

# **Memoria del proyecto**

para el

## Trabajo de Fin de Grado sobre el manipulador *pArm*

Javier Alonso Silva  
Mihai Octavian Stanescu  
José Alejandro Moya Blanco

*Universidad Politécnica de Madrid*  
*Ingeniería de Computadores*  
*Trabajo de Fin de Grado*  
*Tutor: Norberto Cañas de Paz*

Madrid, 28 de junio de 2020

# Índice general

<b>Índice de figuras</b>	<b>IV</b>
<b>Índice de cuadros</b>	<b>VI</b>
<b>Historial de versiones</b>	<b>VII</b>
<b>Definiciones, siglas, y abreviatura</b>	<b>X</b>
<b>1. Motivación y objetivos</b>	<b>1</b>
1.1. [REVISADO] Estado del arte . . . . .	1
1.1.1. Desarrollo de la robótica a lo largo de la historia . . . . .	1
1.1.2. Los brazos robóticos . . . . .	4
1.1.3. La actualidad . . . . .	5
1.2. Motivaciones y objetivos del desarrollo del proyecto . . . . .	10
1.3. Metodología . . . . .	11
<b>2. [REVISADO] Explicación de la estructura del proyecto</b>	<b>13</b>
2.1. Matemáticas . . . . .	14
2.2. <i>Hardware</i> . . . . .	16
2.3. <i>Software</i> . . . . .	16
<b>3. Diagramas, requisitos y diseño</b>	<b>18</b>
3.1. Requisitos . . . . .	18
3.2. Diagramas y diseño . . . . .	18
<b>4. Las matemáticas del proyecto</b>	<b>22</b>
4.1. Cinemática directa . . . . .	22

---

4.2. Cinemática inversa . . . . .	22
4.3. Funciones jacobianas . . . . .	22
4.4. Implementación final realizada . . . . .	22
<b>5. Hardware</b>	<b>23</b>
5.1. [REVISADO] Impresión en 3D . . . . .	23
5.2. [REVISADO] Teoría de la estructura física . . . . .	25
5.3. [REVISADO] Microcontrolador utilizado . . . . .	27
5.4. Desarrollo y componentes de la PCB . . . . .	28
5.5. Comunicaciones . . . . .	28
5.6. [REVISADO] Motores empleados (actuadores) . . . . .	28
<b>6. Software</b>	<b>32</b>
6.1. S1 . . . . .	32
6.1.1. UI/UX . . . . .	32
6.1.2. Protocolo de comunicación . . . . .	32
6.1.3. Pseudolenguaje de comunicación . . . . .	32
6.1.4. Logs . . . . .	32
6.1.5. Otros . . . . .	32
6.2. S2 . . . . .	32
6.2.1. Protocolo de comunicación . . . . .	32
6.2.2. Interpretación del pseudo-lenguaje . . . . .	32
6.2.3. Cálculo de movimientos/trayectorias . . . . .	32
6.2.4. Control de los componentes . . . . .	32
6.2.5. Otros . . . . .	32
<b>7. Casos de estudio</b>	<b>33</b>
7.1. Decisiones tomadas . . . . .	33
7.2. Desarrollo de las distintas partes del proyecto . . . . .	33
<b>8. Calidad y pruebas</b>	<b>34</b>
8.1. Batería de pruebas . . . . .	34
8.2. Explicación de las pruebas . . . . .	34

8.3. Resultados esperados   resultados obtenidos . . . . .	34
8.4. Reflexión - solución . . . . .	34
<b>9. Demostración</b>	<b>35</b>
<b>10. Planificación, costes y tiempo empleado</b>	<b>36</b>
10.1. Diagramas de Gantt . . . . .	36
10.2. Sueldos propuestos y costes obtenidos . . . . .	36
10.3. Coste de los materiales inicial - coste de los materiales final . . . . .	36
10.4. Evolución del tiempo empleado . . . . .	36
10.5. Contratiempos y tiempo de desarrollo final . . . . .	36
<b>11. Conclusiones</b>	<b>37</b>
11.1. Conclusiones técnicas . . . . .	37
11.2. Experiencia personal en el desarrollo del proyecto . . . . .	37
11.3. Conocimientos adquiridos y nuevas competencias . . . . .	37
<b>12. Futuras mejoras</b>	<b>38</b>
12.1. Desarrollos e implementaciones que no han podido realizarse . . . . .	38
12.2. Ideas propuestas pero no implementadas . . . . .	38
12.3. Otras . . . . .	38
<b>Bibliografía</b>	<b>39</b>
<b>Anexos</b>	<b>42</b>

# Índice de figuras

1.1.	Flautista y tamborilero de Vaucanson [2]. . . . .	1
1.2.	En 1774, “lady musician” por Jaquet-Droz [3]. . . . .	2
1.3.	“The Turk”, creado por von Kempelen en 1769 [4]. . . . .	2
1.4.	Barco a control remoto de Nicola Tesla, en 1898 [5] . . . . .	3
1.5.	“Alpha”, el primer robot diseñado con fines militares y su posterior evolución, “Elektro”. . . . .	3
1.6.	Exposición mundial del 2005 en Japón [6]. . . . .	4
1.7.	Grados de libertad de un brazo robótico y estructura del cuerpo humano [1]. .	4
1.8.	Vista exterior del Canadarm2 [7]. . . . .	5
1.9.	Rover “Spirit”, desarrollado por la NASA y desplegado en 2004 [8]. . . . .	6
1.10.	Rover “Opportunity”, desarrollado por la NASA y desplegado en 2004 [9]. . . .	6
1.11.	Lugares de aterrizaje de los rovers de la misión espacial a Marte [10]. . . . .	7
1.12.	Robot Roomba en la estación de carga [11]. . . . .	7
1.13.	Lo que ve un Tesla cuando está en conducción autónoma nivel 2 [12]. . . . .	8
1.14.	Robot “Big-Dog” de Boston Dynamics [14]. . . . .	8
1.15.	Robot “Atlas” de Boston Dynamics [14]. . . . .	9
1.16.	Robot “KR-1000 Titan” de KUKA [15]. . . . .	9
1.17.	Robot $\mu$ Arm de UFACTORY [16]. . . . .	10
2.1.	Ejemplo de cadena cinemática [20]. . . . .	15
2.2.	Diagrama del modelo cinemático [21]. . . . .	15
3.1.	Diagrama de bloques del S2 . . . . .	19
3.2.	Diagrama de estados del método main() del orchestrator . . . . .	20

5.1. Concepto inicial del brazo robótico. . . . .	24
5.2. Logotipos de las herramientas utilizadas. . . . .	25
5.3. Flujo de trabajo del desarrollo y la impresión 3D. . . . .	25
5.4. Una estructura pantográfica que transmite el movimiento de un punto al siguiente [24]. . . . .	26
5.5. Área de trabajo de <i>pArm</i> [18]. . . . .	27
5.6. Motor de corriente continua . . . . .	29
5.7. Motor paso a paso . . . . .	29
5.8. Servomotor de corriente continua . . . . .	30
5.9. Control de posición mediante PWM [31]. . . . .	30
5.10. Servomotor Parallax utilizado [32] . . . . .	31

# Índice de cuadros

# Historial de versiones

Revisión	Fecha	Autor(es)	Descripción
0.0	27.05.2020	J. Alonso, M. Stanescu, A. Moya	Comienzo del desarrollo de la memoria.

# Definiciones, siglas, y abreviaturas

**SDK** *Software Development Kit*

**ROS** *Robot Operating System*

**SW** *software*

**OS** *Open-Source*

**OH** *Open-Hardware*

**S1** Sistema 1 – ordenador

**S2** Sistema 2 – *p*Arm

**GUI** *Graphical User Interface*

**GTK** *GIMP Toolkit*

**SoC** *System On Chip*

**PWM** *Pulse-Width Modulation*

**GPIO** *General Purpose Input/Output*

**UART** *Universal Asynchronous Receiver-Transmitter*

**RAM** *Random Access Memory*

**PLA** Ácido Poliláctico

**ABS** Acrilonitrilo Butadieno Estireno

**DSP** *Digital Signal Processor*

**PLL** *Phase Loop Lock*

- *Software Development Kit* (SDK) – colección de herramientas *software* (SW) disponibles para instalar en un único paquete.
- *Robot Operating System* (ROS) – conjunto de librerías SW que ayudan a construir aplicaciones para robots.
- *Firmware* – SW programado que especifica el orden de ejecución del sistema.
- *Graphical User Interface* (GUI) – siglas que significan “Interfaz Gráfica de Usuario” (en castellano).
- *GIMP Toolkit* (GTK) – biblioteca de componentes gráficos multiplataforma para desarrollar interfaces gráficas de usuario.
- *System On Chip* (SoC) – tecnología de fabricación que integra todos o gran parte de los módulos en un circuito integrado. Un ejemplo muy común es la placa base de un teléfono móvil, la cual es un SoC que integra todos los componentes (antenas, conversores, sensores, etc.).
- *Pulse–Width Modulation* (PWM) – señal en la cual se modifica el ciclo de trabajo de una señal periódica, para transmitir información por un canal de comunicaciones o para controlar la cantidad de energía que se envía a una carga.
- *General Purpose Input/Output* (GPIO) – pin genérico cuyo comportamiento puede ser controlado en tiempo de ejecución.
- *Universal Asynchronous Receiver–Transmitter* (UART) – estándar de comunicación dúplex simultáneos.
- Dúplex – término que define a un sistema que es capaz de mantener una comunicación bidireccional, enviando y recibiendo mensajes de forma simultánea.
- Widget – la parte de una GUI (interfaz gráfica de usuario) que permite al usuario interconectar con la aplicación y el sistema operativo.
- *Random Access Memory* (RAM) – memoria principal del ordenador, donde se guardan programas y datos, sobre la que se pueden efectuar operaciones de lectura y escritura.

- *Deep-Sleep* – estado de un microcontrolador por el cual consume muy poca cantidad de energía.
- *bit* – unidad mínima de información de una computadora.

# Capítulo 1

## Motivación y objetivos

### 1.1. [REVISADO] Estado del arte

#### 1.1.1. Desarrollo de la robótica a lo largo de la historia

El mundo de la robótica da acceso a resolver una gran variedad de problemas donde el ser humano estaba limitado físicamente: levantar cargas de gran peso, realizar tareas repetitivas durante tiempos prolongados, etc. Además, como bien se sabe, ha permitido el desarrollo de cadenas de producción en masa para poder desarrollar y crear los productos que usamos diariamente, desde el coche hasta el teléfono móvil.

Desde que se empezó a investigar en este campo, el desarrollo de los brazos robóticos ha sido exponencial: se empezó trabajando con pequeños autómatas hasta el desarrollo de la revolución industrial [1].

Los primeros modelos, como se puede ver en la figura 1.1, empezaron intentando hacer representaciones de las manos humanas. En particular, se crearon un flautista y un tamborilero los cuales eran capaces de tocar los respectivos instrumentos utilizando un complejo sistema de cables y engranajes para poder mover los “dedos” de los músicos.

