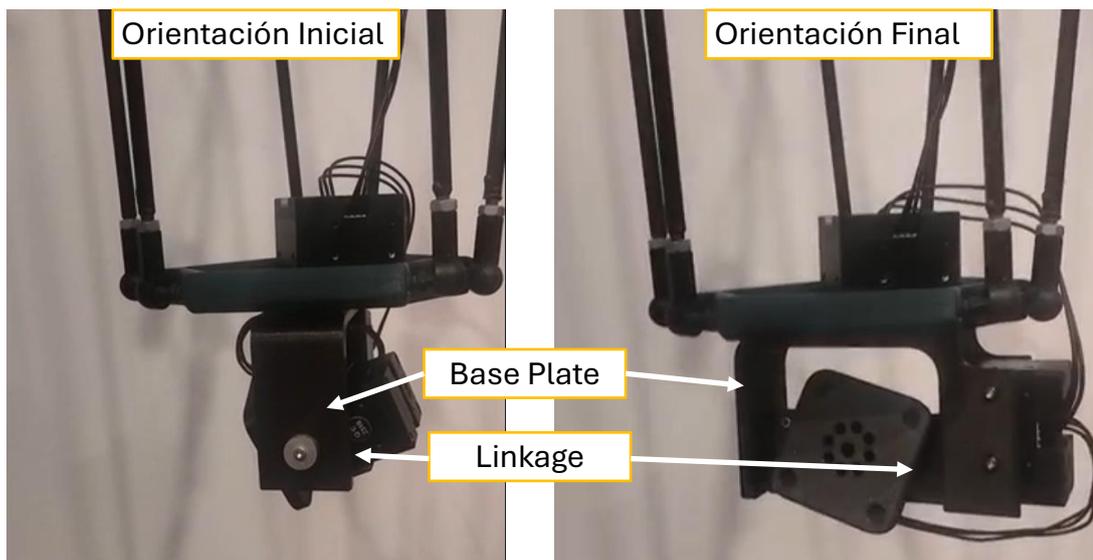


Figura 3.19: Rotación de Base Plate y Linkage

Dado que en la figura 3.19 no se aprecian bien las piezas, también hemos incluido la figura 3.20 generada en Onshape. Esta imagen muestra la muñeca en la misma orientación que en los frames tomados, permitiendo apreciar claramente las distintas piezas

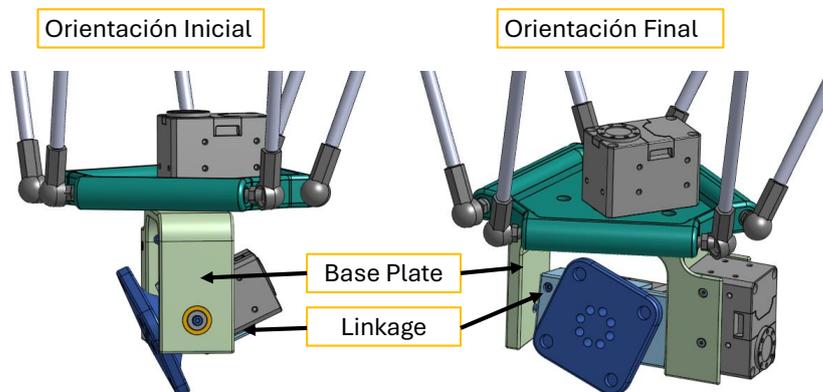


Figura 3.20: Rotación de Base Plate y Linkage en Onshape

⁵[Enlace al video del experimento 1.5](#)

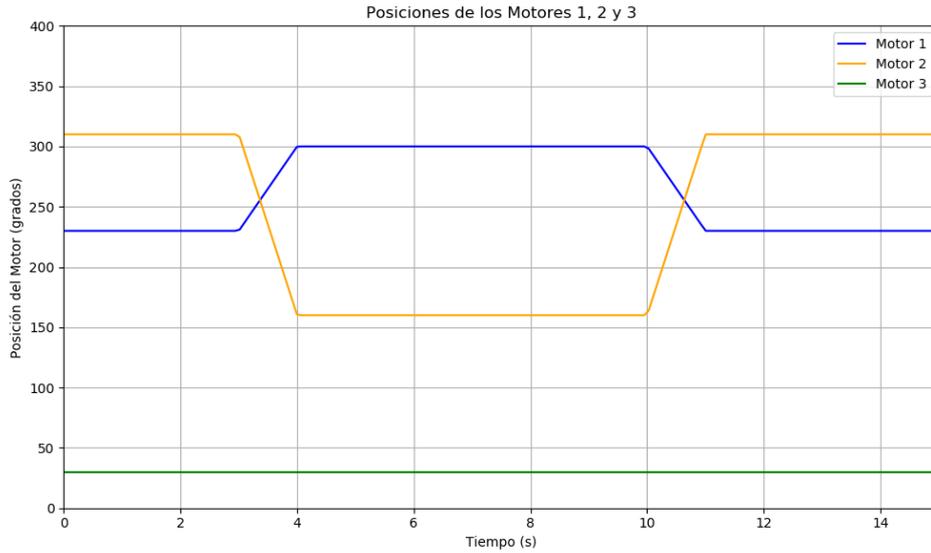


Figura 3.21: Posición angular marcada por los motores a lo largo del experimento 1.5

Se puede apreciar en la gráfica 3.21 que el motor 1 originalmente tenía una orientación de 230° y en el momento de lanzamiento del nodo cambia hasta 300° mientras que el motor 2 pasa de 310° hasta 160° . Ambos, después de 4 segundos vuelven a sus posiciones originales tras recibir la orientación por el topic..

3.2.6. Experimento 1.6 Rotación de muñeca completa

El video ⁶ muestra como la muñeca gira de manera conjunta. La figura 3.22 muestra las posiciones iniciales y finales.

