



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

**“DISEÑO E IMPLEMENTACIÓN DE UN ROBOT PARA LA
AYUDA A NIÑOS CON AUTISMO EN SUS INTERACCIONES
SOCIALES A TRAVÉS DE LA IMITACIÓN”**

INFORME DE PROYECTO INTEGRADOR

Previo a la obtención del Título de:

INGENIERO/A EN ELECTRÓNICA Y AUTOMATIZACIÓN

SEBASTIAN MARCELO FUENZALIDA TAPIA

KEYLA MABEL TOAPANTA TAPIA

GUAYAQUIL – ECUADOR

AÑO: 2019

AGRADECIMIENTOS

Gracias a mi familia y amigos por permitirme conformar parte de sus vidas, aprender de ellos y acompañarme en un camino que muchas veces no ha sido fácil. Gracias a todos que desde niño estuvieron ahí, para verme crecer y ayudarme sin pedir nada a cambio. Gracias por la oportunidad de vivir y de disfrutar esa vida conmigo. Gracias por el apoyo y la confianza entregada en cada decisión, gracias por ser un soporte que contribuyó a un logro más en la vida de un hombre que mira con ansías un futuro brillante. Mis sinceros agradecimientos a todos los profesores que me brindaron conocimiento y una breve, pero fuerte idea sobre los valores. Gracias por creer en mí y por todo el afecto que de tantas formas diferentes he recibido.

SEBASTIAN MARCELO FUENZALIDA TAPIA

A mi familia, por apoyarme a lo largo de toda la carrera y concederme la oportunidad de formarme como profesional en esta prestigiosa universidad. De igual forma, agradezco mi tutor de tesis, quien durante el desarrollo del proyecto aportó con sus conocimientos y consejos mismos que me permiten culminar hoy con este trabajo. Asimismo, a los docentes que, con su conocimiento y apoyo, motivaron mi desarrollo como persona y profesional en la Escuela Superior Politécnica del Litoral que fue sede de todo el conocimiento adquirido a lo largo de estos años.

KEYLA MABEL TOAPANTA TAPIA

DEDICATORIA

Los logros de este proyecto van dedicados a todas las personas que contribuyeron positivamente para su culminación, sobre todo a dos personas que lograron plasmar su infinito amor para que pudiera escribir estas palabras. A mi madre, Leonor Fuenzalida, que luchó incansablemente para mi formación y representó a un ser humano de calidad y un modelo a seguir por sus valores e inteligencia. Ella ha sido la base sobre la que me he desarrollado y la consejera fiel en un largo camino que sobrellevamos juntos. Sin sus sacrificios esto no hubiera sido posible. A Jorge Espinel, un padre adoptivo que siempre me guio, apoyó y confió en mí, y que espero haber enorgullecido en cada momento. Sin él y todo su aprecio mi vida no hubiera sido tan alegre y divertida. Me enorgullezco en haber conocido a tan gran persona y querido a un excelente amigo. Todo lo positivo en mí se lo debo a ellos.

SEBASTIAN MARCELO FUENZALIDA TAPIA

El presente proyecto esta dedicado a mis padres Guillermo y Jenny quienes a lo largo de la carrera demostraron su apoyo incondicional, amor y sacrificio que fueron claves para lograr cumplir esta meta trazada varios años atrás. A todos mis familiares y amigos que siempre estuvieron presentes con oraciones, consejos y palabras de aliento que de una u otra manera aportaron a mi formación profesional.

KEYLA MABEL TOAPANTA TAPIA

DECLARACIÓN EXPRESA

"La responsabilidad y la autoría del contenido de este Trabajo de Titulación, me(nos) corresponde exclusivamente; y doy(damos) mi(nuestro) consentimiento para que la ESPOl realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"

Sebastian Fuenzalida

Keyla Toapanta

RESUMEN

El presente proyecto tiene sus fundamentos en la necesidad de mejorar la calidad de vida de niños con TEA (Trastorno de Espectro Autista), estos niños se caracterizan por sus complicaciones en la interacción social, comportamientos repetitivos e intereses limitados. Estas son las mismas características que no permiten que el niño tenga un proceso de aprendizaje estándar. Debido a que estos niños aprenden mejor en entornos tecnológicos se desarrolla a Loly, un robot interactivo que trabaja en paralelo con una aplicación móvil, creando un entorno de aprendizaje acorde las necesidades de los pacientes con TEA.

Para el desarrollo de Loly se aplica la metodología de producción de prototipos estándar, que consiste en planeación, conceptualización, diseño esquemático, desarrollo fabril, pruebas y lanzamiento. Es necesario recalcar que las herramientas que se utilizaron para el diseño e implementación del robot son la placa OpenCM9.04 y su respectivo entorno de programación para el módulo de control electrónico, para el módulo de actuadores se utilizan los servomotores Dynamixel y para el control general se utilizó un mini CPU Intel NUC.

Para el análisis de resultados se realizaron pruebas del robot funcionando en paralelo con la aplicación, durante estas pruebas se pudo constatar, que se realizó un control de actuadores con alto nivel de precisión, que logra una perfecta sincronización con el audio y las animaciones proyectadas en la pantalla del robot.

Finalmente se pudo evidenciar la relevancia de un correcto desarrollo de la cinemática inversa, observando como utilizando los datos de orientación adquirido mediante una IMU se logra que el robot imite los movimientos de una persona, obviamente está respetando las limitaciones de su espacio de trabajo.

Palabras Clave: Control, actuadores, cinemática inversa, cinemática directa, robot paralelo

ABSTRACT

This project has its foundations in the need to improve the quality of life of children with ASD (autism spectrum disorder), these children are characterized by their complications in social interaction, repetitive behaviors and limited interests. These are the same characteristics that do not allow the child to have a standard learning process. Because these children learn better in technological environments, an interactive robot that works in parallel with a mobile application is developed to Loly, creating a learning environment according to the needs of patients with ASD.

For the development of Loly, the standard prototype production methodology is applied, which consists of planning, conceptualization, schematic design, factory development, testing and launch. It is necessary to emphasize that the tools that were used for the design and implementation of the robot are the OpenCM9.04 board and its respective programming environment for the electronic control module, for the actuator module Dynamixel servomotors are used and for general control an Intel NUC mini CPU was used.

For the analysis of the results, tests of the robot were carried out running in parallel with the application, during these tests it was possible to verify that an actuator control was carried out with a high level of precision, which achieves a perfect synchronization with the audio and the animations projected in the robot screen.

Finally, the relevance of a correct development of inverse kinematics can be evidenced, observing how using the orientation data acquired through an IMU allows the robot to imitate the movements of a person, of course it is respecting the limitations of its work space.

Keywords: Control, actuators, inverse kinematics, forward kinematics, parallel robot

ÍNDICE GENERAL

DEDICATORIA	3
RESUMEN	I
ABSTRACT	II
ÍNDICE GENERAL	III
ABREVIATURAS	VI
SIMBOLOGÍA	VII
ÍNDICE DE FIGURAS	VIII
ÍNDICE DE TABLAS	IX
CAPÍTULO 1	10
1. Introducción	10
1.1 Descripción del problema	10
1.2 Justificación del problema.....	11
1.3 Objetivos.....	11
1.3.1 Objetivo General	11
1.3.2 Objetivos Específicos	12
1.4 Alcance	12
1.5 Marco Teórico.....	13
1.5.1 Robótica	13
1.5.2 Robótica Social	13
1.5.3 Trastorno del espectro autista.....	14
1.5.4 Servomotores.....	14
1.5.5 OpenCM9.04	15
1.5.6 ROBOTIS OpenCM IDE.....	15
1.5.7 Teleoperación de robots.....	16

1.5.8	Robot Paralelo	17
1.5.9	IMU.....	18
1.5.10	Ejes principales de navegación	19
1.5.11	Cinemática Directa e Inversa de un robot	19
CAPÍTULO 2		21
2.	Metodología	21
2.1	Metodología de diseño.....	21
2.2	Esquema de los módulos del proyecto	23
2.3	Módulo Electrónico de control de movimiento	23
2.3.1	Submódulo placa de desarrollo	23
2.3.2	Submódulo de actuadores	26
2.4	Obtención de funciones de la Cinemática Directa y la Cinemática Inversa ..	29
2.5	Programación de la placa OpenCM9.04	30
2.6	Obtención de posición y orientación con IMU de movimientos de una cabeza humana.....	32
2.7	Aplicación de cinemática inversa para la obtención de grupos de valores articulares por cada movimiento	32
2.8	Concepto de diseño	33
3.	Análisis De Resultados	35
3.1	Precisión en el servomotor Dynamixel AX-12A	35
3.2	Análisis de Carga para Dynamixel AX-12A.....	36
3.3	Comunicación para la teleoperación.....	37
3.4	Análisis de transmisión de audio y video	39
3.5	Programación del OpenCM 9.04.	40
3.6	Análisis de la cinemática directa e inversa	40

3.6.1	Programación para la resolución de sistemas de ecuaciones no lineales	42
-------	---	----

CAPÍTULO 4		44
4.	Conclusiones Y Recomendaciones	44
	Conclusiones	44
	Recomendaciones	45
5.	Bibliografía	46
ANEXOS		48

ABREVIATURAS

ESPOL	Escuela Superior Politécnica del Litoral
ASTM	American Society for Testing and Materials
TEA	Trastorno del Espectro Autista
IMU	Inertial Measurement Units