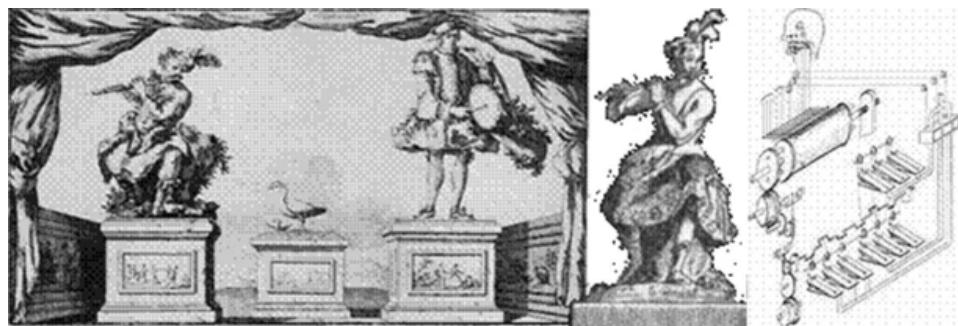


Figura 1.1: Flautista y tamborilero de Vaucanson [2].

Siguiendo con esta idea, se fue mejorando y desarrollando el modelo de imitación de las

articulaciones y los miembros de los humanos, llegando a construir estructuras más complejas y avanzadas, pensadas en su momento para poder tocar el clavicordio mediante un muñeco, como se muestra en la figura 1.2:

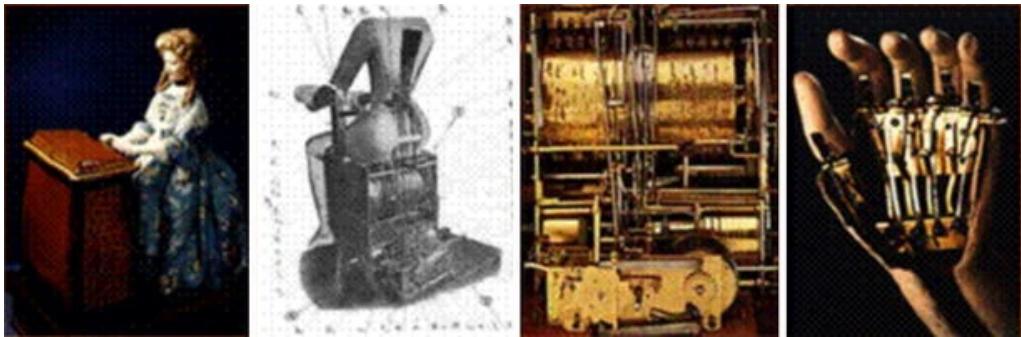


Figura 1.2: En 1774, “lady musician” por Jaquet-Droz [3].

Durante los años siguientes, el proceso se fue refinando hasta el punto de desarrollar un autómata el cual era capaz de jugar al ajedrez, llamado “The Turk” **standage\_tom\_turk\_2002**, construido en 1769. La estructura comprendía un conjunto de mecanismos los cuales eran controlados por un operador, encargado de realizar los movimientos del brazo izquierdo del autómata.

En la figura 1.3 se puede ver cómo está diseñado el sistema para mover un controlador pantográfico sobre el tablero de juego, controlado por el operador externo antes mencionado:

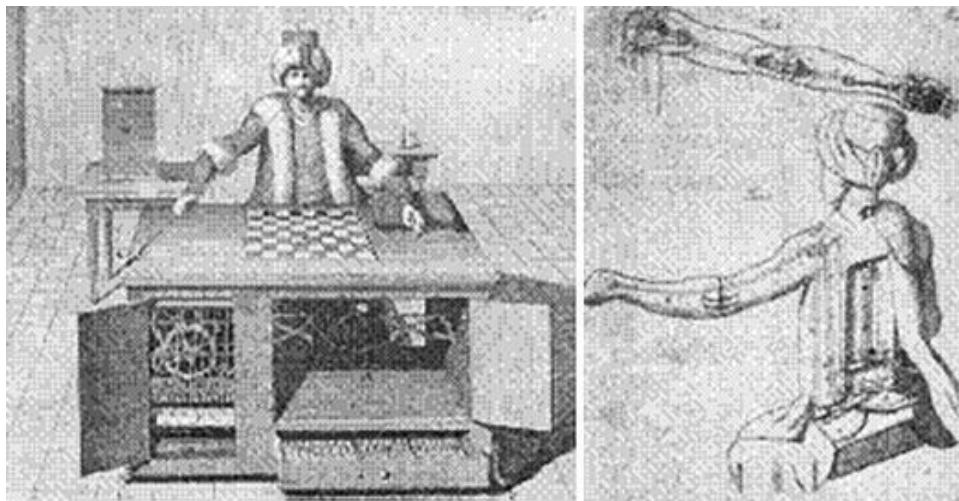


Figura 1.3: “The Turk”, creado por von Kempelen en 1769 [4].

Desde entonces, la robótica ha evolucionado y crecido de manera exponencial. Por una parte, debidas las distintas guerras que han habido en los últimos 200 años, se ha dado un gran impulso a la industria encargada de crear distintos dispositivos con fines de defensa y ataque. En particular, se potenciaron mucho los desarrollos de dispositivos por control remoto, destacando el diseño de NiKola Tesla en 1898 de un barco completamente automatizado, controlado por control remoto y sumergible, como se puede ver en la figura ??:

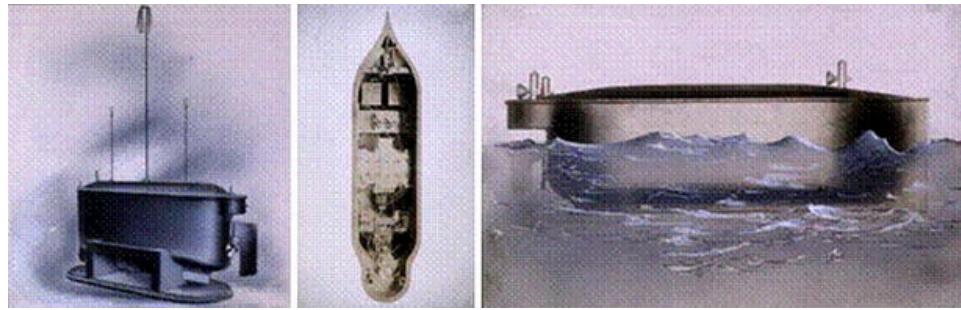


Figura 1.4: Barco a control remoto de Nicola Tesla, en 1898 [5]

Por otro lado, dada la cantidad de bajas de las Primera y Segunda Guerras Mundiales, se empezaron a desarrollar robots que permitieran sustituir a los militares en el campo de batalla, destacando en este campo el robot “Elektro”, creado por la compañía Westinghouse. Dicho robot supuso un gran éxito en la industria de los robots y armamentística, pudiendo moverse completamente, disparar armas, mover elementos faciales para “expresar emociones” e inclusive poder comunicarse.

En la figura 1.5, se puede ver a la izquierda la primera versión “Alpha” y, a la derecha, la versión mejorada “Elektro”:

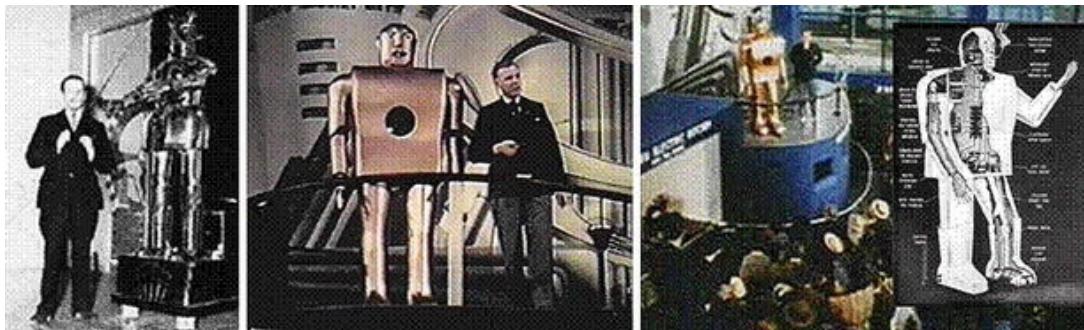


Figura 1.5: “Alpha”, el primer robot diseñado con fines militares y su posterior evolución, “Elektro”.

Toda esta evolución ha desembocado en la robótica moderna, en donde tenemos robots sofisticados y con distintos actuadores, pudiendo interactuar con muchísimos elementos de nuestro entorno y trabajar en distintas fases de producción de cadenas de montaje en serie. Además, se trabaja continuamente para que cada vez los robots puedan realizar más tareas de los humanos, mejorando cada vez más los “*end-effectors*” (controladores del final de los extremos del brazo). En la figura 1.6 se puede ver cómo robots medianamente antiguos (del 2005) ya podían realizar diversas actividades, como interactuar con las personas o tocar un instrumento.

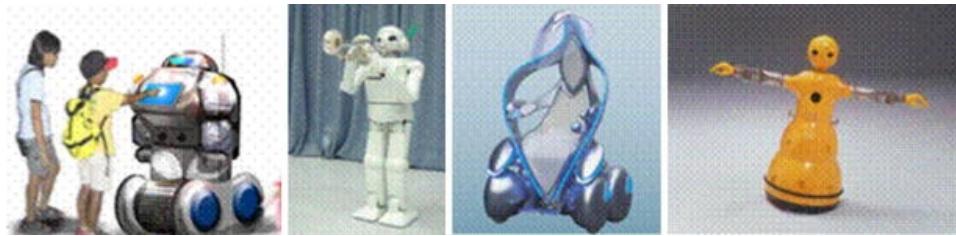


Figura 1.6: Exposición mundial del 2005 en Japón [6].

### 1.1.2. Los brazos robóticos

Con los avances actuales, el mundo de la robótica ha evolucionado a un nuevo nivel: con la inclusión de los transistores en lugar de las válvulas de vacío se han podido desarrollar circuitos integrados que manejan de manera mucho más sofisticada el control del brazo robótico.

En 1962, la empresa “Unimate” introdujo su primer brazo robótico de carácter industrial. Aproximadamente, se vendieron 8500 unidades. Este hito es importante en tanto a que se valoraron por primera vez los grados de libertad que debían de tener los brazos robóticos.

Estos planteamientos derivaron en distintos robots famosos que incluso siguen en activo hoy día. En 1969, Victor Scheinman, de la Universidad de Standford, desarrolló un brazo robótico que funcionaba alimentado por la electricidad y que se podía mover en los seis ejes, el cual se llamó “el brazo de Standford”. De forma paralela, Marvin Minsky, del MIT, desarrolló un brazo robótico para la investigación naval, para exploración submarina. En particular, el brazo tenía veinte grados de libertad ya que el brazo funcionaba mediante electricidad impulsando sistemas hidráulicos. Más tarde, Scheinman continuó desarrollando brazos robóticos, creando el “*Programmable Universal Machine for Assembly*”, más conocido como PUMA.

En la actualidad, los brazos robóticos se desarrollan y diseñan para seguir la estructura física del cuerpo humano (ver figura 1.7).

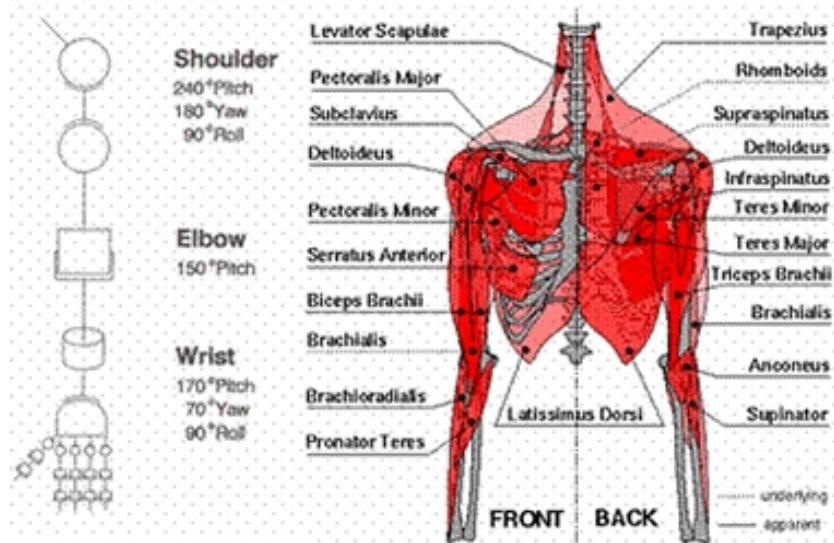


Figura 1.7: Grados de libertad de un brazo robótico y estructura del cuerpo humano [1].