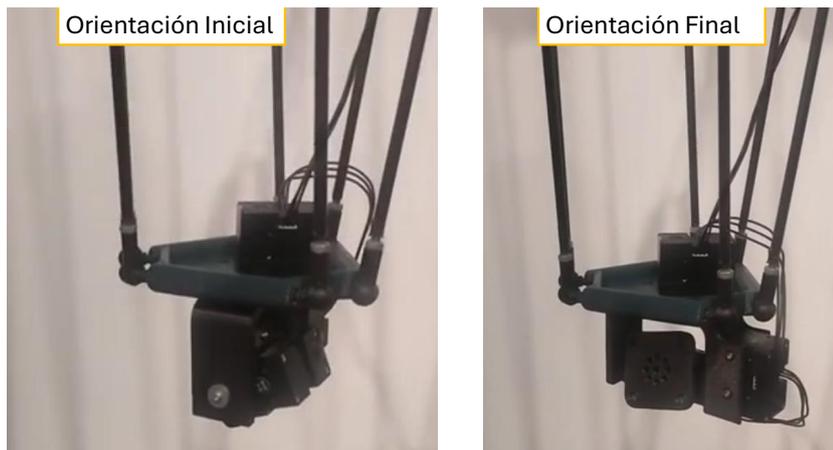


Figura 3.22: Rotación de la muñeca esférica de manera conjunta

Dado que en la figura 3.22 no se aprecian bien las piezas, también hemos incluido la figura 3.23 generada en Onshape. Esta imagen muestra la muñeca en la misma orientación que en los frames

⁶[Enlace al video del experimento 1.6](#)

tomados, permitiendo apreciar claramente las distintas piezas

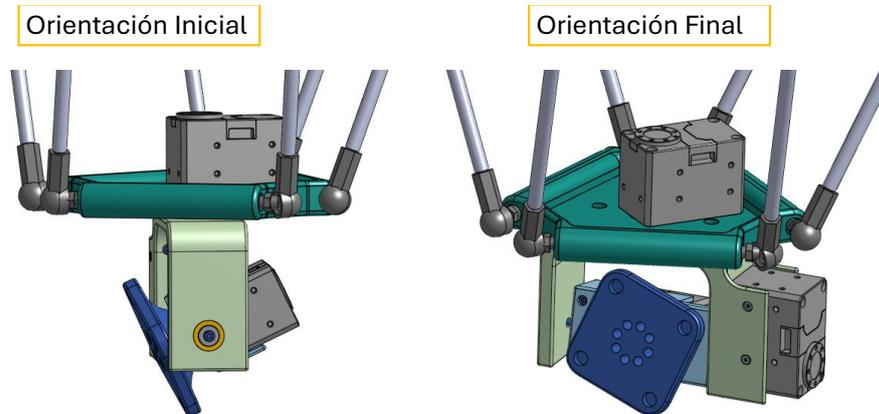


Figura 3.23: Rotación de la muñeca esférica de forma simultánea en Onshape

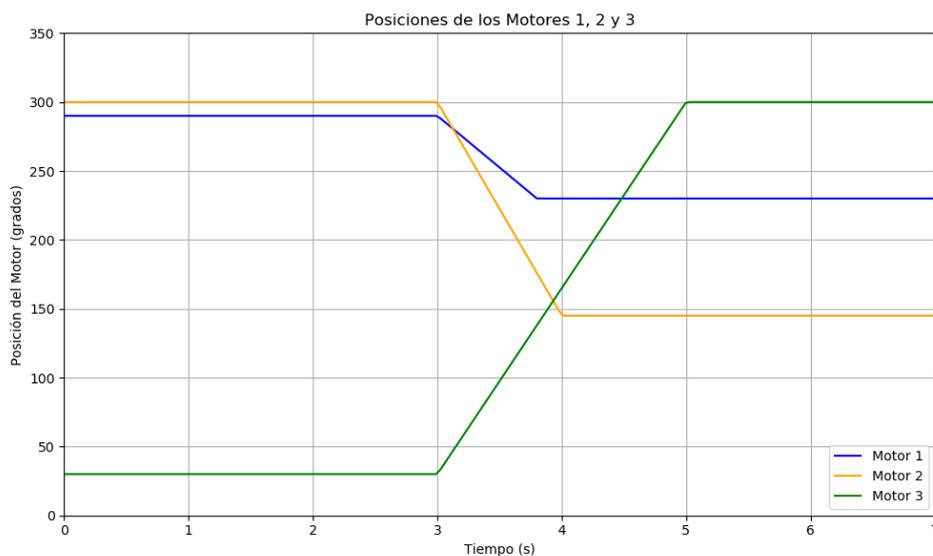


Figura 3.24: Posición angular marcada por los motores a lo largo del experimento 1.6

Se puede apreciar en la gráfica 3.24 que el motor 3 al tener más recorrido tarda 1 segundo más en llegar a la posición objetivo que el resto de los motores.

3.3. Experimento 2. Control por IMU

En este segundo experimento, los datos de orientación no se van a mandar manualmente sino que se van a enviar los obtenidos mediante un sensor MP6050. El sensor envía los datos a través de 3 topics específicos para la orientación cómo se ve en la figura 3.25

```

ordavidro@ordavidro:~/wrist_ws/src/wrist_control/Ejecutables$ ./subwithinterval.sh /roll_angle_topic /pitch_angle_topic /yaw_angle_t
opic
Topic: /roll_angle_topic
data: 154.94204711914062
----
Topic: /pitch_angle_topic
data: 136.26268005371094
----
Topic: /yaw_angle_topic
data: 94.19837951660156
----
Topic: /roll_angle_topic
data: 225.51438903808594
----
Topic: /pitch_angle_topic
data: 52.27104568481445
----
Topic: /yaw_angle_topic
data: 102.55193328857422
----
    
```

Figura 3.25: Datos tomados por el sensor publicados en sus respectivos topics.

El nodo desarrollado, llamado "ControlImu", simula los datos que enviaría el dron en función de su orientación. Esto permite que el efector final se oriente en sincronía con el sensor. Los motores tienen un ínfimo retraso desde que el sensor adopta una orientación hasta que los motores la alcanzan que varía en función de lo brusco que sea el cambio. La frecuencia con la que publica datos el sensor es de 10 Hz que es mayor a la de lectura del motor. Podemos ver el segundo experimento en el [video](#)⁷.