SIMBOLOGÍA

kg	Kilogramo
m	Metro
cm	Centímetro
N	Newton
V	Voltio
A	Amperio

ÍNDICE DE FIGURAS

Figura 1.1 Robot humanoide. [3].....	14
Figura 1.2 Tarjeta de desarrollo OpenCM9.04 [8].....	15
Figura 1.3 ROBOTIS OpenCM IDE [10]	16
Figura 1.4 Robot teleoperado [12]	17
Figura 1.5 IMU de 9 grados de libertad con sensores integrados [15].....	18
Figura 1.6 Ángulos de Navegación [16]	19
Figura 1.7 Cinemática de un robot [18].....	20
Figura 2.1 Esquema general del proyecto [20]	23
Figura 2.2 Diagrama de bloques de OpenCM9.04. [21].....	25
Figura 2.3 Tipos de placas OpenCM9.04. [22]	25
Figura 2.4 Vista Frontal oblicua de la cabeza del Robot Social	27
Figura 2.5 Vista Lateral de la cabeza del Robot Social.....	27
Figura 2.6 Dynamixel AX-12A [23].....	28
Figura 2.7 Dynamixel XL-320 [24].....	29
Figura 2.8 Análisis geométrico de la cabeza robótica.....	30
Figura 2.9 Representación gráfica de Loly [25].....	33
Figura 2.10 Diagrama de conexión del robot.....	34
Figura 3.1 Programa para la resolución del problema cinemático inverso [26].....	43

ÍNDICE DE TABLAS

Tabla 2.1 Metodología de diseño.....	22
Tabla 2.2 Diferencias principales entre los tipos de OpenCM9.04.....	24
Tabla 3.1 Límites de barridos de los servomotores de la cabeza robótica	35
Tabla 3.2 Prueba de Ángulos deseados Vs. Ángulos obtenidos	36
Tabla 3.3 Consumo de potencia del prototipo a plena carga.....	37
Tabla 3.4 Tabla de resultados de pruebas de transmisión	38
Tabla 3.5 Pruebas de Comunicación	39

CAPÍTULO 1

1. INTRODUCCIÓN

La presente tesis se refiere al tema de diseño e implementación de un robot social como una herramienta educativa para la ayuda a niños con autismo en sus interacciones sociales. Considerando el impacto de los robots sociales y las aplicaciones móviles de forma emergente en áreas de la educación, se ha optado por una integración del robot a una aplicación móvil en un dispositivo inteligente. El robot consta de una cámara para la supervisión del paciente con TEA (Trastorno del Espectro Autista), parlantes que emiten sonido según la interacción con la aplicación y una cabeza robótica que simula movimientos reales. Como complemento a este robot, se tiene un pico y dos alas que tienen como rol dar una forma humanoide al robot. Este tipo de robots requiere conocimientos de áreas de Electrónica y Automatización, Computación y Diseño Gráfico; por lo que su desarrollando amerita un equipo multidisciplinario supervisado por un profesional sobre el TEA.

1.1 Descripción del problema

Para la mayoría de las personas, la interacción social es parte de la cotidianidad y no representa ninguna dificultad, sin embargo, para muchos niños con trastorno del espectro autista (TEA) esto representa un verdadero desafío. Las manifestaciones más relevantes de esta familia de trastornos son: la dificultad para mantener contacto visual, reconocer el lenguaje corporal y comprender las emociones de otras personas.

En Ecuador no existe un registro oficial sobre la población con trastorno del espectro autista. Sin embargo, las distintas fundaciones privadas dedicadas únicamente al TEA se encuentran actualmente realizando sus propias estadísticas en las que se estima que en el país 1 de cada 144 niños padece el mencionado trastorno.

Durante la última década, diferentes universidades de Ecuador han desarrollado proyectos de investigación basados en el mencionado trastorno. Entre estos se encuentran aplicaciones móviles educativas, mismas que no han proporcionado los resultados esperados; a pesar de ser diseñados acorde las necesidades del grupo objetivo. Una de las causas

identificadas es la dificultad de mantener al paciente enfocado en una sola actividad. Sin mencionar que cada paciente con TEA es diferente, por ende, algunos métodos resultan efectivos en ciertos pacientes y en otros no.

1.2 Justificación del problema

Para analizar la problemática que nos lleva a desarrollar un robot social es necesario mencionar sus causas. Pacientes diagnosticados con un trastorno neurológico del espectro autista (TEA) cursan con alteraciones de la interacción social y de la comunicación, y un repertorio de comportamientos muy restringido y repetitivo. Las alteraciones en las funciones ejecutivas, problemas de atención, hiperactividad e impulsividad son características distintivas de los mismos. Es común observar que niños con TEA no hablen más allá de unas pocas palabras, y que les cueste relacionarse socialmente; incluso con las personas que le son más queridas. Parecen estar siempre ensimismados, encerrados en sí mismo, y, de hecho, muchas veces les resulta prácticamente imposible centrar su atención en cualquier cosa o persona que se presente a su alrededor. Los robots sociales, a diferencia de las personas, presentan patrones de comportamiento sencillos y fácilmente segregables. Las expresiones faciales que presentan son fácilmente identificables independientemente, y los niños con TEA tienen menor dificultad para interactuar con sus secuencias de comunicación. Los estímulos simples y repetitivos logran que el niño preste atención gracias a la facilidad de interacción y provocación de interés. Para estos niños, resulta más fácil interactuar con robots que con otro ser humano, y se pueden programar para repetir tareas o adaptarse a las necesidades concretas de cada uno. El robot pasa a tomar el rol de alguien amistoso, algo que en los humanos les cuesta interpretar con efectividad.

1.3 Objetivos

1.3.1 Objetivo General

Desarrollar una herramienta educativa que ayude a niños con autismo en sus problemas de atención, mediante la integración de un robot social con una aplicación móvil.

1.3.2 Objetivos Específicos

Implementar la transmisión y recepción de instrucciones de movimiento, del servidor (robot) al microcontrolador, para el control de posición del robot.

Realizar la programación de un microcontrolador y el diseño electrónico para la generación de señales de control de los servomotores.

Analizar el modelo cinemático directo e inverso del robot social para el cumplimiento de rutinas de movimiento.

1.4 Alcance

Para este proyecto, se realizará la implementación de una cabeza robótica que servirá como interfaz entre una aplicación móvil educativa y el paciente con autismo; proporcionando a éste indicaciones de cómo utilizarla y realizando movimientos para mantener la atención del niño enfocada.

Posteriormente al desarrollo de un robot funcional, con el uso de una cámara y sensores se espera realizar visión artificial que contribuya a que el robot detecte y siga el rostro del paciente. La detección y seguimiento del rostro logrará que el paciente con TEA juegue con el robot a través de movimientos de la cabeza.

Mediante este proyecto se otorga la capacidad a un especialista que controle el robot a distancia, mismo que será teleoperado con el fin de realizar un tratamiento específico para cada paciente. El especialista tendrá acceso al monitoreo e historial de las actividades del paciente con el robot; por ello, el paciente podrá comportarse naturalmente sin la interacción de terceros en un nuevo ambiente de juego y aprendizaje controlado.

Este proyecto tiene como meta ofrecer una nueva posibilidad de tratamiento por medio de la teleoperación a un especialista que no se vea limitado al tratamiento tradicional solamente presencial. Usando el desarrollo tecnológico en áreas como la robótica y a fines, se pretende contribuir en la investigación y el desarrollo de proyectos de la Universidad para la sociedad; con el fin de tener la información necesaria para crear robots sociales y

evitar desventajas como los costos de importación o desconocimiento de la constitución del robot.

1.5 Marco Teórico

1.5.1 Robótica

La robótica moderna, marco su punto de partida en la década en los 60s, desde entonces esta ciencia encargada de la creación de sistemas artificiales capaces de realizar tareas humanas de manera autónoma ha tenido un desarrollo acelerado.

En la actualidad la robótica está experimentando una notoria ampliación de sus campos de actuación, los robots están tomando posiciones en aplicaciones fuera de la industria, caracterizada fundamentalmente por la falta de estandarización del producto y del proceso y la no estructuración del entorno. Estos robots denominados sociales o de servicio están aún en estado de desarrollo incipiente, cabe recalcar que es previsible un considerable desarrollo de estos. [1]

1.5.2 Robótica Social

Un robot social es un agente autónomo físicamente incorporado que se comunica e interactúa con los humanos en un nivel emocional, también siguen patrones de comportamiento social, tienen varios "estados de ánimo" y se adaptan a lo que aprenden a través de sus interacciones. [2]

Los robots sociales en su mayoría son humanoides o animaloides en su forma, misma de alta relevancia, ya que su función es la interacción con humanos en un nivel emocional; este tipo de interacción tiene como fundamento la percepción visual, táctil y la comunicación verbal.

Los parámetros que caracterizan a un robot social son forma, función y contexto desde un punto de vista holístico. Cada robot tiene una forma estética dependiendo de su función a cumplir. Además, cada robot tiene diferentes tipos de comportamiento, es decir, funciones variables. Finalmente, el robot social con su forma estética y sus funciones se encuentra dentro de varios contextos; contando con una amplia gama de tipos encargado cada uno de satisfacer diferentes necesidades.



Figura 1.1 Robot humanoide. [3]

1.5.3 Trastorno del espectro autista

El autismo es un trastorno neuropsiquiátrico cuyas características son el deterioro severo en la interacción social, la deficiencia de comunicación y los patrones de comportamiento e interés restringidos, estereotipados, o ambos, los síntomas comienzan en la infancia, por lo general, en los 3 primeros años [4]. Trastorno del espectro autista (TEA) y autismo son términos generales para un grupo de trastornos complejos del desarrollo cerebral.

El término “espectro” hace referencia a la amplia gama de síntomas y fortalezas que pueden tener las personas con estos trastornos. El diagnóstico de los trastornos del espectro autista incluye principalmente el trastorno autista, el síndrome de Asperger y el trastorno generalizado del desarrollo no especificado. Aunque los trastornos del espectro autista comienzan en la niñez, pueden durar toda la vida. [5]

1.5.4 Servomotores

Un servomotor es un actuador giratorio que proporciona un control preciso de posición angular, velocidad y aceleración. Para llegar a la posición objetivo, estos movimientos se llevan a cabo con la conversión de energía eléctrica en mecánica. La simplicidad de los servomotores es la característica que hace que este actuador sea tan confiable. [6]

Para este proyecto, los servos interactúan con las piezas de la estructura de la cabeza robótica, permitiendo un control de movimiento mediante el control de pulsos.