De esta estructura anterior, se deducen las siguientes partes:

- Articulación del hombro: dispone de tres grados de libertad que permiten subir y bajar, ir a la izquierda y derecha, y rotar sobre sí mismo.
- Articulación del codo: el codo permite extender, contraer y reorientar tanto la muñeca como la mano. Por lo general, se estima la extensión del codo en unos 150°.
- La muñeca: compone el último elemento del brazo robótico antes de llegar al “*end-effector*”. Es de los elementos más importantes debido a su gran capacidad de movimiento en las tres dimensiones. Sin esta articulación, el brazo robótico se asemejaría en funcionalidad a un robot pantográfico. Cada vez más, las articulaciones de la muñeca se vuelven complejas y sofisticadas. La muñeca humana, por ejemplo, puede moverse 45° desde el centro, pero se reduce mucho la capacidad de rotación de la misma. En la actualidad se está investigando cómo poder mejorar la relación de movimientos para permitir una mayor movilidad, pero las singularidades siguen siendo un gran problema. Por ejemplo, el robot quirúrgico da Vinci, pese a lo avanzado que pueda parecer, tiene problemas de bloqueo de las muñecas del mismo cuando se acerca a posiciones singulares.
- La mano: supone un “*end-effector*” diferenciado que define el propósito y la capacidad del brazo robótico. La mano es una herramienta capaz de realizar múltiples acciones muy variadas entre sí. Actualmente, se sigue investigando de forma activa sobre ello para intentar implementar controles sensoriales, de presión y de movimiento en los “*end-effector*” de los robots.

### 1.1.3. La actualidad

Durante los dos últimos decenios la robótica ha evolucionado de manera exponencial. Se ha trabajado de forma activa en mejorar ciertas condiciones industriales, espaciales y en el día a día de las personas. Por un lado, en el 2001 se puso en la ISS el brazo robótico “Canadarm2”, conocido oficialmente como “*Space Station Remote Manipulator System*” (SSRMS) (figura 1.8).



Figura 1.8: Vista exterior del Canadarm2 [7].



Figura 1.9: Rover “Spirit”, desarrollado por la NASA y desplegado en 2004 [8].

Además, en 2005, se desplegaron en Marte los rovers “Spirit” (figura 1.9) y “Opportunity” (figura 1.10).

El primero supuso un gran avance de la ingeniería, ya que crearon un robot completamente autónomo para enviarlo a un terreno muy hostil. Con las seis ruedas que tenía permitía una movilidad bastante elevada en terrenos muy desiguales, usando las ruedas delanteras y traseras como ejes directrices y las centrales como ejes motrices (detalle en la figura 1.9). Además, la fisionomía de las mismas y su elevación permitía que el dispositivo se desplazara por distintos tipos de terreno de una manera óptima, pudiendo elevar y bajar ejes a placer para una mejor sujeción y tracción.



Figura 1.10: Rover “Opportunity”, desarrollado por la NASA y desplegado en 2004 [9].

Por otra parte, se puede apreciar cómo el “Opportunity” tenía una estructura bastante similar

al “Spirit” pero añadía alguna que otra mejora. La diferencia principal entre ambos era el lugar de aterrizaje (figura 1.11), ya que ambos servían para recorrer distintos puntos de la superficie marciana para recopilar datos y tomar muestras.

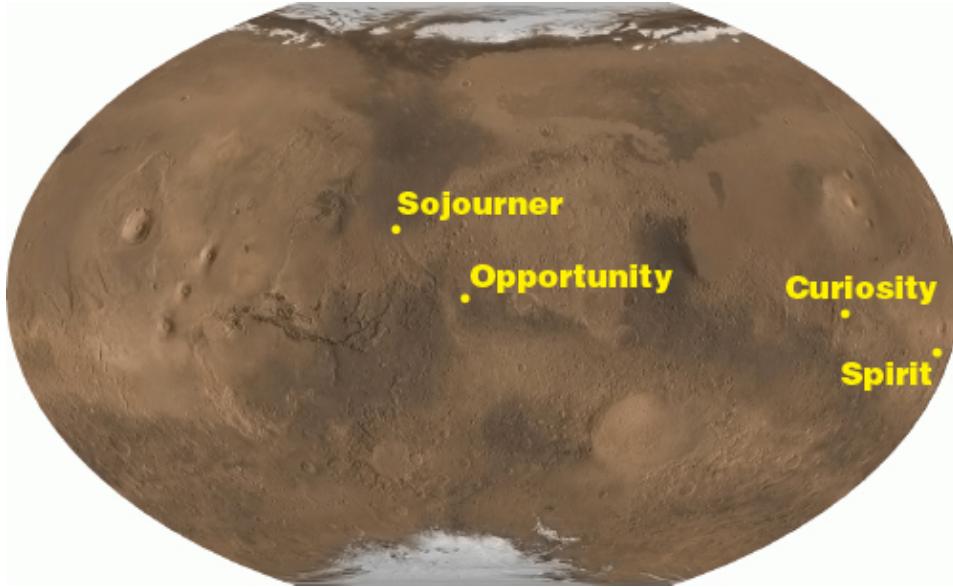


Figura 1.11: Lugares de aterrizaje de los rovers de la misión espacial a Marte [10].

La misión del “Spirit” duró 2623 soles frente a los 90 inicialmente planeados [8], y la misión del “Opportunity” duró 5352 soles frente a los también 90 inicialmente planteados [9]. Esto se traduce en 2695 días terrestres (6 años, 9 meses y 12 días) y 5498 días terrestres (15 años) respectivamente.

Por otra parte, los robots de aplicación doméstica también han ido creciendo cada vez más, naciendo en 2002 el popular Roomba (figura 1.12), de la empresa iRobot, o distintos brazos articulados para, por ejemplo, ayudar a personas que carezcan de dichos miembros o asistir a personas mayores en sus hogares.



Figura 1.12: Robot Roomba en la estación de carga [11].

Además, la industria de los coches autónomos también ha crecido de forma exponencial, sobre

todo con la llegada al mercado en 2003 de Tesla Motors y sus coches eléctricos que disponen del sistema “Autopilot” (figura 1.13), encontrándose actualmente en el nivel 2 de autonomía según la lista del SAE [12][13].



Figura 1.13: Lo que ve un Tesla cuando está en conducción autónoma nivel 2 [12].

Por otro lado, se ha avanzado mucho a nivel de robots militares y humanoides. Un ejemplo de ello es la empresa *Boston Dynamics*, la cual ha desarrollado múltiples robots con un grado de libertad bastante elevado. Dichos robots se caracterizan por una gran estabilidad y la amplia variedad de movimientos que pueden realizar: andar, correr, saltar, subir escaleras, abrir puertas, etc. Actualmente, dos robots son los principales: “*Big-Dog*” (figura 1.14) y “*Atlas*” (figura 1.15).



Figura 1.14: Robot “*Big-Dog*” de Boston Dynamics [14].

El primero tiene un amplio uso militar: debido a su forma, es muy útil para llevar cargas pesadas.

das durante largas distancias. Además, la configuración cuadrúpeda y los avanzados sistemas *software* y *hardware* del que dispone permite que el robot sea estable incluso en condiciones bastante complicadas, como puede ser un suelo helado.



Figura 1.15: Robot “Atlas” de Boston Dynamics [14].

“Atlas”, por otra parte, es un trabajo en progreso que permitirá, en un futuro, poder utilizarlo con fines militares y domésticos. Puede llevar objetos pesados con sus brazos y moverse con mucha agilidad, haciéndolo un robot muy polivalente según se quiera utilizar.

Finalmente, el desarrollo de los brazos articulados con múltiples finalidades también ha evolucionado mucho. Desde robots industriales tales como los desarrollados por la empresa KUKA, como el “KR-1000 Titan” (figura 1.16), o el brazo “M-2000”, de FANUC hasta brazos más pequeños pero igualmente hábiles, como el  $\mu$ Arm, de UFACTORY (figura 1.17).



Figura 1.16: Robot “KR-1000 Titan” de KUKA [15].



Figura 1.17: Robot  $\mu$ Arm de UFACTORY [16].

Estos robots tienen múltiples propósitos: el primero, levantar y trasladar piezas muy grandes y pesadas con muchísima precisión. El segundo, disponer de un robot para poder aprender y utilizarlo para tareas como, por ejemplo, impresión 3D. Se desarrolló con la intención de ser accesible e intentar introducir en el mundo de la robótica a aquellos que pudieran estar interesados, pero su alto coste impide el acceso a aquellos con capacidad adquisitiva más baja.

## 1.2. Motivaciones y objetivos del desarrollo del proyecto

Durante el primer semestre del cuarto curso de Ingeniería de Computadores, hay dos asignaturas las cuales propiciaron el desarrollo de este proyecto: robótica y sistemas empotrados.

Con la primera, se vio la potencia de los brazos robóticos y se desarrolló un estudio sobre un controlador el cual se ha hablado con anterioridad: el  $\mu$ Arm [17]. Con la segunda asignatura, se vio cómo con sistemas de aplicación específica se podían desarrollar circuitos con suficiente potencia como para poder tomar el control de otros dispositivos más grandes y complejos aplicando la lógica estudiada a lo largo de los años.

Se tomaron en cuenta los conocimientos obtenidos de las asignaturas anteriores para empezar un desarrollo que uniera esos dos campos: diseñar un brazo robótico impreso en 3D el cual estuviera gobernado por un microcontrolador en una placa de control de propósito específico. Para ello, se parte de los diseños 3D provistos en la web de UFACTORY [18] para su posterior adaptación y reutilización. En lo referente a la placa de control, el brazo original utiliza una placa Arduino Mega [19], por lo que se decidió (para dar mayor peso a la parte de ingeniería e intentar reducir costes) diseñar e implementar al completo una placa con otro microcontrolador para gobernar dicho brazo robótico.

Principalmente, este trabajo se desarrolla bajo las dos perspectivas siguientes:

- Aplicar en un proyecto de ingeniería real las competencias y técnicas que se han ido aprendiendo a lo largo de los distintos cursos del Grado de Ingeniería de Computadores (61CI).
- Construir una alternativa asequible y accesible, tanto a niveles de *Open-Source* (OS) y *Open-Hardware* (OH), de un brazo robótico de manera que cualquier persona interesada en este ámbito de la ingeniería pueda introducirse y aprender, e incluso montar el brazo por sí mismo.