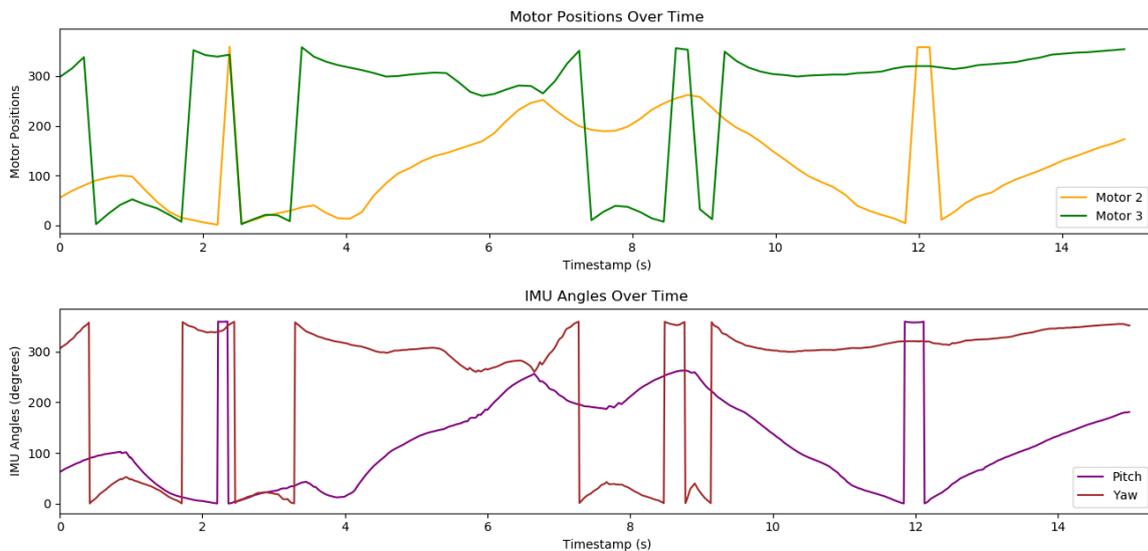


Figura 3.26: Orientación de los motores vs orientación del sensor inercial.

Durante este se han sacado y graficado los datos cómo se vé en la figura 3.26. En esta se puede observar la linealidad que lleva la posición del ángulo proporcionado por el sensor inercial con la orientación que adopta el motor una vez recibido el dato. La gráfica de arriba muestra la orientación de los motores mientras que la de abajo muestra los datos aportados por la imu. Se aprecia que Pitch y Yaw están vinculados con el motor 2 y motor 3 respectivamente mientras que el motor 1, dada la limitación del cable encontrada en el experimento 1 no se ha introducido.

⁷[Enlace al video del experimento 1.6](#)

3.4. Resultados y discusión

Los resultados obtenidos durante los experimentos demuestran que el sistema es capaz de controlar la rotación de las piezas de la muñeca esférica de manera efectiva. Se llevaron a cabo varias pruebas para evaluar el rendimiento del sistema en términos de precisión de la orientación y estabilidad del efector final. A continuación, se presentan los hallazgos más relevantes:

3.4.1. Control de la Orientación

Las pruebas iniciales se centraron en verificar la capacidad del sistema para realizar rotaciones precisas utilizando los ángulos Roll-Pitch-Roll (RPR). Los motores Dynamixel XM430-W210, programados mediante ROS, lograron ajustar la orientación del efector final con un error mínimo en los ángulos deseados. La gráfica 3.26 muestra una comparación entre los ángulos deseados y los ángulos alcanzados por el sistema durante una serie de movimientos programados.

3.4.2. Limitaciones Observadas

A pesar del éxito en el control y la estabilidad, se presentaron algunas limitaciones que afectan el rendimiento y la eficacia del sistema:

- **Conectividad de los motores:** En ocasiones, se produjeron desconexiones momentáneas que interrumpieron la ejecución de los movimientos programados. Esto se atribuye a la longitud insuficiente de uno de los cables de conexión entre motores, lo que limita el rango de movimiento del motor 1. Para evitar este problema, se acotaron los valores de giro del motor, restringiendo su rotación. Esta limitación debe ser abordada mediante la adquisición de cables más largos o la reubicación de los motores para asegurar una conexión estable y continua.
- **Calibración de Sensores:** Durante las pruebas con el sensor inercial (IMU), se observó que pequeñas variaciones en la calibración inicial afectaban significativamente la precisión de la orientación del efector final. Es crucial realizar una calibración precisa y constante del sensor IMU para minimizar estos errores.
- **Latencia en la Comunicación:** En algunos casos, se detectó una latencia en la comunicación entre los nodos de ROS, lo que resultó en retardos en la ejecución de los comandos de movimiento. Esta latencia puede ser mitigada optimizando la configuración de la red y el hardware utilizado para ejecutar ROS.

3.4.3. Discusión de los Resultados

Los resultados de este proyecto muestran que la integración de una muñeca esférica controlada mediante motores Dynamixel y programada con ROS es viable y efectiva para aplicaciones de

robótica aérea. La precisión en la orientación y la estabilidad del efector final son puntos fuertes del sistema, lo que lo hace adecuado para tareas que requieren alta precisión.

No obstante, las limitaciones observadas indican áreas de mejora necesarias para optimizar el rendimiento del sistema. La solución a los problemas de conectividad y latencia en la comunicación, así como la mejora de la calibración de los sensores, son pasos fundamentales para robustecer el sistema. Asimismo, considerar mejoras en la capacidad de carga del manipulador ampliará su rango de aplicaciones potenciales.

En conclusión, este proyecto proporciona una base sólida para el desarrollo de manipuladores aéreos precisos y estables, con aplicaciones potenciales en diversas áreas de la robótica industrial y de servicio. Las mejoras sugeridas en este documento guiarán futuras iteraciones y optimizaciones del sistema.

Conclusión

Finalmente, tras la revisión y desarrollo del proyecto, se presenta una síntesis de los hallazgos y logros alcanzados a lo largo de esta investigación. En este Trabajo de Fin de Grado (TFG), se ha abordado el desafío de diseñar y controlar una muñeca esférica que añade 3 Grados de Libertad (GDL) a brazos utilizados en manipuladores aéreos.

El proyecto ha seguido un plan de trabajo estructurado que ha incluido diversas etapas, como la investigación bibliográfica, el modelado y la impresión de las piezas, la integración y programación de los motores, así como la implementación y experimentación del sistema completo. A través de este enfoque, se ha logrado desarrollar un sistema que amplía los grados de libertad del brazo, permitiendo un control más efectivo de la orientación del efector final.

Entre los logros destacados del proyecto, se encuentra el diseño exitoso de una muñeca esférica integrada con tres motores, fabricada mediante técnicas de impresión 3D. La integración de estos actuadores en el sistema se llevó a cabo utilizando el entorno Robot Operating System (ROS), lo que permitió implementar un controlador que maneja eficazmente la orientación.

Los experimentos realizados demuestran la eficacia del sistema desarrollado, mostrando un control satisfactorio de la orientación del efector final. Sin embargo, también se han identificado áreas que requieren investigación y desarrollo adicionales:

1. **Optimización del Diseño Mecánico:** Explorar el uso de materiales más ligeros y resistentes para incrementar la eficiencia del sistema, o miniaturizar algunos componentes para reducir el tamaño y el peso, lo que facilitaría su uso en drones de tamaño menor al usado.
2. **Control Acoplado de Posición y Orientación:** Implementar un control integrado que maneje de forma conjunta la posición y la orientación entre el brazo delta y la muñeca esférica.

- 3. Rediseño de la Muñeca:** Mejorar el diseño de la muñeca para evitar que los cables se enreden, lo que podría comprometer el rendimiento del sistema.