1.5.5 OpenCM9.04

La placa OpenCM9.04 es una placa de desarrollo basada en un microcontrolador, útil para el desarrollo de proyectos con servomotores por soportar diferentes tipos de comunicación. Destaca como característica especial su aplicación en programación y control de servomotores Dynamixel por soportar la comunicación TLL. Los códigos fuente de esta placa son de código abierto, esto quiere decir que el desarrollo de software se basa en la colaboración abierta; teniendo la atención de usuarios que comparten contenido buscando beneficios prácticos en sus proyectos. La programación de esta placa se puede llevar a cabo con la plataforma de Arduino basada en el lenguaje C. Diversos códigos para sensores y actuadores utilizados en robótica gozan de librerías que facilitan la programación en esta plataforma. [7]

La placa OpenCM9.04 interactúa con los servomotores, controlando sus posiciones mediante el envío de tramas de datos dependiendo de cada tipo de servomotor.



Figura 1.2 Tarjeta de desarrollo OpenCM9.04 [8]

1.5.6 ROBOTIS OpenCM IDE

Este software de código abierto facilita escribir código y cargarlo en cualquier placa de desarrollo compatible de la familia OpenCM. Se puede modificar en Windows, Mac OS X y Linux. La programación se basa en el lenguaje del sistema abierto ARDUINO, por lo que es accesible a un amplio perfil de usuarios. El entorno de desarrollo integrado también llamado IDE (sigla en inglés de Integrated Development Environment) se compone por un conjunto de herramientas de programación. Consiste en

un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). Puede dedicarse en exclusiva a un solo lenguaje de programación o bien puede utilizarse para varios lenguajes. [9]

En la programación con este software podemos usar trozos de código hechas por terceros denominados librerías. Estas librerías, en propósitos de robótica, tienen funciones que reciben datos y entregan información de sensores o controlan actuadores varias veces en un mismo código. La programación y la abstracción del programa se ve simplificada con la reducción de código repetitivo en una línea.

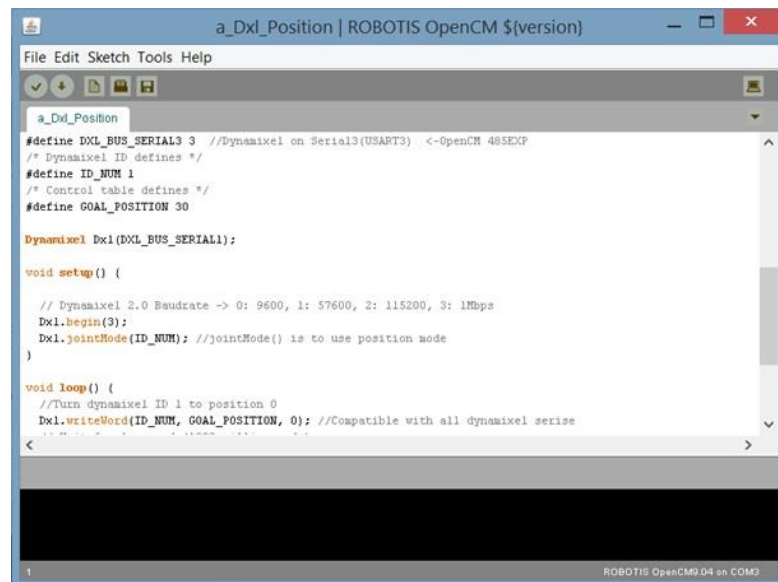


Figura 1.3 ROBOTIS OpenCM IDE [10]

1.5.7 Teleoperación de robots

La teleoperación de robots consiste en la actividad de controlar a los mismos a distancia por un usuario en una estación remota. El robot está controlado por medio de un programa interno con el objeto de realizar tareas propias y tener mayor autonomía. Generalmente, este programa se encuentra ubicado en la memoria de un sistema de procesamiento de datos como un microcontrolador o un DSP (procesador digital de señales). Los microcontroladores, como es la opción escogida para el presente

proyecto, son computadores compactos que poseen características como generar señales fácilmente manipulables.

Una estación de teleoperación común consiste en un computador y, en este proyecto, de dispositivos de entrada y de salida para interactuar a distancia. Los dispositivos de entrada son cualquiera que permita introducir información o instrucciones para el control del robot, como un teclado o una palanca de control. Los de salida, pueden ser un monitor y los parlantes que transmitan información al usuario. [11]

Este término se introduce en la robótica para destacar que la operación de los robots móviles tendrá asistencia de inteligencia humana. Este proceso en el que el usuario podrá recibir información desde un robot pueda tomar decisiones y generar órdenes para su ejecución automáticamente permite una intervención más efectiva al disminuir retardos de procesamiento de información y evaluación de casos.



Figura 1.4 Robot teleoperado [12]

1.5.8 Robot Paralelo

Un robot paralelo consiste en una plataforma móvil y una fija unidas por varias cadenas cinemáticas formando un mecanismo cerrado. Estas cadenas cinemáticas son independientes entre sí. Se caracterizan por ser más veloces y precisos que un robot de cadena abierta, como un brazo robótico. Los elementos estructurales actúan de manera simultánea, pudiendo distribuir la carga soportada sobre todos ellos. [13]

1.5.9 IMU

Una unidad de medición inercial o IMU (Inertial Measurement Unit en inglés) corresponde a un dispositivo electrónico que realiza mediciones de velocidad, fuerza gravitacional, orientación, y hasta fuerza y dirección de un campo magnético. Esto es posible gracias a la incorporación de acelerómetros, giróscopos y magnetómetros. Por lo general, estas unidades de medición inercial son usadas para maniobrar desde vehículos aéreos tripulados o no, hasta naves espaciales.

Una IMU es un componente esencial de sistemas de navegación inercial usados en artefactos y vehículos guiados realizando un control de posición y orientación. Para estas aplicaciones, los datos recolectados por los sensores de una IMU permiten a un computador corregir errores y contribuir a la toma de decisiones en caso de fallas. [14]

Por otro lado, es importante saber que para conocer totalmente la posición y orientación de un cuerpo se deben conocer 3 valores de rotaciones y 3 valores de traslaciones geométricas respecto a un sistema coordinado de 3 dimensiones. Un cuerpo aislado puede desplazarse libremente en un movimiento que se puede descomponer independientes en estos 6 valores. Estos 6 movimientos independientes se consideran grados de libertad. Para que una IMU pueda realizar la obtención de estos 6 valores, debe medir 9 valores (aceleraciones, orientaciones y fuerzas magnéticas) para cada eje del sistema de referencia. Así, para aplicaciones donde se requiere saber posición y orientación a detalle se utiliza una IMU denominada de 9 grados de libertad.



Figura 1.5 IMU de 9 grados de libertad con sensores integrados [15]

1.5.10 Ejes principales de navegación

Un objeto en el espacio puede girar libremente según un sistema coordinado en tres dimensiones con respecto a tres ejes. Estos ejes se denominan como vertical, transversal y longitudinal en ámbitos de vuelo u otros similares. Para analizar al objeto se supone que estos ejes se mueven con el objeto y giran en relación con un sistema de referencia constante. Tomando como ejemplo un avión, se pueden distinguir estos tres tipos de giro: yaw, la nariz del vehículo gira hacia la derecha o izquierda desde una vista superior; pitch, la nariz se mueve hacia arriba o hacia abajo desde una vista lateral; y roll, la nariz se mueve hacia la derecha o izquierda desde una vista frontal. Estas definiciones se aplican de manera análoga desde mediados del siglo XX para la construcción de aviones, naves espaciales u otros vehículos de similar fin.

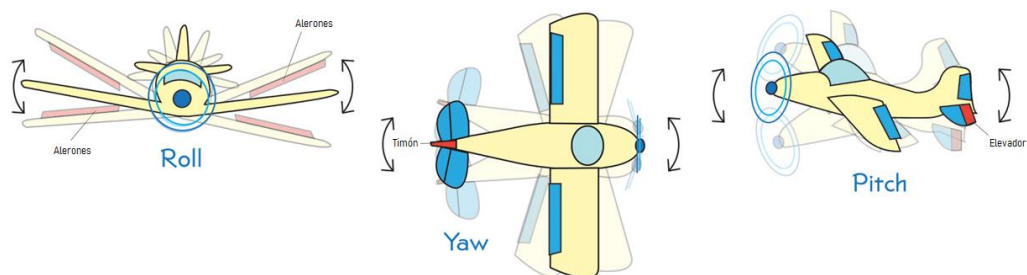


Figura 1.6 Ángulos de Navegación [16]

1.5.11 Cinemática Directa e Inversa de un robot

La cinemática en un robot estudia su movimiento con respecto a un sistema de referencia. La cinemática consiste en una búsqueda de la descripción analítica del movimiento del robot en el espacio con respecto al tiempo, y de la determinación de relaciones entre posición y orientación de este dependiendo de los valores asignados de coordenadas articulares.

La cinemática directa representa un problema que requiere una solución para encontrar el valor de la posición final de un robot. Esta solución es una función de los valores articulares, equivalente a los valores de

traslación o rotación de las articulaciones; estos son expresados muchas veces como ángulos de servomotores que comandan la estructura. Existen varios métodos para resolver este problema, uno de los primeros desarrollados y más rústicos en su interpretación frente al margen de posibles soluciones es la utilización de consideraciones geométricas como púnica forma de parametrizar la posición y orientación. Este método como otros, ofrece la ventaja de conocer tanto la posición final de cada segmento o parte física del robot. [17]

En general el modelo cinemático inverso, obtenido a partir del directo, es el más usual de utilizar en cálculos. La necesidad de posicionar una sección específica de un robot o alguna herramienta para realizar alguna tarea demanda tener la posición en el espacio en función de las rotaciones y translaciones de los elementos constitutivos.