Para la primera perspectiva, la forma de afrontarla y desarrollarla está detallada en el punto siguiente (1.3). Para la perspectiva de desarrollo de un producto accesible y asequible, se partió desde el abaratamiento de costes: el brazo original  $\mu$ Arm se encuentra disponible en venta por aproximadamente \$749. Dicho precio, pese a no ser especialmente elevado, impide a muchas personas el acceso a la robótica en un brazo que pretende ser educativo y útil. Por este motivo, se desarrolla este proyecto principalmente para resultar barato. Además, siguiendo con la política del brazo original, el proyecto se desarrolla bajo las premisas OS y OH, de manera que inclusive para aquellos que no puedan imprimir el brazo 3D se dispone de forma universal todos los diagramas, planos, esquemas, diseños y código fuente que se ha empleado para acabar desarrollando el brazo  $\mu$ Arm.

### 1.3. Metodología

Dado que se pretende hacer un desarrollo de ingeniería completo, la metodología es un punto muy importante en este proyecto.

Primeramente, antes de hacer ningún tipo de desarrollo o implementación, se hizo un estudio del problema, y de lo que se pretendía obtener. Por una parte, se comprobó hasta qué punto podrían ser reutilizables los diseños provistos por UFACTORY en su página de GitHub. Esto permitió diseñar elementos nuevos, adaptar los recursos a lo que hay disponible, etc. Por otro lado, se estudió qué placa de control se quería utilizar. Debido a la familiaridad de los integrantes del equipo con las placas de la familia “Microchip”, se plantearon distintas alternativas:

- Controladores de gama media de la familia PIC16F.
- Controladores de gama superior de la familia PIC32F.
- Controladores digitales de la señal, de la familia dsPIC.

Se optó por utilizar los últimos mencionados, ya que disponen de un control específico matricial y matemático para poder agilizar las operaciones realizadas, de forma que los cálculos necesarios se podrían realizar íntegramente en el microcontrolador.

Además, se estudió cómo se quería plantear la comunicación con el brazo: de forma completamente autónoma o mediante un equipo auxiliar. Para evitar la complejidad extra que habría surgido de desarrollar un sistema de control completamente autónomo del brazo por sí solo, se

decidió conectarlo a un equipo auxiliar externo que lo gobierne, y que el *pArm* no funcione si no es estando conectado.

Una vez se definieron estos puntos, se pasó al diseño lógico del sistema que deberán tener tanto Sistema 1 – ordenador (S1) como Sistema 2 – *pArm* (S2), mediante especificación de requisitos, diagramas lógicos, diagramas físicos, diagramas de diseño, etc. Esta parte del proyecto es de las más importantes, ya que sustenta las ideas y las funcionalidades que habrán de estar presentes en el producto final. Mientras tanto, se han ido desarrollando pruebas y mecanismos de control para ir asegurando la correcta calidad del trabajo.

Finalmente, una vez completada esta parte de diseño, se pasa a la implementación real. Dada la situación del COVID–19, esta fase de implementación se ha retrasado sobremanera, impidiendo pues presentar el proyecto en el mes de julio, como estaba previsto, y teniendo que acotar los plazos de implementación a, posiblemente, un mes. En el momento de implementación, se creará la placa diseñada y se empezará la impresión de distintas piezas 3D, para comprobar su funcionamiento en conjunto e ir solucionando los posibles errores que aparezcan.

Durante este proceso, se ha ido desarrollando además de forma paralela la memoria que acompaña el proyecto, permitiendo ir actualizándola con los últimos cambios y mejoras que se han considerado de interés para aparecer descritas.

## Capítulo 2

# [REVISADO] Explicación de la estructura del proyecto

El diseño y construcción de un brazo robótico es un proceso multidisciplinar en el que se deben emplear diversas áreas del conocimiento. Desde un primer momento, este proyecto se postuló como un proyecto completo de ingeniería, y es precisamente por eso que el proyecto está dividido en varios bloques, los cuales desempeñan una función clave en el desarrollo correcto del mismo.

El proyecto está dividido en tres grandes bloques: modelo matemático, elementos *hardware* y elementos *software*. Cada una de estas partes se encuentra a su vez subdividida en diferentes partes o hitos. Sin embargo, no es necesario describirlos con tanta precisión por el momento para poder comprender la estructura completa del proyecto.

Cabe destacar que, desde un punto de vista de ingeniería, a cada uno de los grandes bloques anteriormente mencionados se le puede asociar a una función dentro del proyecto:

- El modelo matemático es la parte más teórica del proyecto y su función es la de aportar una base formal y lógica que permita realizar cálculos sobre la cinemática del brazo robótico. Este bloque se encuentra ubicado en el apartado 4 de la memoria.
- Los elementos *hardware* del proyecto constituyen la realidad física del brazo robótico y están estrechamente relacionados con la construcción del mismo, así como con el control de los actuadores y demás componentes físicos presentes en el brazo robótico. Este bloque se encuentra ubicado en el apartado 5 de la memoria.
- Los elementos *software* del proyecto constituyen el principal mecanismo para implementar el modelo matemático y la lógica de funcionamiento del sistema completo mediante de la programación de los elementos *hardware* y de los sistemas que necesitan comunicarse con los mismos. Este bloque se encuentra ubicado en el apartado 6 de la memoria.

En cada uno de los bloques anteriores, ya sean *hardware* o *software* y requieran construcción física o implementación mediante programación, se incluye la realización de pruebas de funcionamiento así como las revisiones pertinentes.

Es importante remarcar que, debido a la complejidad del sistema, el mismo está dividido en dos subsistemas que aglutinan funcionalidades vitales para el correcto funcionamiento del manipulador robótico:

- S1 está formado por la aplicación gráfica de usuario que se utiliza para controlar manualmente los movimientos del robot. Este subsistema es un elemento *software* íntegramente y está descrito en el apartado 6.1 de la memoria.
- S2 está formado por la estructura física del manipulador, los actuadores y la placa de circuito impreso de control. Este subsistema combina elementos *hardware* y *software*, así como conceptos del modelo matemático. Los elementos *software* se describen en el apartado 6.2 de la memoria, mientras que los elementos *hardware* se describen en el apartado 5 de la memoria.

A continuación, se describen de forma detallada todos los bloques descritos anteriormente y a su vez, se mencionan las principales subdivisiones de cada uno de ellos.

## 2.1. Matemáticas

Los modelos matemáticos aplicados a proyectos de manipuladores robóticos son usados principalmente para realizar cálculos relacionados con los aspectos cinemáticos y dinámicos de los mismos.

Los aspectos cinemáticos de un manipulador robótico describen cómo es el movimiento y las trayectorias del mismo sin tener en cuenta las fuerzas que lo afectan, mientras que los aspectos dinámicos describen cómo se ve afectado dicho movimiento en función de las fuerzas que actúan sobre él.

Ambos aspectos anteriormente mencionados deben de ser descritos mediante un modelo matemático que permita realizar cálculos sobre los movimientos del manipulador.

En este proyecto, se ha llevado a cabo únicamente el modelo cinemático, dado que debido a las características físicas del prototipo a construir, es decir, velocidades de desplazamiento, peso de las articulaciones o masa máxima de carga; se ha concluido que el modelo dinámico no aportaría demasiada información útil para llevar a cabo el control del manipulador. Cabe destacar que el modelo dinámico suele presentar una complejidad mucho más elevada que el modelo cinemático en términos matemáticos y por ello se ha desecharido la posibilidad de llevarlo a cabo.

Desde un punto de vista técnico, el modelo cinemático de un manipulador robótico expresa cuál es la posición del extremo del mismo con respecto al tiempo y en función de la posición de las articulaciones del mismo. Normalmente, los brazos robóticos se pueden describir matemáticamente mediante el concepto de cadena cinemática:

Tal y como se puede apreciar en la figura 2.1, las articulaciones pueden rotar y permiten la movilidad de cada uno de los segmentos del manipulador. Dado que estas articulaciones

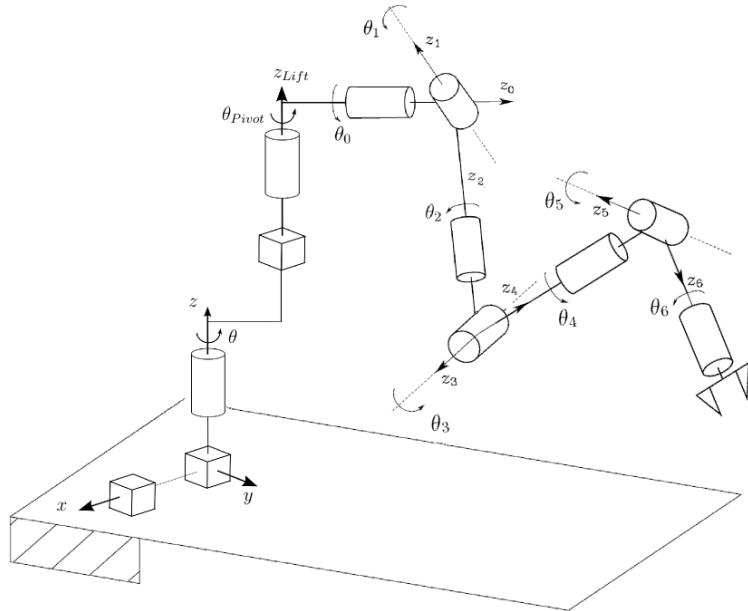


Figura 2.1: Ejemplo de cadena cinemática [20].

rotan, su posición se expresa numéricamente mediante unidades angulares. El concepto de cadena cinemática hace referencia a que, dado que cada una de las articulaciones esta unida a la siguiente mediante un segmento, se genera una cadena de movimientos en la que la posición espacial de cada una de las articulaciones se ve afectada por la posición angular de las anteriores.

Aplicando este principio, el modelo cinemático expresa matemáticamente la posición cartesiana del extremo del robot en función de las coordenadas angulares de las articulaciones. Existen pues dos perspectivas del modelo cinemático:

- El modelo de cinemática directa expresa la posición espacial del extremo del manipulador en función de las coordenadas angulares de las articulaciones.
- El modelo de cinemática inversa expresa las coordenadas angulares de las articulaciones en función de las coordenadas cartesianas del extremo del manipulador.

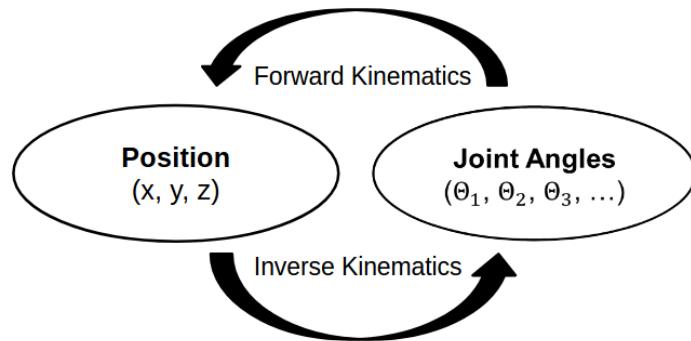


Figura 2.2: Diagrama del modelo cinemático [21].

En conclusión, el modelo matemático conforma la base teórica y formal que permite realizar el estudio de los movimientos del manipulador y es por ello que representa un bloque crucial dentro del proyecto.

## 2.2. *Hardware*

Los elementos *hardware* conforman la implementación física del manipulador y de todos los componentes empleados para controlarlo.

En términos generales, el *hardware* usado en el proyecto se descompone en diferentes elementos:

- Impresión 3D de la estructura física del manipulador.
- Motores empleados en el manipulador.
- Desarrollo de la placa de circuito impreso de control y microcontrolador empleado.
- Comunicaciones entre los diferentes subsistemas.