En conclusión, el proyecto ha logrado establecer las bases para futuras investigaciones en el ámbito de los manipuladores aéreos, abriendo nuevas oportunidades para la optimización y mejora del sistema desarrollado. A medida que avanza la tecnología, se prevé que estos desarrollos contribuyan a un control más preciso y eficiente de los manipuladores, lo que a su vez tendrá un impacto positivo en diversas aplicaciones industriales y de investigación.

Bibliografía

- [1] SICMA, “Robots industriales: tecnología y aplicaciones,” 2023, accessed: 2024-05-21. [Online]. Available: <https://www.sicma21.com/robots-industriales-tecnologia-y-aplicaciones/>
- [2] AER Automation, “Robótica aérea de manipulación para inspección y mantenimiento,” 2023, accessed: 2024-05-28. [Online]. Available: https://www.aer-automation.com/automation_review/robotica-aerea-de-manipulacion-para-inspeccion-y-mantenimiento/
- [3] CIT UPC, “Aeroarms project,” 2023, accessed: 2024-05-27. [Online]. Available: https://cit.upc.edu/es/portfolio-item/aeroarms_iri/
- [4] A. Ollero Baturone, *Robótica: manipuladores y robots móviles*. Marcombo S.A, 2001, accessed: 2024-05-10. [Online]. Available: <https://el.uy/doc/robotica/Ollero%20Baturone,%20An%C3%ADbal.%202001.%20Rob%C3%B3tica,%20manipuladores%20y%20robots%20m%C3%B3viles.pdf>
- [5] M. Canseco, “Diseño de un robot paralelo,” *ITECKNE*, vol. 6, 12 2009. [Online]. Available: https://www.researchgate.net/publication/280797475_Disen%C3%B3_de_un_robot_paralelo, note={Accessed:2024-05-28}
- [6] E. Dimitrakakis, G. Dwyer, L. Lindenroth, P. Giataganas, N. L. Dorward, H. J. Marcus, and D. Stoyanov, “A spherical joint robotic end-effector for the expanded endoscopic endonasal approach,” *Journal of Medical Robotics Research*, vol. 05, no. 03n04, p. 2150002, 2020. [Online]. Available: <https://doi.org/10.1142/S2424905X21500021>
- [7] K. Abe, K. Tadakuma, and R. Tadakuma, “Abenics: Active ball joint mechanism with three-dof based on spherical gear meshings,” *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1806–1825, 2021.
- [8] YouTube, “Muñeca esférica de un robot manipulador,” 2023, accessed: 2024-06-01. [Online]. Available: <https://www.youtube.com/watch?v=liuwgcACc1c>
- [9] formlabs, “Guía de materiales de impresión 3d: Tipos, aplicaciones y propiedades,” accessed: 2024-04-20. [Online]. Available: <https://formlabs.com/es/blog/materiales-impresion-3d/>

- [10] N. MECHATRONICS, “Tutorial mpu6050, acelerómetro y giroscopio,” accessed: 2024-04-20. [Online]. Available: https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html

- [11] J. Joseph, L. & Cacace, *MasteringROSforRoboticsProgramming : Design,build,andsimulatecomplexrobotsusingtheRobotOperatingSystem*. Packt Publishing Ltd., 2018.

Apendice A

En este apéndice se presentan los planos detallados del proyecto. Los planos incluyen todas las vistas necesarias para la construcción y ensamblaje del sistema diseñado.