Para abordar el problema de la cinemática inversa y el cálculo respectivo de los valores articulares necesario para posicionar un punto en el espacio referenciado al sistema de coordenadas global existen diferentes formas. Generalmente, se propone un conjunto de ecuaciones cerradas que, a través de relaciones matemáticas, determinan los puntos adecuados para hacer que el robot haga el seguimiento de trayectoria, proporcionando así una solución en tiempo real.

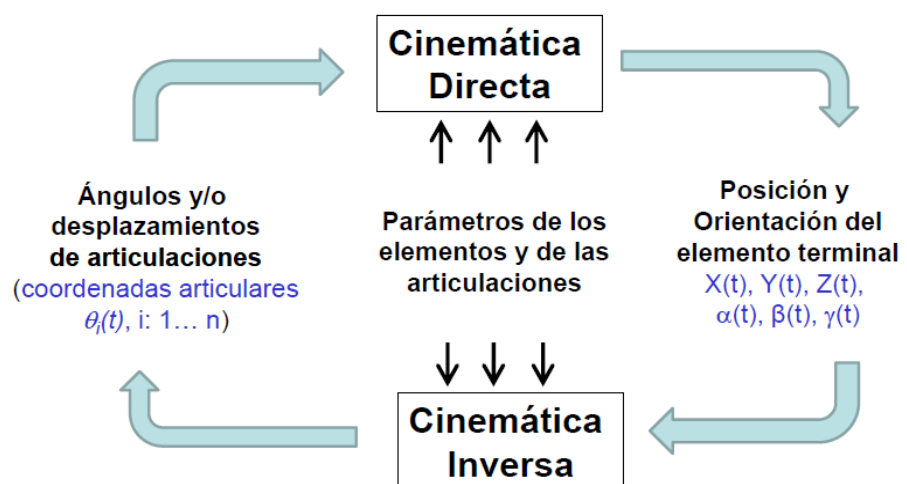


Figura 1.7 Cinemática de un robot [18]

CAPÍTULO 2

2. METODOLOGÍA

Frente al desafío que representa que niños con trastorno del espectro autista (TEA) mantengan su atención en herramientas educativas como aplicaciones móviles o similares, se escogió desarrollar por un equipo multidisciplinario una herramienta educativa que contribuya al desarrollo de sus habilidades sociales por medio de la imitación. Así, un robot social al presentar patrones de comportamiento sencillos y expresiones faciales fácilmente identificables presenta menor dificultad para interactuar y provocar, con el fin de mantener al paciente enfocado en una sola actividad.

Este proyecto al utilizar esta tecnología en un esfuerzo por ser efectivos en actividades de ayuda a niños con TEA tiene como beneficio a largo plazo aumentar la disposición de estos niños a interactuar con terapeutas humanos y disminuir la incomodidad durante las sesiones de la terapia.

En este capítulo se detallan todos los métodos y herramientas que se utilizaron para: transmitir y receptar instrucciones a un microcontrolador desde un servidor, programar un microcontrolador y realizar el diseño electrónico para la generación de señales de control de servomotores, y analizar el modelo cinemático directo e inverso del robot social.

La cabeza robótica del robot social desarrollada en el presente proyecto tiene como función principal la interacción con pacientes con TEA y que estos mantengan la atención en una sola actividad. A su vez, envía datos en tiempo real a personal especializado en su tratamiento para monitoreo de las reacciones a diferentes estímulos. Se utiliza una pantalla LCD de 8 pulgadas para mostrar expresiones faciales principales: felicidad, tristeza, enojo y entusiasmo. Para la interacción verbal, se utiliza un altavoz mediante el cual se emiten líneas de dialogo previamente establecidas. La cabeza robótica producirá movimientos de cabeceo mediante servo motores controlados por un microcontrolador.

2.1 Metodología de diseño

Durante la concepción, el desarrollo del presente proyecto y su implementación es necesario proveer como tendrán que ser los

procedimientos e idear los instrumentos adecuados a los objetivos preestablecidos. Debido a que se trata del desarrollo de un producto, se utilizó el proceso genérico de desarrollo, ver la tabla 2.1 donde se muestra la secuencia lógica. Sin embargo, al tratarse de un producto personalizado, este proceso tendrá ciertas variables. [19]

En el presente proyecto se presentan dos etapas generales. La primera etapa son tareas que permiten la ampliación y exploración de posibilidades para el desarrollo del proyecto procurando respetar los límites previamente establecidos. En la segunda etapa es necesaria la concreción o síntesis del proyecto.

Análisis			Síntesis		
Planeación	Conceptualización	Diseño esquemático	Desarrollo fabril	Pruebas	Lanzamiento
Construcción y validación del documento informativo del producto	Definición de conceptos y estrategias Prediseño	Desarrollo formal del prototipo	Especificación, desarrollo técnico y producción	Testeo, análisis de resultados y ajustes	Presentación del producto

Tabla 2.1 Metodología de diseño.

2.2 Esquema de los módulos del proyecto

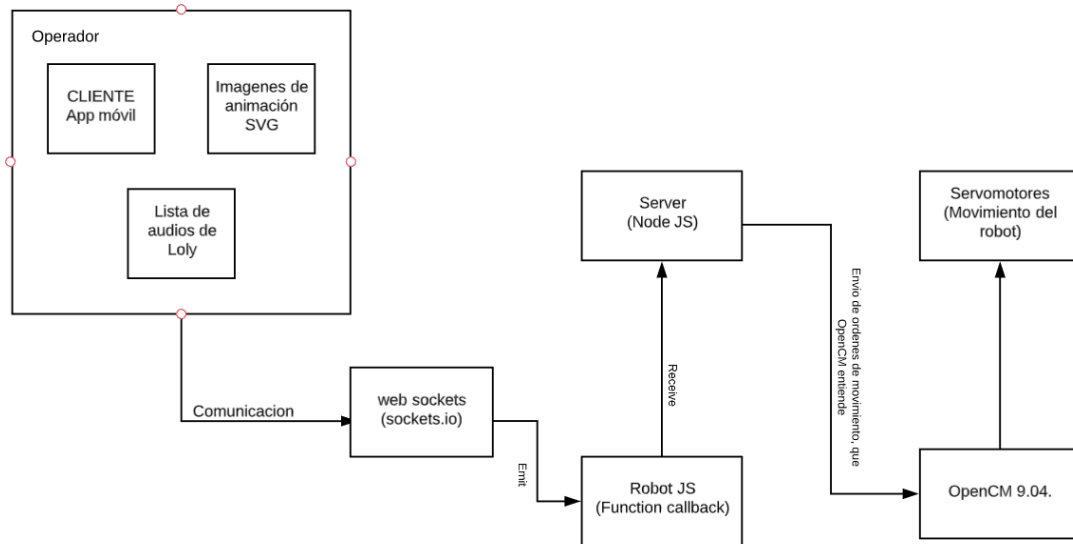


Figura 2.1 Esquema general del proyecto [20]

En la presente imagen se muestra un esquema del proyecto general, se realizó una segmentación de bloques en los que se separan varias tecnologías que se conectan de forma sincronizada durante todo el desarrollo de las funciones del robot, mismas que se realizan en paralelo y tiempo real.

2.3 Módulo Electrónico de control de movimiento

El módulo electrónico destinado al control de movimiento del robot social a través de 6 servomotores está conformado por los submódulos: placa de desarrollo y actuadores.

2.3.1 Submódulo placa de desarrollo

La placa de desarrollo es el sistema embebido a utilizar para la recolección de datos, recepción y transmisión de información para el control de los movimientos de la cabeza robótica. Con el fin de desarrollar un proyecto asequible y de bajo costo se seleccionó un sistema embebido de código abierto.

Las opciones para considerar en la adquisición de datos y mando de este proyecto fueron las diferentes versiones de placa OpenCM9.04.

Placa	OpenCM9.04 Tipo A	OpenCM9.04 Tipo B	OpenCM9.04 Tipo C
Power Switch	X	1	1
User Button	1	1	1
Pin Battery(LBS-40)	X	2	2
JTAG/SWD	X	1	1
Micro B USB	X	1	1
5-Pin Port	X	4	4
DYNAMIXEL TTL BUS	X	4	4
4-Pin Communication	X	1	1

Tabla 2.2 Diferencias principales entre los tipos de OpenCM9.04

Para la elección del dispositivo que constituirá el sistema de adquisición y mando se consideraron estas opciones, principalmente por su versatilidad para la comunicación y puertos para servomotores Dynamixel sin necesidad de una interfaz electrónica adicional.

Los criterios para la selección entre las mismas fueron: el costo, seguridad y capacidad de transmisión de datos. Adicionalmente, para todas las opciones el software de programación consta de la IDE de Arduino; por lo que el elemento diferenciador será el hardware.

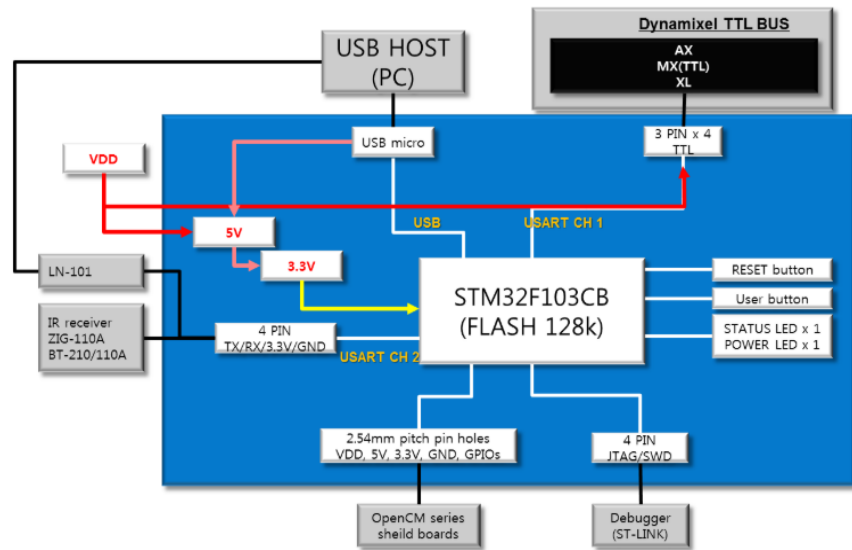


Figura 2.2 Diagrama de bloques de OpenCM9.04. [21]

Entre la comparación de las 3 versiones, la OpenCM9.04 Tipo A quedó rápidamente en desventaja al carecer de puertos TTL necesarios para el control de los servomotores a utilizarse. Entre las dos opciones restantes, la OpenCM9.04 Tipo C se quedó atrás al limitarse sus puertos a un modelo específico de servomotores, los de la serie XL-320. Es así como la OpenCM9.04 Tipo B fue la opción elegida por contar con puertos capaces de controlar servomotores Dynamixel de las series XL-320 y AX/MX.