En primer lugar, la impresión 3D es la tecnología seleccionada para la fabricación de la estructura física del manipulador debido a su bajo coste y accesibilidad. Esta parte del proyecto se centra en llevar a cabo la fabricación y construcción de la estructura física del manipulador, así como su ensamblado y testeo. Este apartado se ubica en el apartado 5.1 de la memoria.

En segundo lugar, la elección de los motores que dotan de movilidad a la estructura es una decisión crucial y que depende principalmente de cuales sean las características físicas del manipulador, así como de las tareas que se quieran realizar con el mismo. Existen numerosas opciones en cuanto a motores, por ejemplo, motores DC, servomotores, motores paso a paso, etc. Este apartado se ubica en el apartado 5.5 de la memoria.

En tercer lugar, el desarrollo de la PCB de control y elección del microcontrolador representan la parte más importante dentro del bloque hardware del proyecto. El objetivo principal de esta parte del proyecto es llevar a cabo el diseño y construcción de una PCB personalizada, adaptada especialmente a los actuadores y microcontrolador usados para llevar a cabo el control del movimiento del manipulador. Se considera que esta PCB representa uno de los elementos hardware esenciales para el correcto desarrollo del proyecto. Este apartado se ubica en el apartado 5.3 de la memoria.

En último lugar, el diseño e implementación de los canales de comunicación y protocolos necesarios para comunicar los dos subsistemas principales requiere desarrollo hardware y software de forma equitativa, además, también representa uno de los elementos cruciales del proyecto. Este apartado se ubica en el apartado 5.4 de la memoria.

## 2.3. *Software*

Los elementos *software* del proyecto abordan los siguientes aspectos:

- Desarrollo de la aplicación de control del brazo robótico, implementada mediante una interfaz gráfica de usuario para garantizar su accesibilidad y facilidad de uso. Esta implementación se lleva a cabo en S1.
- Programación del microcontrolador e implementación del modelo matemático en la práctica con el objetivo de controlar los movimientos del brazo robótico. Esta implementación se lleva a cabo en S2.

En primer lugar, mediante el desarrollo de la aplicación de usuario se busca ofrecer una forma de controlar los movimientos del robot de forma fácil y accesible, para ello se ha desarrollado una interfaz de usuario que se ejecuta en un ordenador auxiliar. Desde esta aplicación el usuario puede controlar los movimientos del robot, además de monitorizar el estado del mismo. Las órdenes dadas por el usuario desde esta aplicación son enviadas al microcontrolador para su realización mediante los canales de comunicación mencionados anteriormente. Este desarrollo se ha llevado a cabo mediante el lenguaje de programación Python. Este apartado se ubica en el apartado 6.1 de la memoria.

En segundo lugar, la programación del microcontrolador representa una parte esencial del proyecto, ya que toda la lógica de funcionamiento y control de los actuadores del brazo robótico se lleva a cabo en el mismo. Es por ello que la labor principal del microcontrolador es orquestar el funcionamiento de los actuadores, así como de realizar el computo necesario para transformar las órdenes del usuario en movimientos consecuentes del brazo robótico. La programación del microcontrolador se ha llevado a cabo mediante el lenguaje C. Este apartado se ubica en el apartado 6.2 de la memoria.

# Capítulo 3

## Diagramas, requisitos y diseño

Una parte importante de un proyecto integral de ingeniería es la elicitation de requisitos y la creación de diagramas que representen el sistema de manera abstracta en base a dichos requisitos.

El sistema de gobierno del p-Arm esta compuesto de dos subsistema al ser necesaria tanto una placa de control como un ordenador auxiliar desde el cual un operario humano pueda interactuar con el brazo. El software del sistema de control del ordenador será representado mediante un diagrama de clases mientras que el software que ira cargado en la placa de control será representado por un diagrama de bloques general y varios diagramas de estados que detallarán el comportamiento del sistema.

Para realizar dichos diagramas se ha empleado Papyrus, una herramienta de edición gráfica para realización de diagramas. Para modelizar los diagramas del sistema de control que ira en el ordenador auxiliar se ha empleado el estándar UML2 definido por la OMG, por otro lado, para realizar los diagramas del software que será cargado en la placa de control se ha empleado el estándar SysUML 1.4 ya que permite mejor representación del sistema empotrado.

A continuación se pueden observar los requisitos que el equipo de desarrollo ha elicited además de una explicación de los diagramas que representan los dos subsistemas que componen el proyecto,

### 3.1. Requisitos

Copiar requisitos aqui para hacer Jutsu de + 10 paginas de memoria

### 3.2. Diagramas y diseño

En base a los anteriores requisitos el grupo de desarrollo ha generado los siguientes diagramas para el software de la placa de control.

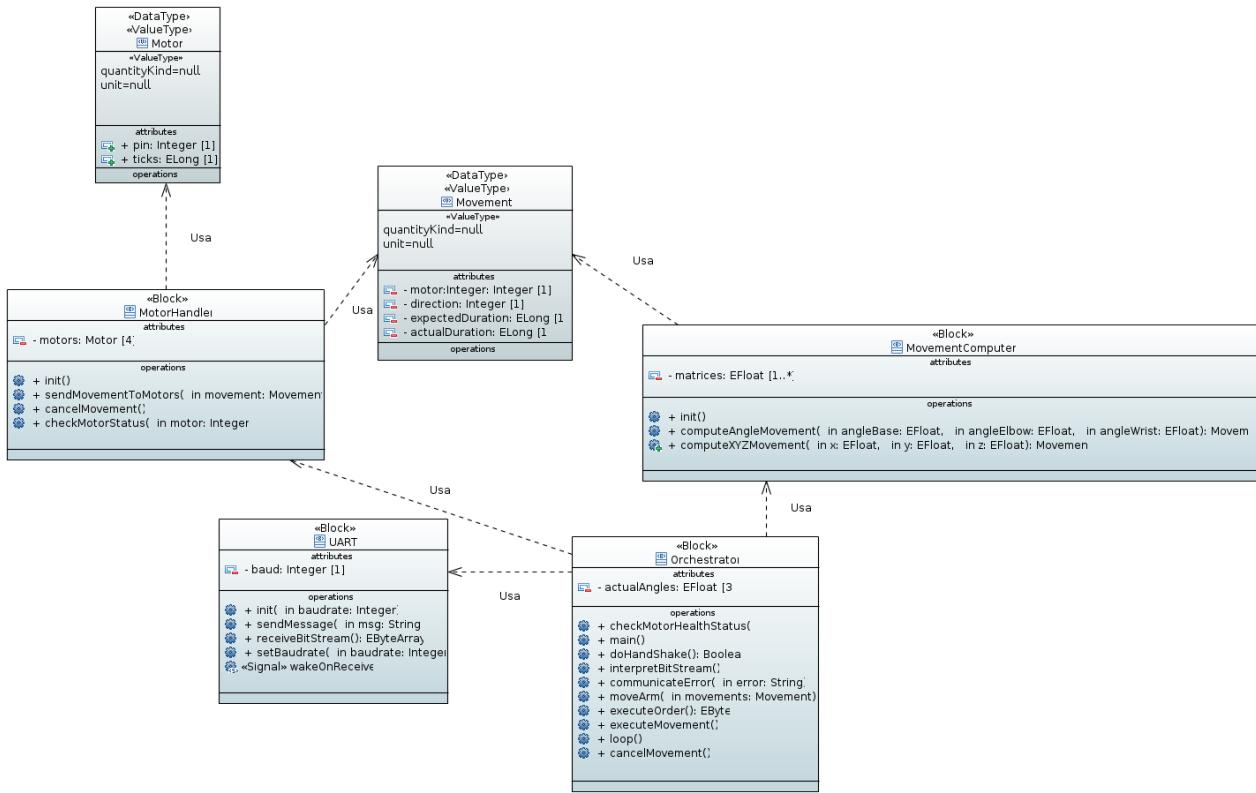


Figura 3.1: Diagrama de bloques del S2

En el diagrama 3.1 se pueden observar los bloques que componen el S2 además de dos tipos de datos los cuales han sido creados para facilitar el control de los motores del brazo.

A continuación se explican cada uno de los bloques:

- **MotorHandler**: Este bloque es capaz de controlar los motores de manera directa empleando el tipo de dato "Movement" enviando la señal necesaria para realizar el movimiento requerido. Además permite verificar el estado de los motores y cancelar los movimientos si esto fuese necesario.
- **UART**: Este bloque es el encargado de la comunicación asíncrona entre el S1 y el S2. Controla el ratio de baudios de la comunicación y realiza la transmisión y la recepción de información hasta y desde el S1. A través de este bloque se reciben las órdenes procedentes del S1 y se envían los errores y la posición del brazo al S1 desde el S2.
- **Orchestrator**: Encargado de coordinar los demás bloques. En él se encuentra la lógica principal del S2. Algunas de sus funciones más importantes son interpretar el flujo de bits que llega desde el S1 para obtener una orden concreta; ordenar el movimiento del brazo empleando los demás bloques o hacer el handshake inicial entre el S1 y el S2. Posteriormente se entrará mas en detalle en el comportamiento de este bloque al analizar los diagramas de estados.
- **MovementComputer**: Se encarga de computar el movimiento que se tendrá que comunicar a los motores. Para ello deberá obtener las posiciones deseadas gracias al bloque UART

y al Orchestrator.

A continuación se explican las dos estructuras de datos que se aprecian en el diagrama 3.1

- Motor: Este tipo de dato es empleado por el MotorHandler para saber a que pin debe mandar la señal PWM que gobierna los motores y durante cuantos ticks deberá estar activa dicha señal
- Movement: El MovementComputer genera un array de 3 posiciones de este tipo de dato, uno por cada motor de giro del brazo. El atributo "motor" guarda un integer que representa uno de los motores del brazo; "direction" sirve para conocer la dirección de giro de dicho motor; .expectedDuration" guarda la duración

A continuación se explican los diagramas de estados de cada uno de los métodos que aparecen en el diagrama de bloques general.

En el caso del Orchestrator tenemos los siguientes diagramas.

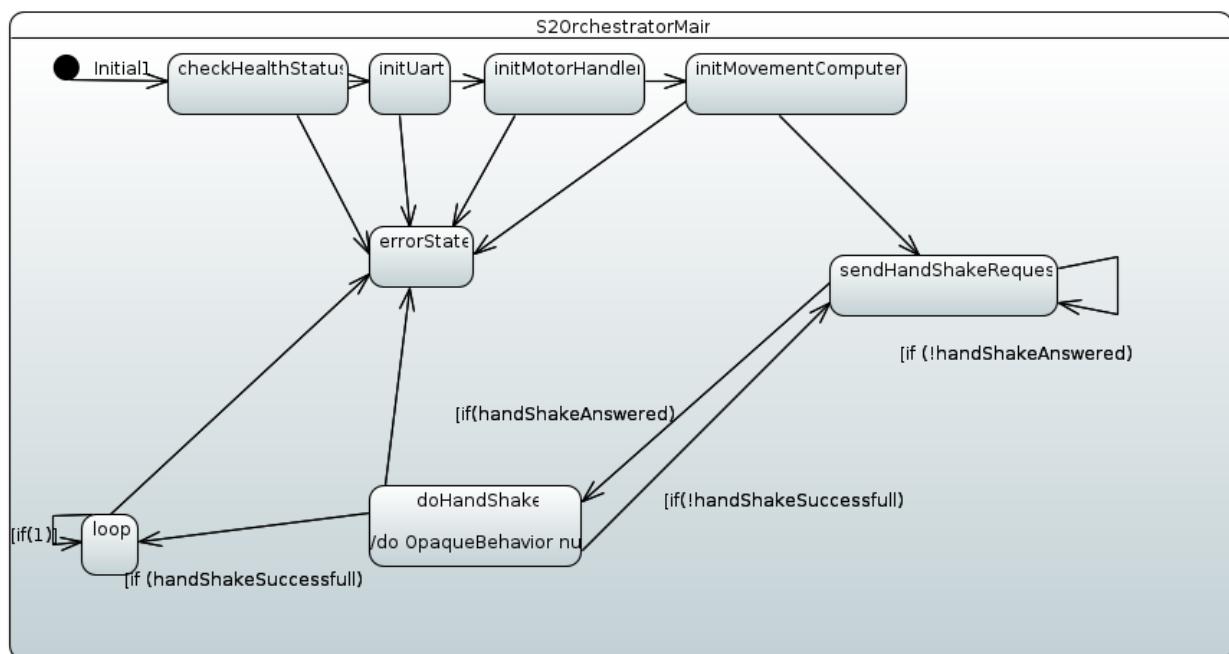


Figura 3.2: Diagrama de estados del método main() del orchestrator

Este método solo se ejecutará una vez, en cuanto el sistema se ponga en marcha.