Índice de Planos

Plano 1: Vista General de la muñeca ensamblada

Plano 2: Vista Explosionada de la muñeca

Plano 3: Vista en detalle de la Pieza 1

Plano 4: Vista en detalle de la Pieza 2

Plano 5: Vista en detalle de la Pieza 3

6

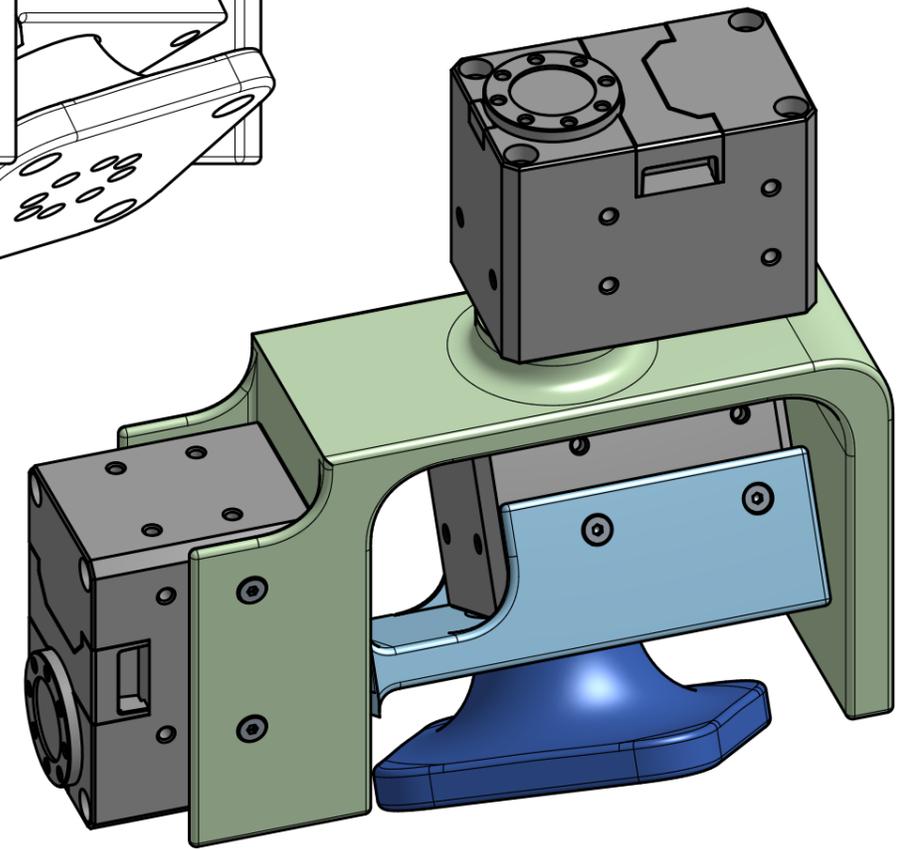
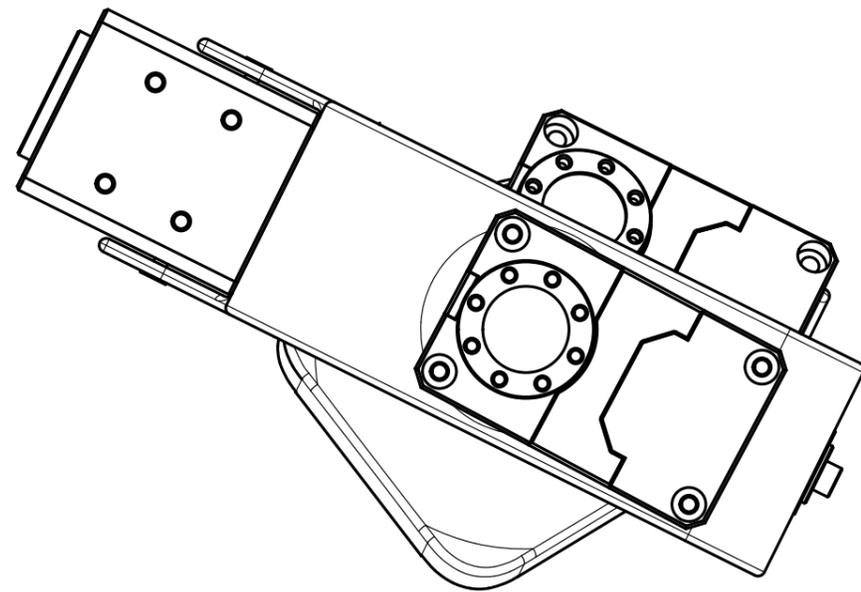
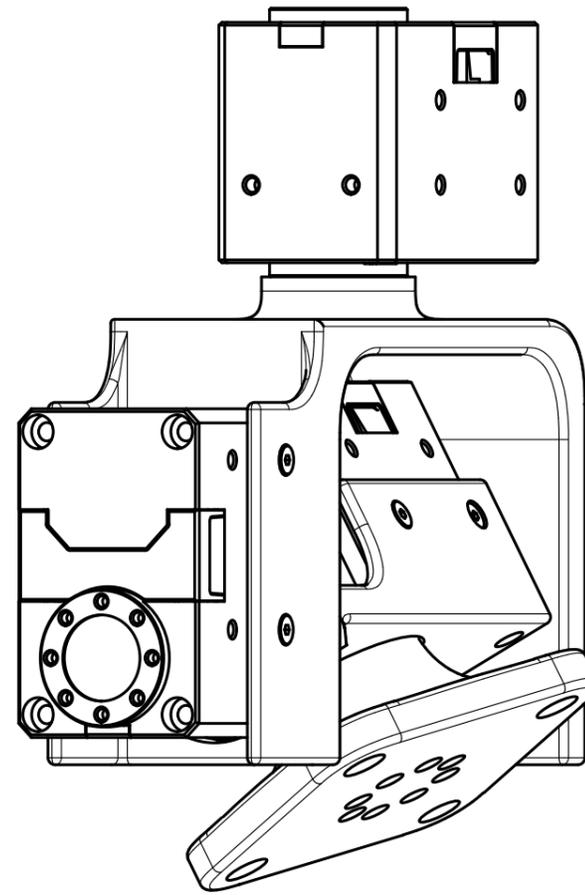
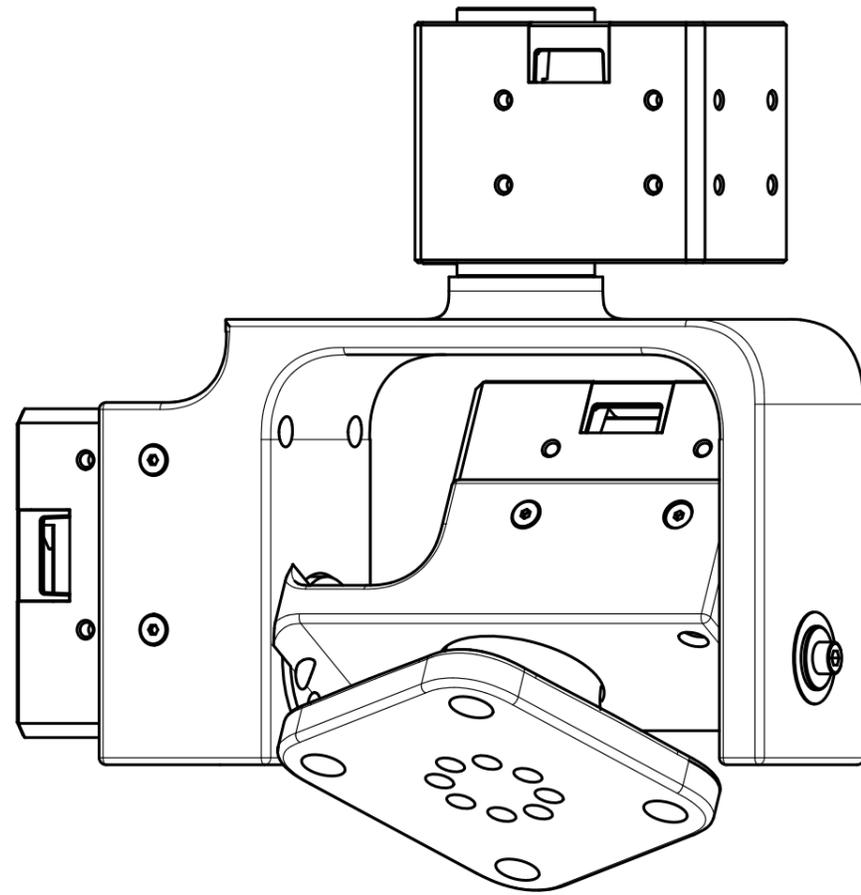
5

4

3

2

1



	NOMBRE	FIRMA	FECHA	TÍTULO:			
DIBUJADO	David Rodríguez		2024-05-27	DISEÑO Y CONTROL DE UNA MUÑECA ROBÓTICA DE 3 GRADOS DE LIBERTAD PARA UN MANIPULADOR DELTA LIGERO			
REVISADO	Francisco Pastor		2024-05-29				
APROBADO	Juan Manuel Gandarias		2024-05-30				
FIRST ANGLE PROJECTION		NOTAS		TAMAÑO:	DWG NO.	PLANO Nº:	
				A3	MUÑECA ESFÉRICA	1	
ESCALA:	1:2	PESO:	360 g	MATERIAL:	PLA	HOJA:	1 of 1