[OpenCM9.04 A-Type]



[OpenCM9.04 B-Type]



[OpenCM9.04 C-Type]

Figura 2.3 Tipos de placas OpenCM9.04. [22]

2.3.2 Submódulo de actuadores

La función de los actuadores de movimiento es la generación de energía rotacional a los componentes de la cabeza robótica.

Dimensionamiento de servomotor

Para dimensionar los seis servomotores a utilizar, se procede a receptar el diseño utilizando el software SolidWorks, el cual contiene las propiedades físicas de cada pieza.

Para determinar el torque experimentado T se tomará en cuenta los siguientes factores:

$FactSeg$: Factor de seguridad para el mecanismo.

η : Rendimiento del motor

$Pérdidas$: Perdidas por transmisión.

Tt : Torque teórico necesario

La relación entre estos parámetros y el Torque está dada por la ecuación:

$$T = \frac{FactSeg}{\eta * Pérdidas} * Tt \quad (2.1)$$

Donde:

$$FactSeg = 2$$

$$\eta = 0,8$$

$$Pérdidas = 0,5$$

Entonces:

$$T = 5 * Tt$$

Cálculo de torque de servomotores para movilidad de las alas

Con uso del software SolidWorks se saca el centro de masa con respecto al eje de movimiento del brazo. Este cálculo sirve tanto para el ala izquierda como para el ala derecho.

$$Tt = Peso * Distancia \text{ al eje de giro} \quad (2.2)$$

Donde:

$$Peso = 1.238 [N]$$

$$\text{Distancia al eje de giro} = 7.52 \text{ [cm]}$$

Entonces, según las ecuaciones (2.1) y (2.2)

$$T_t = 9.31 \text{ [N * m]}$$

$$T = 46.6 \text{ [N * m]}$$

Cálculo de torque de servomotores para movilidad de la cabeza

Con el uso del software SolidWorks se obtiene el centro de masa con respecto al eje de movimiento, característica determinada en tres dimensiones (expresado como ejes coordenados: x, y, z) que sirve para obtener la distancia al eje de giro que determina el torque aplicado. Una vez simulados los servomotores con la estructura predeterminada, el software entrega el torque requerido en la práctica.

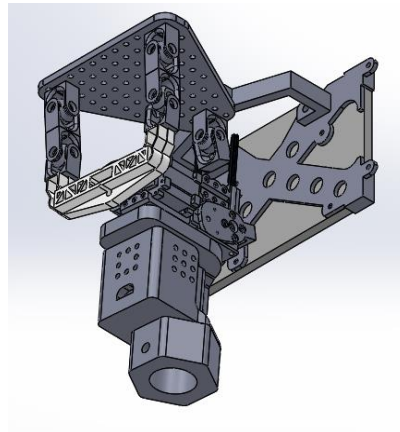


Figura 2.4 Vista Frontal oblicua de la cabeza del Robot Social

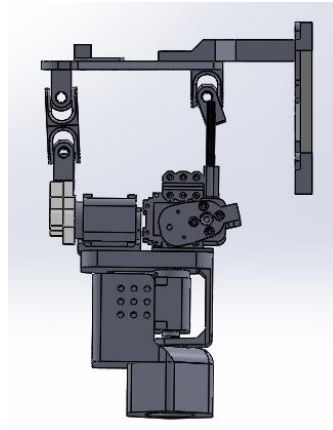


Figura 2.5 Vista Lateral de la cabeza del Robot Social

Elección de servomotor

Se escogió al actuador robótico DYNAMIXEL AX-12A por ser un componente de robot sofisticado. Este servomotor destaca por contar con un microcontrolador que tiene la capacidad de entender 50 comandos para los cuales se fijan o leen parámetros que definen su comportamiento. Cada uno de los 6 servomotores utilizados para este proyecto tiene la capacidad de controlar su velocidad, temperatura, posición, tensión, entre otros parámetros. El algoritmo que controla la posición de cada uno puede modificarse independientemente, esto permite el control con retroalimentación de velocidad y carga soportada por los actuadores. Destacan instrucciones de control sobre el torque aplicado, informe de su estado y capacidad de carga que supera lo requerido, que lo convierten en la mejor opción para este proyecto. La cantidad de tensión aplicada al servomotor y su velocidad son programables para ser proporcional a la distancia que éste necesita viajar. Así, si el eje necesita regresar una distancia grande, el motor regresará a toda velocidad. Si este necesita regresar sólo una pequeña cantidad, el motor correrá a una velocidad más lenta.



Figura 2.6 Dynamixel AX-12A [23]

Por otro lado, se seleccionó el servomotor Dynamixel XL320 como otro actuador para el desarrollo del proyecto, su bajo tamaño y peso de solo 16.7gr lo hacen idóneo para realizar movimientos repetitivos de piezas pequeñas, cuenta con una velocidad de 114rpm trabajando a 7.4V y la cantidad de tensión aplicada al servomotor y su velocidad son programables, permitiendo realizar un control de velocidad eficiente. Sin embargo es importante recalcar que, para la alimentación energética de este actuador, es necesario realizar un proceso de reingeniería en la

placa OpenCM, debido a que los niveles de voltajes que necesita el Dynamixel XL320 no es proporcionado de manera directa, por tanto, se coloca un regulador de voltaje, para permitir una correcta alimentación del servomotor.



Figura 2.7 Dynamixel XL-320 [24]

2.4 Obtención de funciones de la Cinemática Directa y la Cinemática Inversa

Para abordar el problema cinemático directo existen varios métodos para determinar la localización de las secciones del robot. Es importante notar que para cualquiera se deben conocer los parámetros geométricos del robot. Destacan dos tipos de métodos conocidos por su facilidad en la robótica: métodos geométricos y métodos basados en cambios de sistema de referencia.

Los métodos geométricos son métodos no sistemáticos y más complejos al aumentar los grados de libertad. Utilizan relaciones geométricas para obtener directamente la posición en función de las variables articulares. Requieren visión e intuición espacial de quien realiza el análisis. En el proyecto se optó por un método geométrico, en vez de uno basado en cambios de sistemas de referencia. Aunque estos métodos son sistemáticos y adoptan convenciones que se repiten en cada cambio, las matrices de transformación resultan redundantes con ecuaciones por métodos más simples. Por lo general, en robots con pocos grados de libertad, en las ecuaciones se tienden a eliminar una gran cantidad de términos; resultando en grandes expresiones que se reducen una vez realizadas todas las

multiplicaciones matriciales. Así, cuando se comparó la utilización de matrices de transformación homogénea contra un análisis de desplazamientos en los tres ejes coordenados, se destacó que tiene un mayor cálculo y eliminación de términos con la misma cantidad de ecuaciones. Finalmente, la cinemática inversa se obtuvo por el despeje de las variables articulares en función de las variables de orientación. Las variables de posición (los 3 valores de desplazamientos lineales) se dejaron en función de las de orientación (los 3 ángulos de referencia con respecto a cada eje coordenado), por ser un robot con 3 grados de libertad que tiene limitado el control de sus parámetros en el espacio.

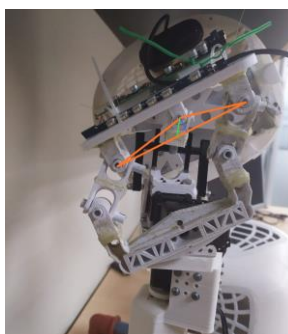


Figura 2.8 Análisis geométrico de la cabeza robótica

2.5 Programación de la placa OpenCM9.04

Para la programación de la placa de desarrollo, se tomó en consideración la comunicación con un CPU que funciona como emisor de las posiciones. Al no poder predecir cuándo llegará una nueva instrucción y tener el requerimiento que estas deban ejecutarse lo más rápido posible, se configuró la recepción de posiciones como interrupciones. Esta programación ejecutará lo más rápido posible el movimiento, para que el programa pueda volver donde se quedó en su secuencia lógica.

Destaca el uso de la librería Dynamixel para el control de posición y velocidad de los servomotores. Con esta librería es posible optimizar el código en una versión más limpia, donde con pocas funciones se puede leer parámetros y mover varios tipos de servomotores a diferentes velocidades.

El código empieza definiendo valores constantes a valores de voltaje, temperatura, límites de ángulos, límites de torques y comandos de

operación. Posteriormente se realiza la definición de los diferentes tipos de variables a utilizarse a lo largo del desarrollo del código.

Utilizando una función de “setup” se configuran las secuencias lógicas que llevará a cabo el microcontrolador, se establecen algunos criterios que requieren una ejecución única, protocolos a utilizar por los servomotores Dynamixel AX12-A y Dynamixel XL30 respectivamente. Se utiliza el comando “serial.begin” para indicar en el programa que se da inicio a la comunicación serial. Con el uso del comando “pinmode” se indica al microcontrolador que pin físico funcionará como entrada análoga.

Se procede a establecer los distintos casos de posición de los seis servomotores: tres que se encargan el movimiento de la cabeza, dos del movimiento de las alas y uno del movimiento del pico; y los casos que comandan sus respectivos modos de operación.