- checkHealthStatus: Se verifica la situación de los componentes del brazo robótico para confirmar que todos están en un estado adecuado para el funcionamiento.
- initUart: Se inicializa la UART definiendo un ratio de baudios concreto.
- initMotorHandler: Se inicializa el controlador de los motores.

- initMovementComputer : Se inicializa el computador de movimientos.
- sendHandShakeRequest : Se manda una petición de handshake para verificar si hay algún ordenador conectado. Si se detecta alguno se pasa al siguiente estado, si no, se mantiene en ese estado mandando requests.
- doHandShake : Si en el estado anterior se detecta un ordenador se pasa a este estado. Se realiza una serie de intercambios de información para verificar que el ordenador conectado es adecuado para el control del brazo.
- loop: Se pasa al bucle de funcionamiento si el handshake ha sido correcto.
- errorState: Estado de error al que se llega si en alguno de los estados ocurre algún problema inesperado.

## Capítulo 4

### Las matemáticas del proyecto

4.1. Cinemática directa

4.2. Cinemática inversa

4.3. Funciones jacobianas

4.4. Implementación final realizada

# Capítulo 5

## *Hardware*

### 5.1. [REVISADO] Impresión en 3D

Las razones por las cuales se toma la decisión de fabricar la estructura del brazo mediante impresión 3D se detallan a continuación:

- Cumplir con el requisito de replicabilidad y asequibilidad: una de las bases del proyecto es que pueda ser reproducible a bajo coste tanto de recursos como de tiempo. Se decide por tanto construir la estructura física del brazo mediante técnicas de impresión 3D, ya que están altamente extendidas y son cada vez más asequibles.
- Características físicas del material: el plástico utilizado para la impresión es ligero y suficientemente resistente para soportar las cargas para las que está pensado el manipulador.
- Disponibilidad de impresora 3D: dado que la Universidad es capaz de proveer al equipo con una impresora 3D funcional, los costes del proyecto se abaratan si la estructura es realizada con los medios de los que ya se disponen.
- Simplificar el proceso de mejora y personalización: debido a la naturaleza OS y OH del proyecto, se espera que las personas interesadas puedan contribuir a el mejorándolo y/o personalizándolo. Además, la impresión 3D facilita estas acciones.

En particular, la impresora que la Universidad pone a disposición del equipo de trabajo es la “*Ultimaker 3 Extended*”, la cual es capaz de imprimir en una alta variedad de materiales, de los cuales destacan los siguientes:

- Ácido Poliláctico (PLA)[22]: este material permite imprimir de manera segura con alta precisión dimensional y una resistencia a la tracción excepcional, que además soporta grandes velocidades de impresión y es biodegradable, ya que se obtienen a partir de almidón de maíz o de yuca o mandioca, o de caña de azúcar.

- Acrilonitrilo Butadieno Estireno (ABS)[23]: material que presenta buena adhesión entre capas y una resistencia a temperaturas de hasta 85ºC. Permite obtener buenos detalles estéticos.
- Ultimaker Nylon: este material es un tipo de poliamida basada en los polímeros plásticos PA6/66. Presenta una absorción de humedad reducida así como una capacidad considerable de resistencia ante tensiones mecánicas junto con un bajo coeficiente de fricción, haciéndolo un material ideal para construcciones mecánicas.
- CPE y CPE+: este material presenta una alta estabilidad dimensional, con buena resistencia al impacto y a la temperatura. Debido a su alta solidez y su estabilidad dimensional ofrece un buen rendimiento mecánico y gran resistencia al desgaste.

Debido a la naturaleza mecánica del proyecto, el equipo ha decidido emplear materiales con alta resistencia mecánica y el Ultimaker Nylon junto con el CPE cumplen con dicha característica.

Aprovechando la licencia original GPL 3.0 del  $\mu$ Arm, se ha recuperado el modelo 3D proporcionado por UFACTORY como punto de partida. A partir de este modelo se han impreso las piezas que hemos decidido conservar para nuestro proyecto, a saber, la estructura general del brazo, exceptuando los soportes de los motores, los cuales tendrán que ser adaptados a los motores que se han decidido utilizar para este proyecto.

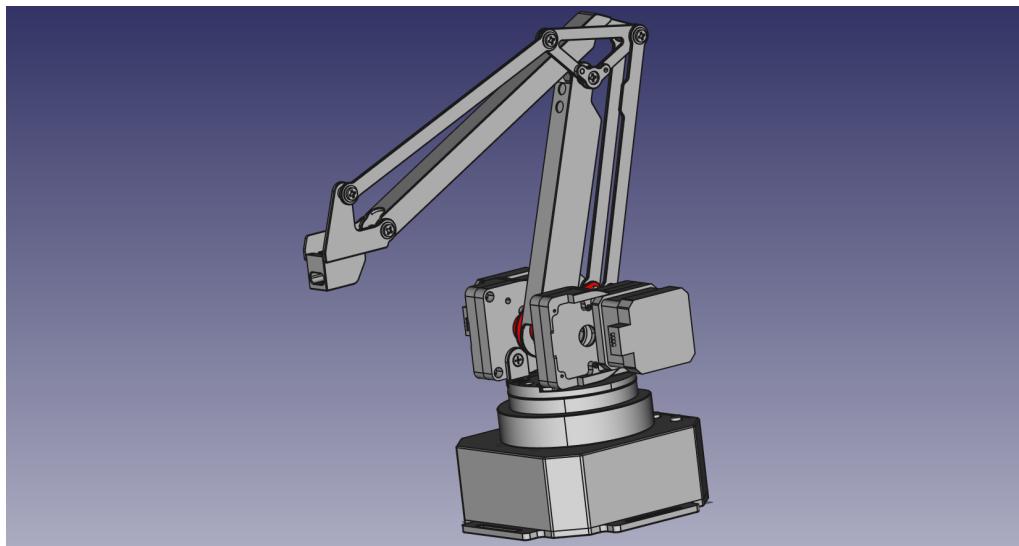


Figura 5.1: Concepto inicial del brazo robótico.

Las herramientas que han sido empleadas para visualizar y modificar el modelo y posteriormente imprimir las piezas han sido respectivamente FreeCAD y Slic3r.



Figura 5.2: Logotipos de las herramientas utilizadas.

El flujo de trabajo que se ha seguido desde el modelo 3D hasta la impresión de una pieza ha sido el mostrado en la figura 5.3:

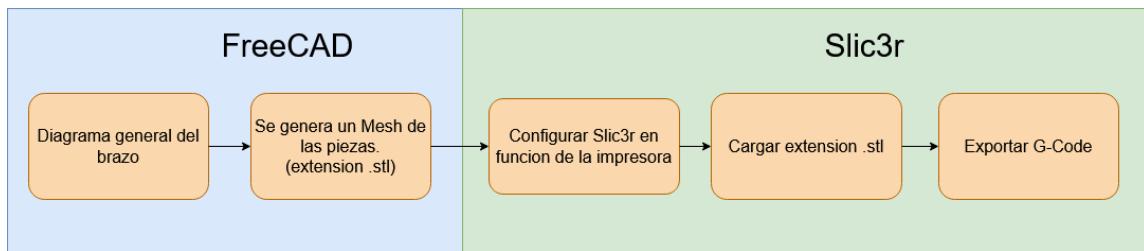


Figura 5.3: Flujo de trabajo del desarrollo y la impresión 3D.

## 5.2. [REVISADO] Teoría de la estructura física

La estructura del brazo es pantográfica, lo cual significa que es un sistema de enlaces mecánicos que reproduce el movimiento de un punto de una articulación en un segundo punto, normalmente a un tamaño o bien más pequeño o bien más grande, como se puede apreciar en la figura 5.4. Se originó en el siglo XVII y su aplicación más conocida es como instrumento de dibujo.

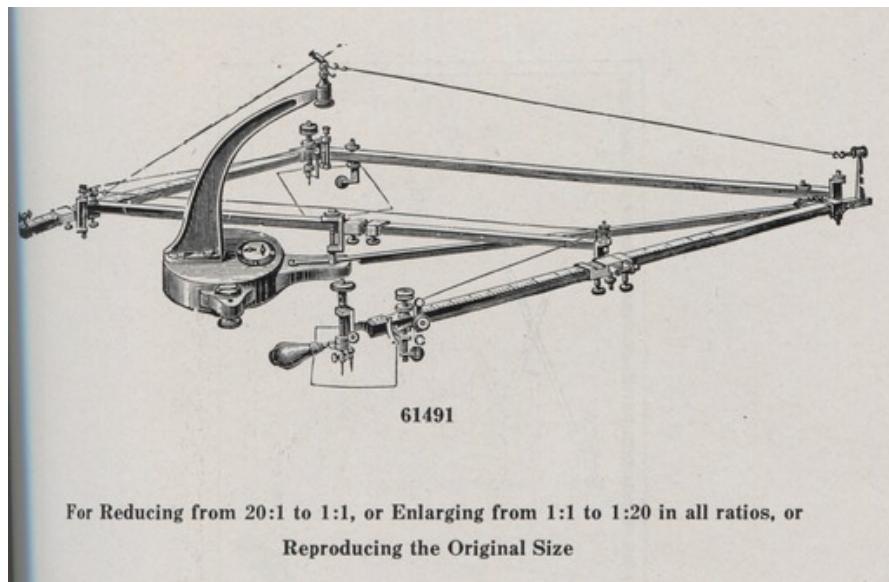


Figura 5.4: Una estructura pantográfica que transmite el movimiento de un punto al siguiente [24].

El hecho de que la estructura del brazo sea pantográfica permite controlar todas las articulaciones mediante motores ubicados en la base. Esto es de especial importancia ya que hace posible que las articulaciones finales no carguen con el peso de los motores, permitiendo emplear materiales como el plástico para la construcción de la estructura y, además, dando capacidad al brazo para levantar cargas más pesadas.

Otro beneficio derivado de la ubicación de los motores en la base es la alta estabilidad del brazo, ya que siendo estos los componentes más pesados y estando ubicados en la base se consigue un centro de gravedad muy cercano a la superficie de apoyo, permitiendo así un amplio rango de movimientos, como se puede ver en la figura 5.5.