6

5

4

3

2

1

D

D

C

C

B

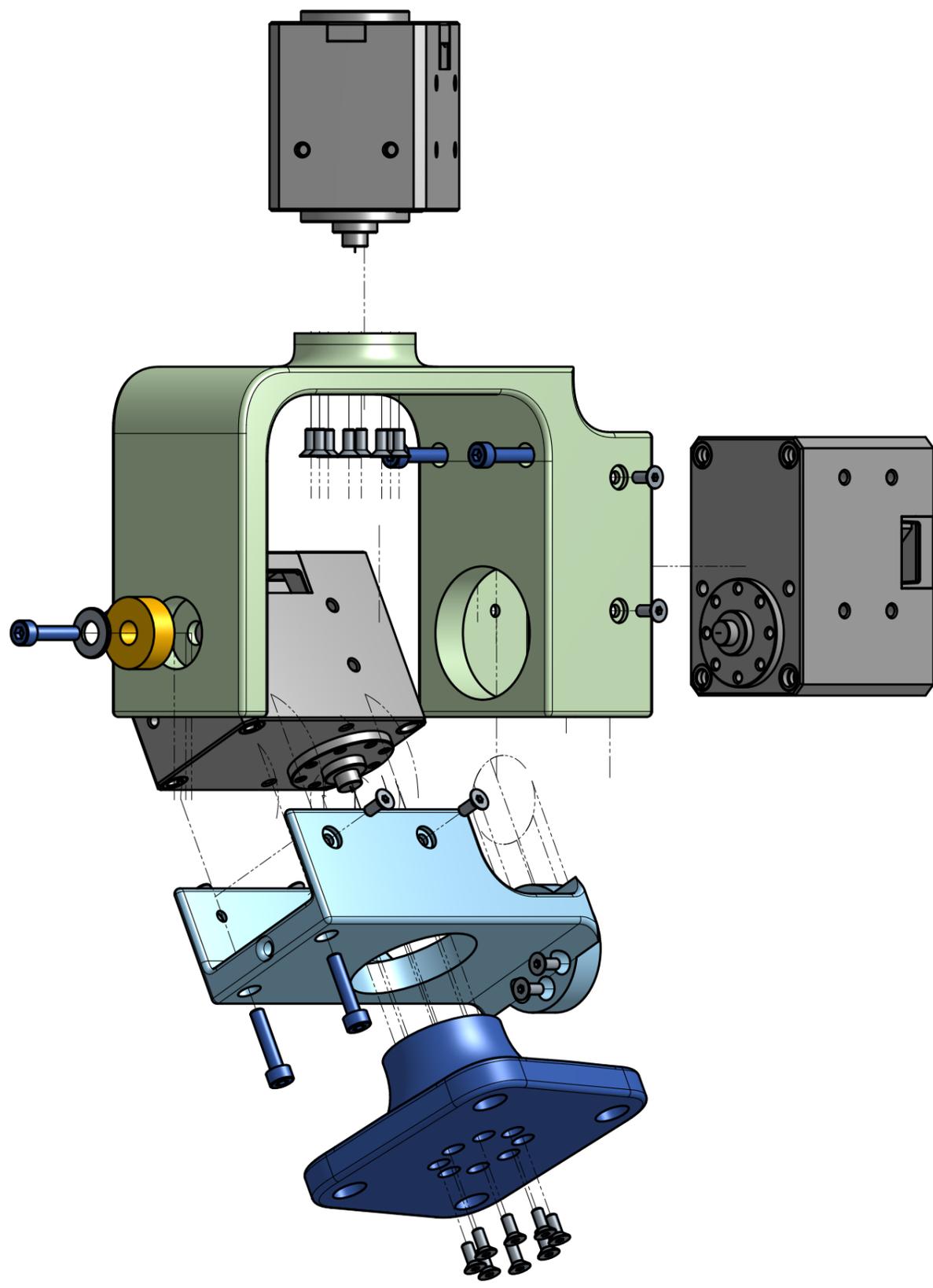
B

A

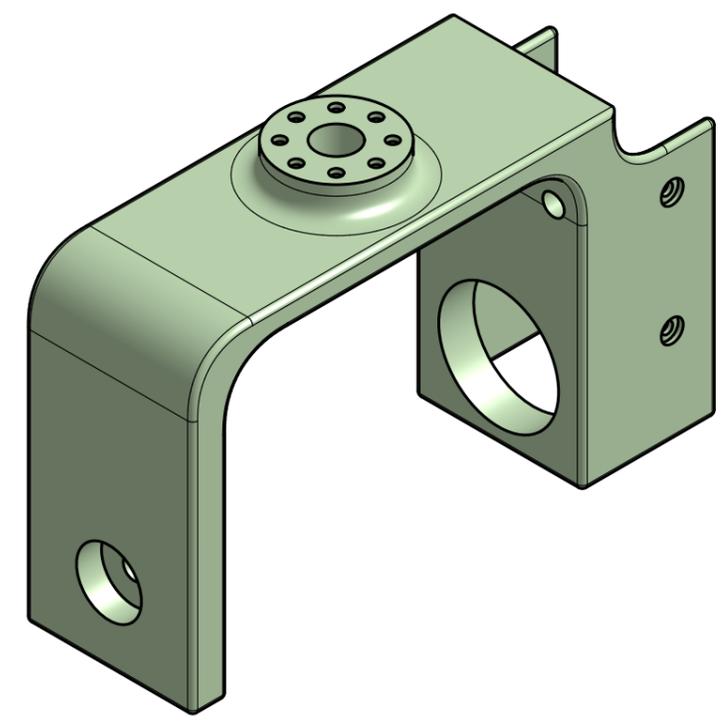
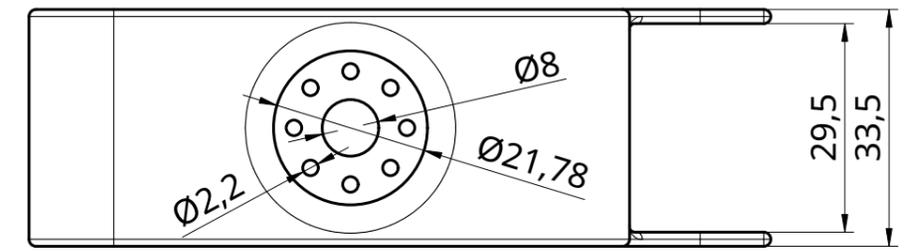
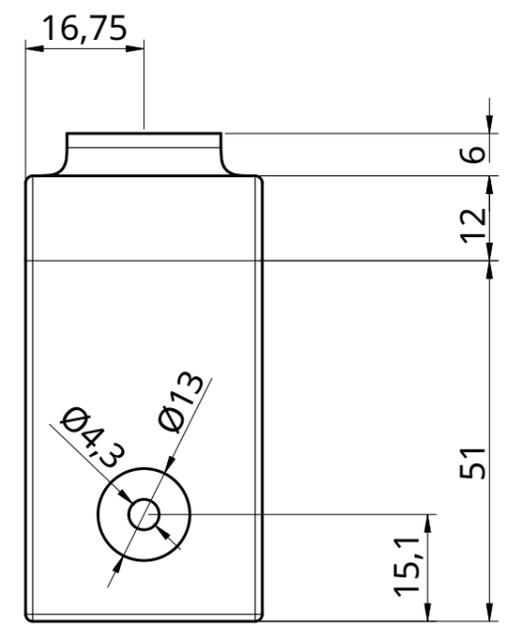
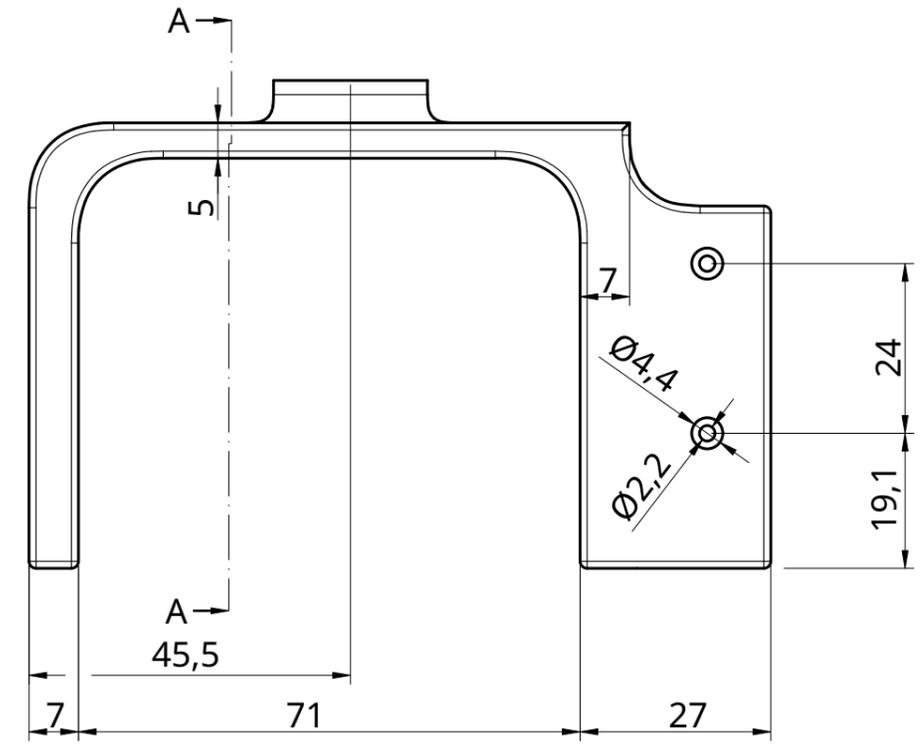
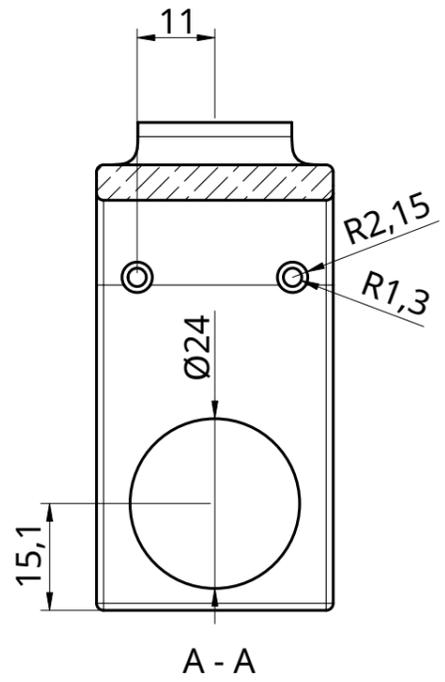
A

LISTADO DE PIEZAS

Elemento nº	Cantidad	Descripción
1	3 uds	Motor Dynamixel XM430-W210T
2	4 uds	Tornillo M2 5X12
3	32 uds	Tornillos M2 0.4X5
4	1 ud	Tornillo M2 5X18
5	1 ud	Arandela M4
6	1 ud	Rodamiento 8x22x7
7	1 ud	Pieza 1
8	1 ud	Pieza 2
9	1 ud	Pieza 3



	NOMBRE	FIRMA	FECHA	TÍTULO:		
DIBUJADO	David Rodríguez		2024-05-27	DISEÑO Y CONTROL DE UNA MUÑECA ROBÓTICA DE 3 GRADOS DE LIBERTAD PARA UN MANIPULADOR DELTA LIGERO		
REVISADO	Francisco Pastor		2024-05-29			
APROBADO	Juan Manuel Gandarias		2024-05-30			
FIRST ANGLE PROJECTION			NOTAS			
			TAMAÑO: A3		DWG NO. VISTA EXPLOSIONADA	PLANO Nº: 2
ESCALA: 1:1		PESO: 360 g	MATERIAL: PLA	HOJA: 1 of 1		