Finalmente, utilizamos la función “loop” para ejecutar un número infinito de veces la lectura de los sensores, ya que necesitamos tener presente estos valores a lo largo de toda la ejecución por motivos de seguridad. Se supervisa temperatura y cualquier otra alerta presente en los servomotores.

La placa recibe una trama que indica los movimientos que va a realizar cada servomotor, esta trama se obtiene utilizando una IMU para la adquisición de datos de orientación del movimiento de la cabeza de una persona. Con un procesamiento matemático de los datos obtenidos, se representan los movimientos a realizarse en porcentajes. La mencionada trama tiene la siguiente estructura: “0x02;81;50;50;100;100;75;25;100;32; 57;92;60;0.1”.

Donde la primera columna es una variable identificadora denominada POSI declarada previamente, las siguientes tres columnas indican en porcentajes las posiciones que van a tomar cada uno de los tres servomotores que comandan la cabeza. La quinta y sexta columna comandan las posiciones de las alas, mientras que la séptima columna se establecen los movimientos que debe cumplir el pico. Finalmente, las seis últimas columnas indican las velocidades a las que se mueven los servomotores con el mismo orden previo de declaración de valores.

2.6 Obtención de posición y orientación con IMU de movimientos de una cabeza humana

Para los movimientos del robot social se decidió emular movimientos de una cabeza humana. Para la obtención de información de estos movimientos, como asentar o negar, se obtuvieron listas a partir de mediciones de posición y orientación mientras una persona realizaba la acción. Las 6 listas obtenidas corresponden a los datos de 3 desplazamientos lineales y 3 ángulos medidas en un intervalo de tiempo constante. Estos datos recolectados con una IMU de 9 grados de libertad se miden con respecto a los 3 ejes del sistema coordenado de 3 dimensiones. La información de estas 6 listas es enviada desde el microcontrolador en la IMU en un computador para su procesamiento y futura conversión a valores articulares.

2.7 Aplicación de cinemática inversa para la obtención de grupos de valores articulares por cada movimiento

Debido a que este robot no puede controlar los 6 parámetros de posición y orientación, se optó por solamente controlar la orientación y dejar la posición del mismo dependiente de ella. A través de la cinemática inversa se planeó obtener los valores de las posiciones articulares para lograr orientaciones específicas. Es importante destacar que la cinemática inversa fue obtenida para reemplazar en las ecuaciones desarrolladas los 3 valores de rotaciones y los 3 valores de traslaciones geométricas, obtenidos de una IMU. Así, los movimientos de la cabeza robótica que emulan la de una humana, pueden estar en función de posiciones articulares controladas por el OpenCM9.04 y los servomotores.

Para la obtención de las listas de velocidades que debe recibir el OpenCM9.04 para controlar los servomotores, se desarrollaron expresiones de estos 6 parámetros en función de la velocidad y el tiempo con la que se desea cada movimiento. Se tuvo que derivar en función de dichos parámetros las 3 ecuaciones de la cinemática inversa.

2.8 Concepto de diseño

Loly el robot interactivo, está diseñado con base en dos estructuras, una que engloba todas las partes mecánicas del robot. Esta estructura siempre se diseña dándole el mayor nivel de relevancia a la eficiencia del sistema, puesto que esta estructura el paciente no lo podrá observar.

La estructura en la que el diseño gráfico es más importante es en la externa, puesto que es la cobertura de robot, este armazón debe tener una forma amigable que tenga un impacto positivo en los pacientes.



Figura 2.9 Representación gráfica de Loly [25]

El diseño del sistema de control del robot cuenta también con un servidor local, el cual permite la emisión de ordenes desde una Tablet conectada a la misma red de wifi, este control consta de movimientos de cabeza, pico y alas, proyección de animaciones en la panta y transmisión de audio.

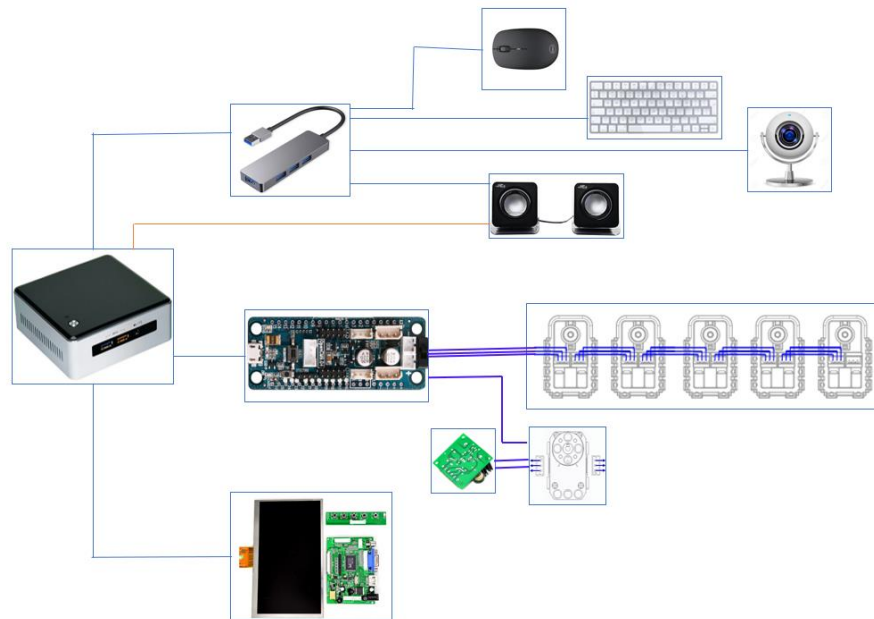


Figura 2.10 Diagrama de conexión del robot

CAPÍTULO 3

3. ANÁLISIS DE RESULTADOS

En el presente capítulo se aborda el análisis de resultados obtenidos a partir del desarrollo de cada módulo del proyecto, para garantizar que se cumple todos los requerimientos planteados al inicio. Se realizaron pruebas de precisión de los servomotores, pruebas de comunicación de envío de datos que comandan los movimientos de servomotores, pruebas de consumo de corriente, análisis de cinemática directa e inversa utilizando datos reales de posición.

3.1 Precisión en el servomotor Dynamixel AX-12A

El análisis de precisión del servomotor se realiza, representando físicamente un ángulo deseado, para realizar unas comparaciones entre el ángulo deseado y el ángulo proporcionado por el servomotor. Para el servomotor que no corresponde a este modelo se aplica la misma analogía basada a su similitud cinemática. Para estas pruebas se repiten varias iteraciones con el mismo ángulo variando factores externos como el peso que debe mover el servomotor y otras pruebas se realizan cambiando bruscamente la posición final que se debe obtener. Se seleccionan ángulos de diferentes cuadrantes dependiendo de la referencia preestablecida, en este caso estas variaciones recorren diferentes rangos, dependiendo del espacio de trabajo de la cabeza robótica, cabe recalcar que aunque los grados se establecen de manera aleatoria, es necesario establecer y respetar los límites de cada servomotor.

Servo	Límite mínimo	Límite máximo
Θ_2	-11.71°	11.71°
Θ_3	10.31°	71.83°
Θ_1	-20.50°	20.50°

Tabla 3.1 Límites de barridos de los servomotores de la cabeza robótica

Movimiento	Ángulo	Evaluación	Estado
------------	--------	------------	--------

Yaw	-10°		Listo
	-5°		Listo
	-1°		Listo
	5°		Listo
	11°	✗	Calibrar
Pitch	12°	✗	Calibrar
	20°		Listo
	30°		Listo
	50°		Listo
	70°	✗	Calibrar
Roll	-10°		Listo
	5°		Listo
	15°		Listo
	18°		Listo
	20°	✗	Calibrar

Tabla 3.2 Prueba de Ángulos deseados Vs. Ángulos obtenidos

En la tabla 3.2 se muestran los resultados de las pruebas de los diferentes ángulos en los diferentes servomotores, se concluyó que existía un conflicto en la calibración de los límites de cada servomotor, después de realizar los cambios necesarios se obtuvo un resultado totalmente satisfactorio en todos los ángulos, siempre y cuando este se encuentre dentro de los límites previamente establecidos en la tabla 3.1.

3.2 Análisis de Carga para Dynamixel AX-12A

Para el análisis de carga, es necesario realizar el cálculo de consumo de la corriente y potencia de cada servomotor implementado en el prototipo, se considera el consumo de cada servomotor establecido por el fabricante dependiendo de su modelo, datos que se muestran de manera explícita en la tabla 3.3.

Elemento	Voltaje (V)	Corriente (A)	Potencia (W)
5 servomotores Dynamixel Ax-12A	12	5.4	64.8
1 Servomotor Dynamixel XL-320	7.4	0,8	5.92
OpenCM Robotis	5	0,04	0,20
Buzzer	5	0,02	0,1
TOTAL		6.26	71.02

Tabla 3.3 Consumo de potencia del prototipo a plena carga

Utilizando el análisis de consumo de cada elemento mostrado en la tabla 2.4, se obtiene como resultado que el consumo total del sistema es de 71.02[W] de potencia y 6.26[A] de corriente. Basado en estos requerimientos para la alimentación del prototipo, se selecciona una fuente de alimentación seccionada, dependiendo de las diferentes partes del robot, para el control de servomotores la placa proporciona la alimentación a los diferentes servomotores y se realiza una modificación, para obtener los diferentes niveles de voltajes necesarios.

En la tabla 3.3 se puede observar, que los requerimientos de voltaje de los dos modelos de servomotores son diferentes, sin embargo, ambos deben ser alimentados por la placa OpenCM 9.04. Se adecuo la salida de alimentación de voltaje de 12[V] que necesita el AX-12A, para la alimentación del servo XL-320. Con una placa que regule ese voltaje positivo y un potenciómetro para la calibración en una placa se obtuvo el voltaje meta de 7.4[V].

3.3 Comunicación para la teleoperación

Para la integración del proyecto, algunos resultados de programación del software de teleoperación son relevantes para el control del robot. Entre ellos, la comunicación para la teleoperación. En el desarrollo de las actividades de este robot social interactivo, es de alta relevancia la teleoperación, ya que los niños con TEA presentan actitudes diferentes cuando tienen que manejar una interacción social con otra persona que con un robot. Así, se realizan pruebas de control de movimiento mediante la

interfaz que maneja un especialista a una distancia prudente en su opinión del robot.