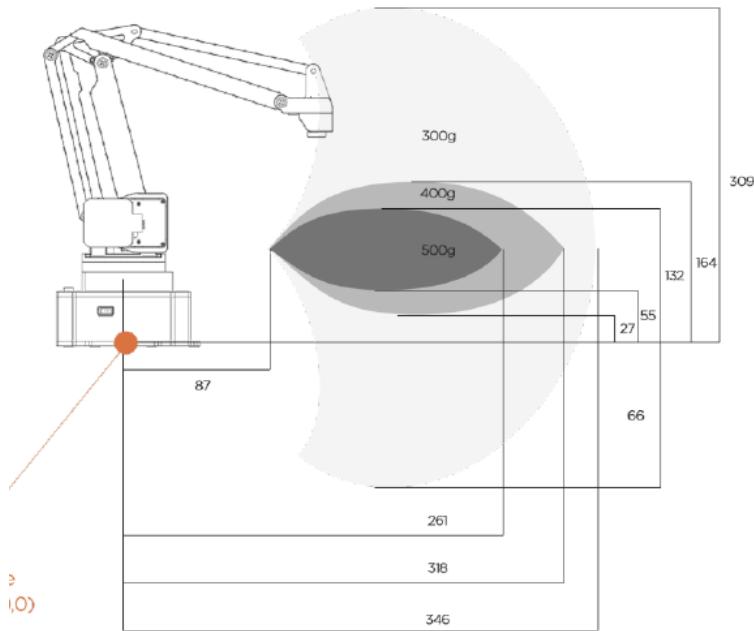


Figura 5.5: Área de trabajo de *pArm* [18].

Otra característica a destacar es que debido a la estructura pantográfica, el *end-effector* mantiene siempre un ángulo perpendicular con la superficie de apoyo del brazo. Esto supone, por un lado, perder ciertos grados de libertad pero por otro, simplificar la estructura y abaratar el coste de producción.

### 5.3. [REVISADO] Microcontrolador utilizado

Para el desarrollo de la placa que conforma el S2 se ha empleado un microcontrolador “dsPIC33EP512GM604”. Los motivos por los cuales se ha usado este modelo son los siguientes:

- En primera instancia, a la cantidad y la precisión de los PWM que este ofrece, ya que son suficientes para poder controlar todos los motores y su precisión permite generar una señal adecuada para controlar a cada uno de ellos.
- Por otro lado, debido a la naturaleza de los cálculos que se deben realizar para convertir posiciones cartesianas a ángulos y viceversa, el *Digital Signal Processor* (DSP) facilita la obtención de los valores de las diferentes operaciones matriciales que permiten esta conversión.
- Otro aspecto importante es la posibilidad de almacenar hasta 512KB de memoria de programa.
- Por último, se ha elegido un DSP de Microchip debido a que todos los integrantes del grupo de desarrollo tiene experiencia previa con este fabricante. Además, dicho fabricante proporciona documentación extensa sobre sus productos.

Una parte crítica del proyecto es la precisión y el control de los motores. En este aspecto, los PWM del DSP permiten generar ciclos de trabajo a partir de un registro de 16 bits. Para una frecuencia de  $50Hz$  (período de  $20ms$ ), un giro de  $360^\circ$  supondría una duración del nivel alto del PWM  $5ms$ . Es decir, se podrá controlar la rotación entre  $0^\circ$  y  $360^\circ$  con  $2^{16}/4 = 16384$  bits de control. Esto supone que obtendremos una precisión máxima de  $360/16384 = 0'02197^\circ$ , lo cual es suficiente para el proyecto.

En cuanto a las operaciones matemáticas, el DSP cuenta con un circuito *Phase Loop Lock* (PLL), el cual permite incrementar la frecuencia del oscilador para conseguir de esta manera una mayor cantidad de microinstrucciones por segundo y, por tanto, un mayor volumen de operaciones.

Debido a la complejidad del proyecto y a la gran variedad de sistemas que debemos emplear, el código de programa se espera sea considerablemente grande. 512KB son suficientes para almacenar la memoria de programa para este proyecto.

Este microcontrolador gracias a que dispone de puertos UART permite recibir las órdenes y los movimientos necesarios desde el S1. Desde S2, se realizará las operaciones matemáticas necesarias y después se encargará de generar las señales PWM necesarias para mover los motores de tal manera que el brazo robótico quede en la posición deseada.

## 5.4. Desarrollo y componentes de la PCB

## 5.5. Comunicaciones

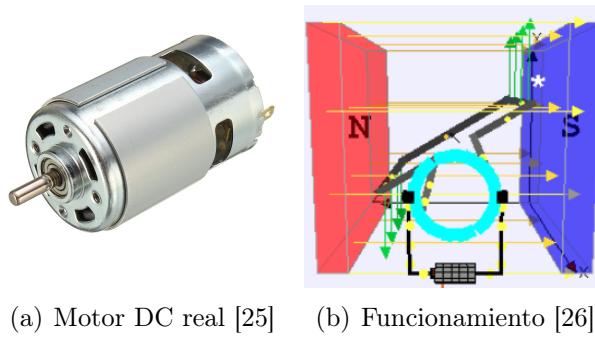
## 5.6. [REVISADO] Motores empleados (actuadores)

Dado que en este proyecto se ha planteado la construcción de un manipulador o brazo robótico, los únicos actuadores empleados han sido motores, los cuales dotan de movilidad a la estructura física del brazo.

Cabe destacar que, debido a la forma de la estructura física del brazo y dado a que el mismo tiene tres articulaciones móviles, se han empleado tres motores principales en cada una de ellas y un motor auxiliar en el extremo o *end-effector* del manipulador.

Existen varios tipos de motores eléctricos que pueden ser usados para dotar de movilidad a proyectos de robótica de pequeña escala. Sin embargo los principales tipos se pueden agrupar en las siguientes tres categorías:

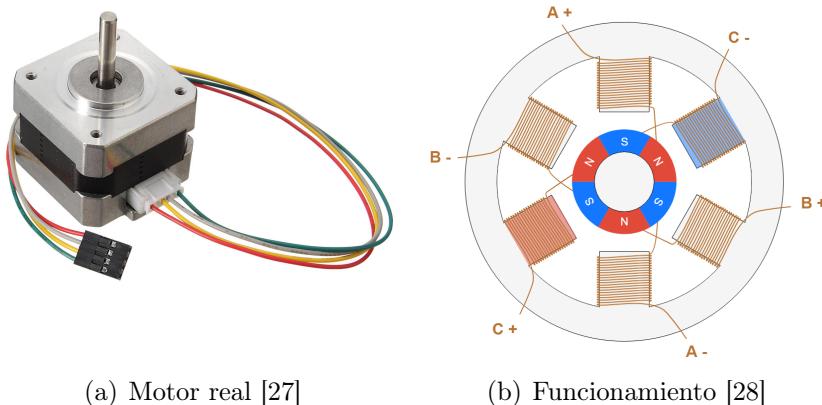
- Motores de corriente continua: son los motores eléctricos más sencillos y básicos. Debido a esto, realizar el control de la posición angular del eje y su velocidad de rotación es complicado y requiere aplicar técnicas de control de lazo cerrado. Además, el control físico de este tipo de motores se lleva a cabo mediante una señal PWM actuando sobre un puente H.
- Motores paso a paso: se trata de motores eléctricos más complejos que ofrecen una



(a) Motor DC real [25] (b) Funcionamiento [26]

Figura 5.6: Motor de corriente continua

precisión muy alta en cuanto a control de posición y velocidad, ya que descomponen su movimiento en pasos de longitud constante. En este tipo de motores se puede realizar control de velocidad y posición del eje del motor mediante técnicas de control de lazo abierto, dado que en este tipo de motores se controla el número de pasos que da el motor, así como cada cuánto tiempo se produce un paso. Este tipo de motores suelen necesitar un *driver* específico para ser controlados.



(a) Motor real [27] (b) Funcionamiento [28]

Figura 5.7: Motor paso a paso

- Servomotores: se trata de motores de corriente continua que incorporan un sistema de control de posición de lazo cerrado. Por ello, este tipo de motores ofrecen un control muy simple de la posición angular del eje del motor. A través de una señal PWM enviada al motor, se puede establecer una posición consigna que el eje del motor debe cumplir. Estos motores incluyen un sensor de posición o *encoder* que determina la posición angular del eje del motor y un pequeño microcontrolador que verifica la posición actual del eje en comparación con la posición de consigna establecida, realizando las correcciones necesarias hasta alcanzar dicha posición angular.

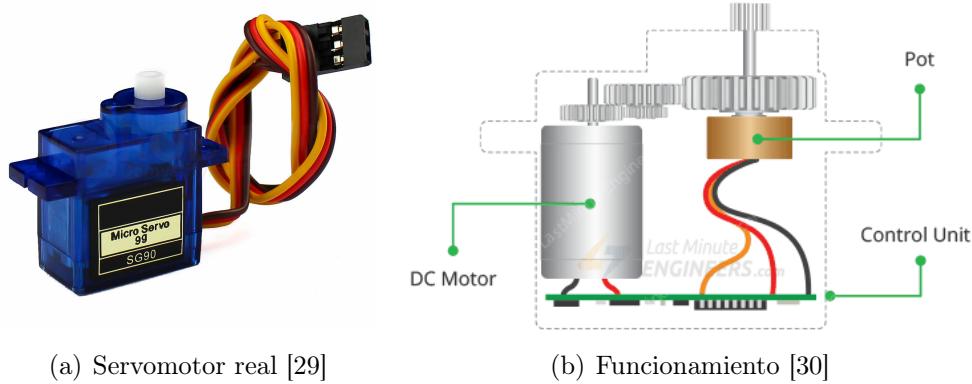


Figura 5.8: Servomotor de corriente continua

Tras analizar los diferentes tipos de motores anteriormente expuestos, se ha decidido utilizar servomotores para dotar de movilidad al brazo robótico. Esta decisión se fundamenta en los siguientes motivos:

- Este tipo de motores ofrece un control de posición preciso y simple mediante una señal PWM. A pesar de que dicho control de posición se realiza mediante lazo cerrado internamente dentro del motor, desde un punto de vista externo, no se necesita ningún tipo de realimentación externa, *driver* adicional o circuito de puente H para controlarlos.

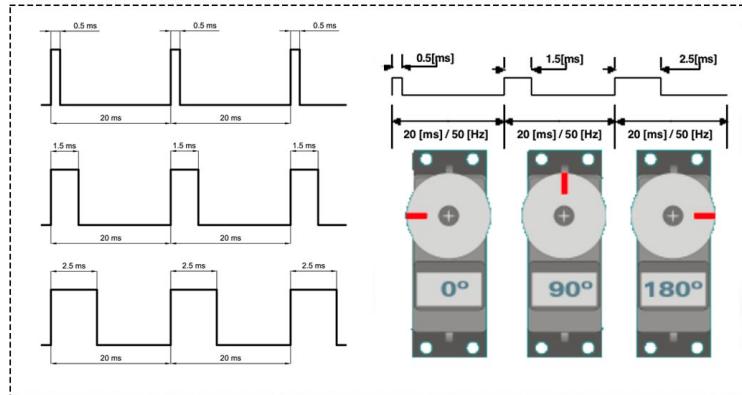


Figura 5.9: Control de posición mediante PWM [31].

- Se trata de motores que se adaptan muy bien para proyectos de robótica de pequeña escala, debido a su bajo coste y sencillez de uso.
- Este tipo de motores está muy extendido en el mercado y existen numerosos modelos con diferentes potencias, tamaños, etc.

Es importante destacar que existen dos tipos de servomotores:

- Servomotores de giro limitado: son aquellos servomotores que tienen un rango de rotación limitado, el cual suele ser normalmente de 180º. Son el tipo de servomotor más sencillo.