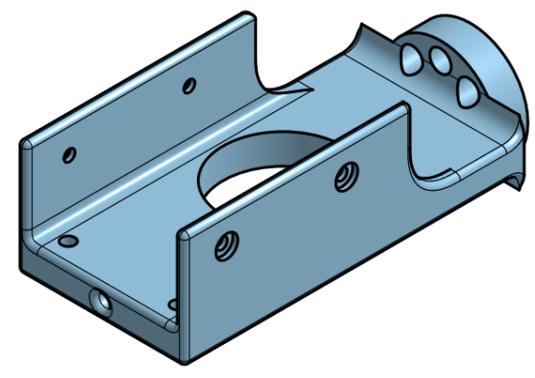
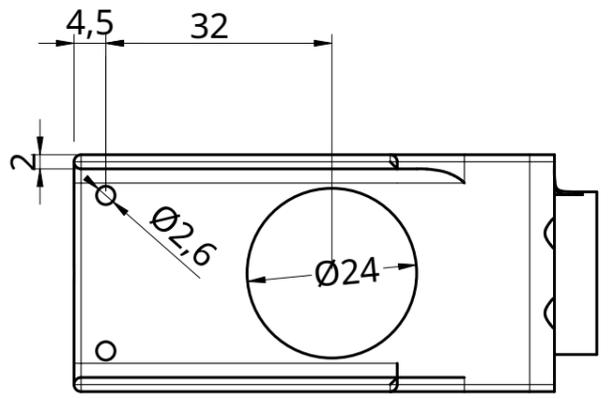
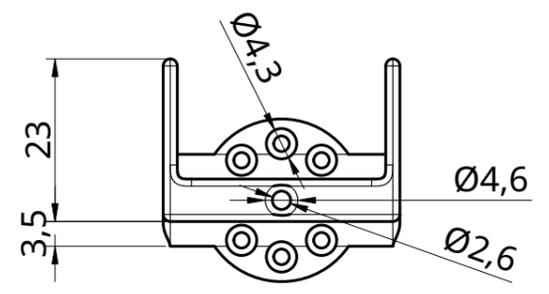
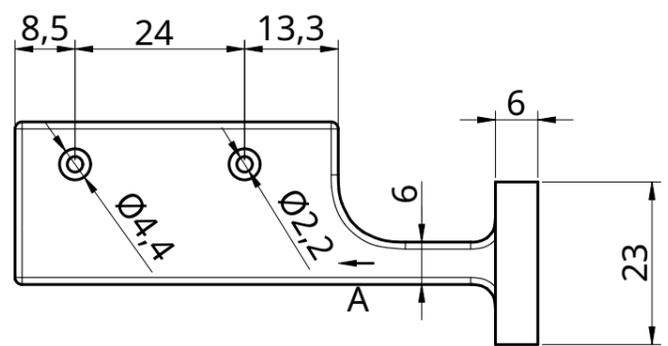
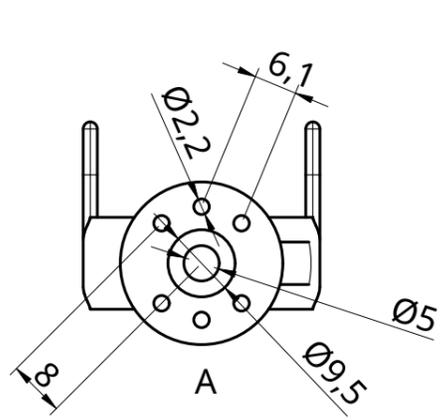


			NOMBRE		FIRMA		FECHA		TÍTULO:		
DIBUJADO			David Rodríguez				2024-05-27		DISEÑO Y CONTROL DE UNA MUÑECA ROBÓTICA DE 3 GRADOS DE LIBERTAD PARA UN MANIPULADOR DELTA LIGERO		
REVISADO			Francisco Pastor				2024-05-29				
APROBADO			Juan Manuel Gandarias				2024-05-30				
FIRST ANGLE PROJECTION			NOTAS								
			TAMAÑO: A3			DWG NO. PIEZA 1			PLANO Nº: 3		
ESCALA: 1:1			PESO: 30 g		MATERIAL: PLA		HOJA: 1 of 1				

6 5 4 3 2 1

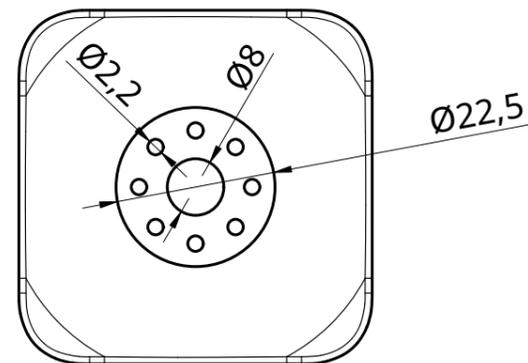
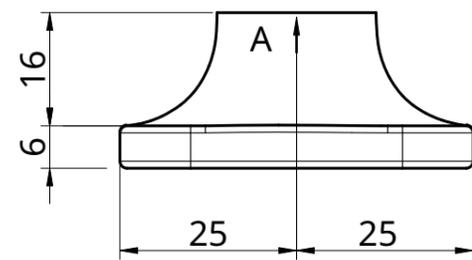
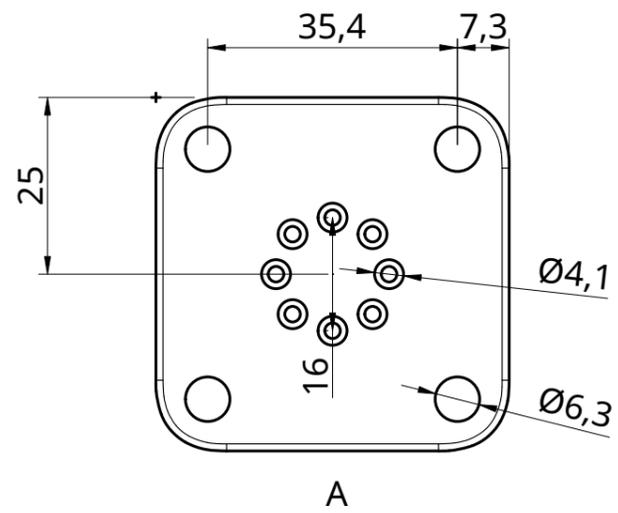
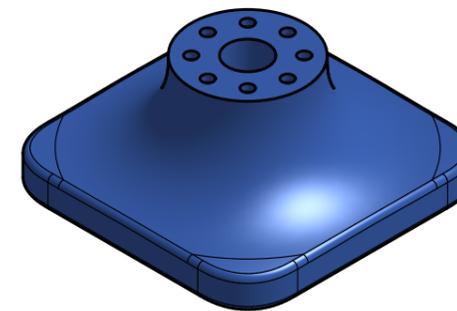
D
C
B
A

D
C
B
A



	NOMBRE	FIRMA	FECHA	TÍTULO:		
DIBUJADO	David Rodríguez		2024-05-27	DISEÑO Y CONTROL DE UNA MUÑECA ROBÓTICA DE 3 GRADOS DE LIBERTAD PARA UN MANIPULADOR DELTA LIGERO		
REVISADO	Francisco Pastor		2024-05-29			
APROBADO	Juan Manuel Gandarias		2024-05-30			
FIRST ANGLE PROJECTION		NOTAS		TAMAÑO:	DWG NO.	PLANO Nº:
				A3	PIEZA 2	4
ESCALA: 1:1		PESO: 13 g	MATERIAL: PLA	HOJA: 1 of 1		

6 5 4 3 2 1



	NOMBRE	FIRMA	FECHA	TÍTULO:				
DIBUJADO	David Rodríguez		2024-05-27	DISEÑO Y CONTROL DE UNA MUÑECA ROBÓTICA DE 3 GRADOS DE LIBERTAD PARA UN MANIPULADOR DELTA LIGERO				
REVISADO	Francisco Pastor		2024-05-29					
APROBADO	Juan Manuel Gandarias		2024-05-30					
FIRST ANGLE PROJECTION			NOTAS		TAMAÑO:	DWG NO.	PLANO Nº:	
					A3	PIEZA 3		5
					ESCALA: 1:1	PESO: 10 g	MATERIAL: PLA	HOJA: 1 of 1