Para esto la comunicación en tiempo real es indispensable. La herramienta empleada consiste en WebRTC, una que cuenta con estándares de comunicación en tiempo real y su funcionamiento tiene como base la comunicación entre pares. Estos son ordenadores que se comportan como iguales entre sí. Ésta utiliza un servidor de conferencias web trabajando simultáneamente con un servidor STUN, que permite una sincronización de las conexiones entre los diferentes nodos incluidos en la comunicación de los dispositivos que operan en el proyecto.

A continuación, se detallan los resultados de las pruebas realizadas de la transmisión, que tiene como objetivo la verificación de los componentes físicos y de los módulos que se conectan al servidor.

Fecha	Progreso	Resultado
26/06/2019	Desarrollo local de transmisión RTC	Se logró hacer transmisión web RTC local, pero la interfaz aún estaba rustica y se necesitaban enviar identificadores de un navegador a otro para hacer pruebas.
17/07/2019	Finalización de récord RTC local	Se logró acoplar con lo desarrollado previamente en WebRTC y hacer una grabación.
29/07/2019	Inclusión de código local en el robot interactivo.	Se arreglaron errores para que funcionen con varios "romos". Se pudo hacer una inclusión y que funcionara correctamente en el robot.

Tabla 3.4 Tabla de resultados de pruebas de transmisión

Actualmente se realiza el análisis de los tiempos de respuestas desde que se emite una señal desde la aplicación móvil, al presionar algún botón con

la respuesta del servidor. Al recibir esta señal el servidor, ejecuta de manera paralela la programación de los movimientos, audio y proyección en el display del robot. Los resultados de estas pruebas se muestran en la tabla a continuación.

	RESPUESTA	EVALUACIÓN
Ejecución de los movimientos	Inmediata	✓
Proyección en el display del robot	Inmediata	✓
Transmisión de audio	Inmediata	✓

Tabla 3.5 Pruebas de Comunicación

3.4 Análisis de transmisión de audio y video

En el presente proyecto, es necesario establecer la comunicación entre una cámara ubicada en la cabeza del robot y una computadora a la que un operador tiene acceso. La función de la cámara es enviar al especialista la grabación del comportamiento del paciente mientras utiliza la aplicación.

Para la transmisión y grabación del video se utilizan las herramientas web RTC y recordRTC respectivamente, cabe recalcar que en este software es necesario declarar variables que contienen atributos. Se establecen dos: uno utilizado para la transmisión del audio y otro para la transmisión del video. Posteriormente se utiliza la herramienta getUserMedia, encargada de crear un video de tipo media Stream. El usuario tiene la libertad de otorgar o rechazar el uso del dispositivo de entrada video/audio.

Para guardar los videos en un registro permanente, se hace uso de la herramienta recordRTC. Se implementaron eventos de grabar y detener, con el fin de que el especialista en caso de considerar necesario pueda registrar archivos solo de eventos puntuales.

Este proceso, cuenta con una transmisión en tiempo real permitiendo al especialista una respuesta rápida ante los requerimientos del paciente. Los archivos de grabación se adquieren sin problemas palpables, y con imágenes claras a simple vista que permiten el análisis de los gestos faciales del paciente.

3.5 Programación del OpenCM 9.04.

La programación que comanda el movimiento de la estructura móvil del robot didáctico se encuentra alojado en la placa OpenCM 9.04, el código utiliza Arduino como lenguaje de programación.

Los resultados de esta programación se analizaron con el ingreso de 22 rutinas de movimiento diferentes, entre movimientos de alas, cabeza y pico simultáneamente; y en ocasiones sólo uno de los elementos. Obteniendo un éxito de 90% en la primera prueba y luego del proceso de corrección se llegó a una precisión del 98%.

Cabe recalcar que, en estas pruebas también se analizó la obtención de datos de orientación de una IMU y el procesamiento matemático de los datos. Todo con el fin de convertirlos en la trama de comunicación con que la placa OpenCM se programó para recibir. Uno de los desafíos a enfrentar fue la sincronización de sonido, movimiento y audio sin un guión previamente elaborado para el manejo común de tiempos, y que se realizó por movimientos en intervalos contantes de tiempo con igual duración de cada audio.

3.6 Análisis de la cinemática directa e inversa

En el presente proyecto, para una estructura cerrada y una base móvil donde su movimiento depende de la posición de tres servomotores, se obtuvieron expresiones que relacionen sus variables articulares con sus variables de posición y orientación en el espacio. Para ello, primero fue necesario expresar sus desplazamientos lineales para cada eje coordinado en función de las variables articulares. Algunos de los cambios de variables, dependen de ángulos de desfase que hacen que sus eslabones no sean paralelos a los ejes coordinados.

$$q_1 = \theta_r - \theta_1 \quad (3.1)$$

$$q_2 = \theta_r + \theta_1 \quad (3.2)$$

$$m_{11} = m_1 - \theta_1 \quad (3.3)$$

$$m_{12} = m_2 - \theta_1 \quad (3.4)$$

El sistema de ecuaciones se obtuvo a partir del análisis de desplazamientos en la estructura cerrada en su forma minimizada, que al tener como punto inicial y final el mismo punto tienen como resultante un desplazamiento nulo. Un valor p servirá como referencia para las proyecciones finales de la base móvil al sistema de referencial general. Esta magnitud es la distancia del elemento terminal en un plano paralelo al de la base del robot. Por otro lado, una variable de interés para la obtención de una variable articular de uno de los tres servos es el resultado de un convertidor lineal. Este convertidor es un sistema móvil de eslabones dependiente de la posición de un servo, donde su posición se define como θ_3 y la altura desde la base del robot como l . La expresión final de esta altura se obtiene por un despeje mediante la ley del coseno, la conformación triangular de tres eslabones tiene dimensiones variables.

$$0 = D * \cos q_1 - h * \sin m_{11} * \cos n_1 - H * \cos m_3 \quad (3.5)$$

$$0 = -D * \cos q_2 - h * \sin m_{12} * \cos n_2 + H * \cos m_3 \quad (3.6)$$

$$y = D * \sin q_1 + h * \cos m_{11} * \cos n_1 + H * \sin m_3 * \cos n_3 \quad (3.7)$$

$$y = D * \sin q_2 + h * \cos m_{12} * \cos n_2 - H * \sin m_3 * \cos n_3 \quad (3.8)$$

$$p = \frac{d}{2} - h * \sin n_1 + H * \sin n_3 * \sin m_3 \quad (3.9)$$

$$p = \frac{d}{2} + h * \sin n_2 - H * \sin n_3 * \sin m_3 \quad (3.10)$$

$$p = d * \cos n_3 - \frac{d}{2} \quad (3.11)$$

$$l = l_o + p c_2 * \cos(\theta_3) + \sqrt{p c_1^2 - (p c_2 * \sin(\theta_3))^2} \quad (3.12)$$

Con el fin de relacionar variables de orientación de la base móvil del robot y las variables articulares, se empleó un método geométrico de distancia euclidianas de un eslabón móvil (3.13). Luego de varias minimizaciones para la aplicación en dos eslabones móviles según el sistema coordinado general, se obtuvieron las siguientes expresiones (3.14) y (3.15):

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} = h \quad (3.13)$$

$$\begin{aligned} &(-H * \cos m_3 - (-D * \cos q_1))^2 + (l - d * \sin n_3 + H * \sin m_3 * \cos n_3 - D * \sin q_1)^2 \\ &+ (d * \cos n_3 - \frac{d}{2} + H * \sin n_3 * \sin m_3 - \frac{d}{2})^2 = h^2 \end{aligned} \quad (3.14)$$

$$\begin{aligned}
& (H * \cos m_3 - D * \cos q_2)^2 + (l - d * \sin n_3 - H * \sin m_3 * \cos n_3 - D * \sin q_2)^2 \\
& + (d * \cos n_3 - \frac{d}{2} - H * \sin n_3 * \sin m_3 - \frac{d}{2})^2 = h^2
\end{aligned} \tag{3.15}$$

Gracias, al desarrollo de programas de resolución de sistemas de ecuaciones no lineales por métodos iterativos, como el método de bisección, se obtienen los valores aproximados de dos incógnitas para las ecuaciones (3.14) y (3.15). Para la resolución del problema cinemático directo, las variables por calcular para encontrar las 6 variables de interés (3 ángulos de orientación y 3 desplazamientos lineales) según los ejes coordenados corresponden a los ángulos de navegación de pitch y roll. Por otro lado, para la resolución del problema cinemático inverso, las variables de interés corresponden a las variables articulares de los servomotores. Así, los valores por encontrar del sistema corresponderían a θ_1 y θ_2 . Se puede notar que tanto para la cinemática directa como la inversa el valor de θ_3 y el del ángulo de navegación yaw no son desconocidas. Esto se debe a que existe una equivalencia directa entre ambos porque el sistema de referencia fue situado en el punto de análisis de este servomotor.