- Servomotores de giro continuo: son aquellos servomotores que tienen rango completo de giro, es decir, pueden realizar giros de  $360^\circ$ .

Dado que ninguna de las articulaciones del motores está diseñada para realizar giros de más de  $180^\circ$ , se han empleado servomotores de giro limitado.

Otro de los datos que es importante clarificar antes de tomar la decisión de que motores van a ser usados en un proyecto de robótica, es la carga máxima que va tener que desplazar el manipulador robótico. Este dato afecta principalmente al diseño de la estructura física del brazo y a la potencia de los motores escogidos, en especial, el torque que ejercen.

Finalmente, el modelo de servomotor elegido para las articulaciones ha sido el *Parallax 900-00005 Standard Servo* el cual tiene las siguientes características técnicas:

- Servomotor de rango limitado de  $180^\circ$ .
- Control mediante señal PWM de 50Hz.
- Alimentación de entre 4V y 6V, utilizando entre 15mA y 200mA. Potencia nominal de 140mA.
- Torque máximo ejercido de  $27N \cdot cm$ , es decir aproximadamente  $2,75Kg \cdot cm$ . Este torque garantiza una carga máxima de unos 80g - 100g para el brazo robótico.
- Peso de 44g.
- Dimensiones 406 x 55,8 x 19 mm



Figura 5.10: Servomotor Parallax utilizado [32]

Teniendo en cuenta los datos técnicos anteriores, este modelo de servomotor se adapta perfectamente a las características del brazo robótico que se ha desarrollado, cumpliendo todas las cualidades deseadas para que el funcionamiento del brazo robótico sea correcto.

# Capítulo 6

## *Software*

### 6.1. S1

#### 6.1.1. UI/UX

#### 6.1.2. Protocolo de comunicación

#### 6.1.3. Pseudolenguaje de comunicación

#### 6.1.4. Logs

#### 6.1.5. Otros

### 6.2. S2

#### 6.2.1. Protocolo de comunicación

#### 6.2.2. Interpretación del pseudo-lenguaje

#### 6.2.3. Cálculo de movimientos/trayectorias

#### 6.2.4. Control de los componentes

#### 6.2.5. Otros

# Capítulo 7

## Casos de estudio

7.1. Decisiones tomadas

7.2. Desarrollo de las distintas partes del proyecto

# Capítulo 8

## Calidad y pruebas

8.1. Batería de pruebas

8.2. Explicación de las pruebas

8.3. Resultados esperados | resultados obtenidos

8.4. Reflexión - solución

# Capítulo 9

## Demostración

# Capítulo 10

## Planificación, costes y tiempo empleado

10.1. Diagramas de Gantt

10.2. Sueldos propuestos y costes obtenidos

10.3. Coste de los materiales inicial - coste de los materiales final

10.4. Evolución del tiempo empleado

10.5. Contratiempos y tiempo de desarrollo final

# Capítulo 11

## Conclusiones

- 11.1. Conclusiones técnicas
- 11.2. Experiencia personal en el desarrollo del proyecto
- 11.3. Conocimientos adquiridos y nuevas competencias

# Capítulo 12

## Futuras mejoras

- 12.1. Desarrollos e implementaciones que no han podido realizarse
- 12.2. Ideas propuestas pero no implementadas
- 12.3. Otras

# Bibliografía

- [1] M. E. Moran, «Evolution of robotic arms,» *Journal of Robotic Surgery*, vol. 1, n.º 2, págs. 103-111, jul. de 2007, ISSN: 1863-2491. DOI: [10.1007/s11701-006-0002-x](https://doi.org/10.1007/s11701-006-0002-x).
- [2] J. de Vaucanson, *Le mécanisme du flûteur automate: présenté à Messieurs de l'Académie royale des Sciences : avec la description d'un canard artificiel et aussi celle d'une autre figure également merveilleuse, jouant du tambourin et de la flûte.* chez Jacques Guerin, imprimeur-libraire, 1738.
- [3] Chapuis, Alfred and Droz, Edmond, *Automata: A Historical and Technological Study.* L'Editions du Griffon, 1958.
- [4] Standage, Tom, *The Turk: The Life and Times of the Famous Eighteenth-Century Chess-Playing Machine.* Walker company, 2002.
- [5] Belarmino, J and Moran, ME and Firoozi, F and Capello, S and Kolios, E and Perrotti, M, «Tesla's Robot and the Dawn of the Current Era. Society of Urology and Engineering,» 7.<sup>a</sup> ép., vol. 19, A214, J Endourol 2005.
- [6] ——, «An Oriental Culture of Robotics—the Coming Maelstrom. Society of Urology and Engineering,» 7.<sup>a</sup> ép., vol. 19, A119, J Endourol 2005.
- [7] *Mobile Servicing System.* feb. de 2020. dirección: [https://en.wikipedia.org/w/index.php?title=Mobile\\_Servicing\\_System&oldid=942154747](https://en.wikipedia.org/w/index.php?title=Mobile_Servicing_System&oldid=942154747) (visitado 14-06-2020).
- [8] Spirit (*Rover*). jun. de 2020. dirección: [https://en.wikipedia.org/w/index.php?title=Spirit\\_\(rover\)&oldid=960186651](https://en.wikipedia.org/w/index.php?title=Spirit_(rover)&oldid=960186651) (visitado 15-06-2020).
- [9] Opportunity (*Rover*). mayo de 2020. dirección: [https://en.wikipedia.org/w/index.php?title=Opportunity\\_\(rover\)&oldid=960035335](https://en.wikipedia.org/w/index.php?title=Opportunity_(rover)&oldid=960035335) (visitado 15-06-2020).
- [10] *The Mars Rovers: Spirit and Opportunity* |textbackslashtextbar NASA Space Place – NASA Science for Kids. dirección: <https://spaceplace.nasa.gov/mars-spirit-opportunity/en/> (visitado 15-06-2020).
- [11] *History of Robots.* jun. de 2020. dirección: [https://en.wikipedia.org/w/index.php?title=History\\_of\\_robots&oldid=961981460](https://en.wikipedia.org/w/index.php?title=History_of_robots&oldid=961981460) (visitado 15-06-2020).
- [12] R. Baldwin, *Tesla's Video Shows What Its Autopilot System Sees.* feb. de 2020. dirección: <https://www.caranddriver.com/news/a30733506/tesla-autopilot-full-self-driving-video/> (visitado 15-06-2020).

- [13] *Tesla's Full Self-Driving Computer Is Now in All New Cars and a next-Gen Chip Is Already 'Halfway Done'.* dirección: <https://social.techcrunch.com/2019/04/22/teslas-computer-is-now-in-all-new-cars-and-a-next-gen-chip-is-already-halfway-done/> (visitado 15-06-2020).
- [14] *Boston Dynamics.* mar. de 2020. dirección: [https://es.wikipedia.org/w/index.php?title=Boston\\_Dynamics&oldid=123964032](https://es.wikipedia.org/w/index.php?title=Boston_Dynamics&oldid=123964032) (visitado 15-06-2020).
- [15] *KR 1000 titan.* dirección: <https://www.kuka.com/es-es/productos-servicios/sistemas-de-robot/robot-industrial/kr-1000-titan> (visitado 15-06-2020).
- [16] *UFACTORY Official Website.* dirección: <https://store.ufactory.cc/> (visitado 15-06-2020).
- [17] *UPM-Robotics/Uarm,* Robotics UPM, 19 de nov. de 2019. dirección: <https://github.com/UPM-Robotics/uarm> (visitado 28-06-2020).
- [18] *UFACTORY xArm | textbackslashtextbaruArm.* dirección: <https://www.ufactory.cc/#/en/uarmswift> (visitado 28-01-2020).
- [19] (). «Arduino Mega 2560 Rev3 | Arduino Official Store,» dirección: <https://store.arduino.cc/arduino-mega-2560-rev3> (visitado 28-06-2020).
- [20] S. Sharma y C. Scheurer, «Generalized Unified Closed Form Inverse Kinematics for Mobile Manipulators With Reusable Redundancy Parameters,» 6 de ago. de 2017. DOI: [10.1115/DETC2017-68104](https://doi.org/10.1115/DETC2017-68104).
- [21] (). «Roboy 2.0 - Inverse Kinematics,» dirección: <https://ik-test.readthedocs.io/en/latest/> (visitado 28-06-2020).
- [22] *Ácido poliláctico.* mayo de 2020. dirección: [https://es.wikipedia.org/w/index.php?title=%C3%81cido\\_polil%C3%A1ctico&oldid=126434778](https://es.wikipedia.org/w/index.php?title=%C3%81cido_polil%C3%A1ctico&oldid=126434778) (visitado 16-06-2020).
- [23] *Acrilonitrilo butadieno estireno.* jun. de 2020. dirección: [https://es.wikipedia.org/w/index.php?title=Acrilonitrilo\\_butadieno\\_estireno&oldid=126716603](https://es.wikipedia.org/w/index.php?title=Acrilonitrilo_butadieno_estireno&oldid=126716603) (visitado 16-06-2020).
- [24] (). «The Pantograph in Context,» dirección: <https://www.circuitousroot.com/artifice/letters/press/typemaking/making-matrices/pantograph-in-context/index.html#:~:text=A%20pantograph%20is%20a%20system,an%20increased%20or%20decreased%20size.&text=The%20so%2Dcalled%20%22parallel%20motion,transmission%20of%20power%20and%20motion> (visitado 23-06-2020).
- [25] (). «775-9009F-C-CC Datasheet ( Hoja de Datos) - DC Motor,» dirección: <http://www.datasheet.es/PDF/917192/775-9009F-C-CC-pdf.html> (visitado 28-06-2020).
- [26] *Motor de corriente continua,* en Wikipedia, la enciclopedia libre, Page Version ID: 127266375, 26 de jun. de 2020. dirección: [https://es.wikipedia.org/w/index.php?title=Motor\\_de\\_corriente\\_continua&oldid=127266375](https://es.wikipedia.org/w/index.php?title=Motor_de_corriente_continua&oldid=127266375) (visitado 28-06-2020).
- [27] Banggood.com. (). «3pcs 42mm 12V Nema 17 Stepper de dos fases motor para impresora 3D Office Equipment from Ordenador & Reds on banggood.com,» dirección: <https://www.banggood.com/es/3pcs-42mm-12V-Nema-17-Two-Phase-Stepper-Motor-For-3D-Printer-p-1556469.html> (visitado 28-06-2020).
- [28] (). «Stepper Motors: Types, Uses and Working Principle | Article | MPS,» dirección: <https://www.monolithicpower.com/en/stepper-motors-basics-types-uses> (visitado 28-06-2020).

- [29] (). «Mini Servomotor - Ebotics,» dirección: <https://www.ebotics.com/es/producto/mini-servomotor/> (visitado 28-06-2020).
- [30] (20 de oct. de 2019). «How Servo Motor Works & Interface It With Arduino,» dirección: <https://lastminuteengineers.com/servo-motor-arduino-tutorial/> (visitado 28-06-2020).
- [31] *Zona Maker - Servo-Motores.* dirección: <https://www.zonamaker.com/electronica/intro-electronica/componentes/motores/servo-motores> (visitado 16-06-2020).
- [32] (). «900-00005 | Servomotor Parallax Inc 140 mA, 4 → 6 V | RS Components,» dirección: <https://es.rs-online.com/web/p/servomotores/7813058/> (visitado 28-06-2020).

## Anexos