$$x = p * \sin \theta_2 \tag{3.16}$$

$$y = l - d * \sin n_3 \tag{3.17}$$

$$z = p * \cos \theta_2 \tag{3.18}$$

$$q_{yaw} = \theta_2 \tag{3.19}$$

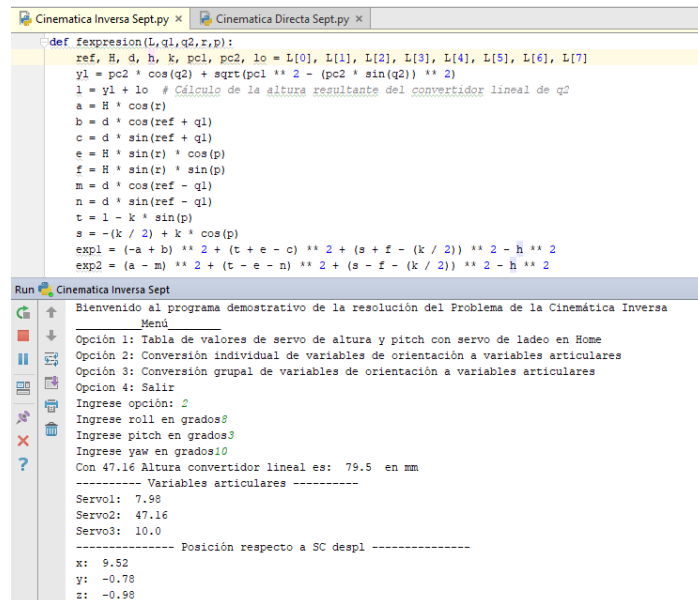
$$q_{pitch} = n_3 \tag{3.20}$$

$$q_{roll} = m_3 \tag{3.21}$$

3.6.1 Programación para la resolución de sistemas de ecuaciones no lineales

Para el problema que significa resolver sistemas no lineales se decidió utilizar el método de bisección y de iteración progresiva en un intervalo cerrado para el cálculo de dos variables. El lenguaje de programación escogido fue Python, y la interfaz Pycharm. Tanto para el problema cinemático directo como el inverso se puede observar que las variables desconocidas serán dos. Para el problema cinemático directo, θ_1 y θ_2 ; y para el inverso, los ángulos de navegación de pitch and roll. Todos los

demás parámetros de interés pueden ser obtenidos en función de las dos variables.



```

def fexpresion(L,q1,q2,r,p):
    ref, H, d, h, k, pc1, pc2, lo = L[0], L[1], L[2], L[3], L[4], L[5], L[6], L[7]
    y1 = pc2 * cos(q2) + sqrt(pc1 ** 2 - (pc2 * sin(q2)) ** 2)
    l = y1 + lo # Cálculo de la altura resultante del convertidor lineal de q2
    a = H * cos(r)
    b = d * cos(ref + q1)
    c = d * sin(ref + q1)
    e = H * sin(r) * cos(p)
    f = H * sin(r) * sin(p)
    m = d * cos(ref - q1)
    n = d * sin(ref - q1)
    t = 1 - k * sin(p)
    s = -(k / 2) + k * cos(p)
    expl = (-a + b) ** 2 + (t + e - c) ** 2 + (s + f - (k / 2)) ** 2 - h ** 2
    exp2 = (a - m) ** 2 + (t - e - n) ** 2 + (s - f - (k / 2)) ** 2 - h ** 2

```

Run Cinematica Inversa Sept

```

Bienvenido al programa demostrativo de la resolución del Problema de la Cinemática Inversa
Menú
Opción 1: Tabla de valores de servo de altura y pitch con servo de lado en Home
Opción 2: Conversión individual de variables de orientación a variables articulares
Opción 3: Conversión grupal de variables de orientación a variables articulares
Opción 4: Salir
Ingrese opción: 2
Ingrese roll en grados8
Ingrese pitch en grados3
Ingrese yaw en grados10
Con 47.16 Altura convertidor lineal es: 79.5 en mm
----- Variables articulares -----
Servo1: 7.98
Servo2: 47.16
Servo3: 10.0
----- Posición respecto a SC despl -----
x: 9.52
y: -0.78
z: -0.98

```

Figura 3.1 Programa para la resolución del problema cinemático inverso [26]

CAPÍTULO 4

4. CONCLUSIONES Y RECOMENDACIONES

El presente trabajo consta del diseño e implementación de un robot interactivo que ayude al proceso de aprendizaje y tratamiento de los niños con TEA. Para ello, la programación de los equipos de control del robot es de gran importancia en la interacción con un software educativo y sus respuestas. Los tipos de movimientos de un robot pueden redirigir la atención en un software específico. Reemplazando los métodos de enseñanza actualmente utilizados, en donde una persona pretende impartir conocimientos a un niño con TEA. Evitando un entorno que no fue diseñado basado en sus necesidades, se obtiene mejoras en el aprendizaje del paciente, ya que ellos tienden a redirigir su atención y mostrar gran interés en tecnología en contraposición de otros seres humanos.

CONCLUSIONES

Se construyó un robot utilizando un diseño modular, la cual permite en caso de fallas reemplazar alguna unidad sin necesidad de realizar cambios en otro elemento que se encuentre operativo. Para lograr satisfacer los requisitos de diseño del robot interactivo se utilizan 6 módulos: control electrónico, estructura externa, estructura móvil, tele-operación, aplicación móvil y fuente de alimentación energética.

Se construyó el robot con tres grados de libertad controlables y se realizó el análisis de su cinemática inversa y directa. Este problema que constituye un desafío por ser un robot paralelo se abarcó por la resolución con técnicas de métodos geométricos aplicadas a sus partes constitutivas para obtención de posiciones y orientaciones en el espacio. Desarrollando así, un método original para la estructura singular inventada. Esta solución es utilizada junto al uso de una IMU realizar adquisición de datos de orientación del movimiento de una persona. Estos datos que son ingresados y procesados en forma de listas de movimientos en el robot en programas de resolución matemática permitiendo que el robot imite el movimiento del humano.

Se realizó la programación de la placa OpenCM 9.04 para la generación de señales de control de servomotores Dynamixel modelo AX-12 y XL320 exitosamente en un robot paralelo. Se logró comandar los movimientos del robot interactivo mediante comunicación serial, desde un interfaz humano-máquina ubicada a una distancia que evite que el paciente perciba que su entorno está siendo invadido.

RECOMENDACIONES

Para mejorar la percepción visual con los pacientes y por ende los resultados del tratamiento, en futuras versiones de Loly se debe generar un repositorio de expresiones más reales haciendo uso de programas de diseño gráfico e imitación de rostros.

Con el propósito de mejorar la apariencia natural del robot se debe realizar en futuros trabajos, un diseño incrementando grados de libertad en las alas del robot para simular extremidades con un mayor número de articulaciones. Esto incide en un vínculo de interacción más alto entre el paciente y el robot interactivo por un mayor grado de similitud del robot humanoide.

5. BIBLIOGRAFÍA

- [1] V. Perez y R. Mayta., *Actualidad y perspectivas de la robótica*, Lima, 2001.
- [2] A. Perez, A. Castro, F. Alonso, J. Carlos y M. Salichs, *Evolución de la robótica social y nuevas tendencias*, 2017.
- [3] W. Knight, «MIT Technology Review,» Opinno, 27 02 2017. [En línea]. Available: <http://www2.technologyreview.es/robotica/53239/darwin-el-robot-que-mejora-las-sesiones-de/>. [Último acceso: 12 06 2019].
- [4] A. Hervas, N. Balmaña y M. Salgado, *Los trastornos del espectro autista (TEA)*, *Pediatr integral* , 2017.
- [5] NIH, *Trastorno del espectro autista*, Maryland, 2016.
- [6] A. G. González, «Panamahitek,» 2 12 2016. [En línea]. Available: <http://panamahitek.com/que-es-y-como-funciona-un-servomotor/>. [Último acceso: 15 07 2019].
- [7] «Robotis e-Manual,» Robotis, 2017. [En línea]. Available: <http://emanual.robotis.com/docs/en/parts/controller/opencm904/#>. [Último acceso: 2019].
- [8] F. M. Abarca, «Wordpress,» 15 08 2017. [En línea]. Available: <https://aprendiendoarduino.wordpress.com/2017/06/18/ide-arduino-y-configuracion/>. [Último acceso: 18 07 2019].
- [9] A. Ollero, *Robotica, manipuladores y robots moviles*, MARCOMBO, 2001.
- [10] E. Oyana, T. Maeda, A. Agah y S. Tachi, *Robots for Telexistence and Telepresence*., Tokyo, 2014.
- [11] D. Ferrer., *Adquisición de datos IMU en un sistema embebido*, Valencia , 2015.
- [12] «Sparkfun,» 04 05 2010. [En línea]. Available: <https://www.sparkfun.com/products/retired/10736>. [Último acceso: 10 06 2019].
- [13] V. E. G. Reynoso y L. V. y J. Reinoso, *Diseño de un manipulador industrial para aplicaciones de limpieza en subestaciones eléctricas*, Monterrey: 2001.
- [14] J. Ramirez y A. Fonseca, *Modelamiento matemático de la cinemática directa e inversa de un robot*, Ingeniería solidaria , 2012.

- [15] K. Ulrich y S. Eppinger, *Diseño y desarrollo de productos*, Mc Graw Hill , 2012.
- [16] «How things fly,» Smithsonian national air and space mountain , 2017. [En línea]. Available: <https://howthingsfly.si.edu/flight-dynamics/roll-pitch-and-yaw>. [Último acceso: 2019].
- [17] M. H. & C. García, *Modelo Cinemático Directo*.
- [18] Autor, *Cinematica directa e inversa*, Guayaquil, 2019.
- [19] C. R. León, *Propuesta metodológica para*, Monterrey.
- [20] Autor, *Modulos del proyecto*, Guayaquil , 2019.
- [21] ROBOTIS, «Support robotis,» 2018. [En línea]. Available: <http://support.robotis.com/en/product/controller/opencm9.04.htm>. [Último acceso: 22 Julio 2019].
- [22] Robotis, «Support robotis,» 2018. [En línea]. Available: <http://support.robotis.com/en/product/controller/opencm9.04.htm>. [Último acceso: 20 Julio 2019].
- [23] ROBOTIS, *Dynamixel AX-12*, 2006.
- [24] R. e-Manual, *XL-320*, 2019.
- [25] A. Leon, *Representación gráfica de Loly*, Guayaquil, 2019.
- [26] S. Fuenzalida, *Resolucion cinemática inversa*, Guayaquil, 2019.

ANEXOS

Video demostrativo del funcionamiento final del robot interactivo vinculado a una aplicación móvil

<https://youtu.be/CR-WejStRbQ>