



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO
CARRERA DE INGENIERÍA ELECTRÓNICA**

**DESARROLLO DE UN BRAZO ROBÓTICO SERIAL PARA ASISTENCIA EN UNA
CLÍNICA ODONTOLÓGICA**

Trabajo de titulación previo a la obtención del
Título de Ingeniero Electrónico

AUTORES: Jonathan Ivan Quilumbaquin Zhunio
Joshua Isaac Reyes Heredia

TUTOR: Andrés Sebastián Calero Calero

Quito-Ecuador
2022

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, Jonathan Ivan Quilumbaquin Zhunio con documento de identificación N° 1725158891 y Joshua Isaac Reyes Heredia con documento de identificación N° 1718068578; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Quito, 05 de septiembre del año 2022

Atentamente,



Jonathan Ivan Quilumbaquin Zhunio
1725158891



Joshua Isaac Reyes Heredia
1718068578


**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE
TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, Jonathan Ivan Quilumbaquin Zhunio con documento de identificación No. 1725158891 y Joshua Isaac Reyes Heredia con documento de identificación No. 1718068578, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto Técnico : “Desarrollo de un brazo robótico serial para asistencia en una clínica odontológica” el cual ha sido desarrollado para optar por el título de: Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana

Quito, 05 de septiembre del año 2022

Atentamente,



Jonathan Ivan Quilumbaquin Zhunio
1725158891



Joshua Isaac Reyes Heredia
1718068578

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Andrés Sebastián Calero Calero con documento de identificación N° 1719252346, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación **DESARROLLO DE UN BRAZO ROBÓTICO SERIAL PARA ASISTENCIA EN UNA CLÍNICA ODONTOLÓGICA**, realizado por Jonathan Ivan Quilumbaquin Zhunio con documento de identificación N° 1725158891 y por Joshua Isaac Reyes Heredia con documento de identificación N° 1718068578, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Quito, 05 de septiembre del año 2022

Atentamente,



Ing. Andrés Sebastián Calero Calero Msc.
1719252346

DEDICATORIA Y AGRADECIMIENTO

A nuestra Familia.

Por los valores y formación inculcados en casa, además de haber impulsado nuestra carrera profesional y darnos su apoyo incondicional en cada uno de nuestros objetivos como profesionales, por la ayuda brindada a lo largo de nuestra formación estudiantil y por confiar en nosotros desde el inicio hasta la actualidad.

A nuestros maestros.

Al Ingeniero Andrés Sebastián Calero Calero por su compromiso y entrega total al proyecto técnico, además de ser un excelente profesional y darnos sus consejos sobre mejoras en el proyecto de titulación.

A la Clínica Ecudental

Por ayudarnos con la logística de materiales que no había en el país, además de su ayuda al permitirnos utilizar su nombre en este documento y financiar parte del proyecto.

A los profesionales odontólogos

Quienes con su conocimiento y experiencia sugirieron en puntos muy importantes mejoras para el proyecto, además de involucrarse probando el equipo en su consultorio.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1	2
Definición del Tema.....	2
1.1 Descripción Del Problema	2
1.2 Delimitación del Problema	2
1.3 Objetivos Generales y Específicos	2
1.3.1 Objetivo General	2
1.3.2 Objetivos Específicos	3
1.4 Justificación	3
CAPÍTULO 2	5
Marco Teórico	5
2.1 Robótica.....	5
2.2 Brazo Robótico Thor	5
2.3 Grados de Libertad.....	6
2.4 Cinemática de un Sistema Robótico	6
2.4.1 Cinemática Directa	7
2.4.2 Cinemática Inversa	7
2.5 Planificador de Trayectorias	8
2.6 Dinámica de un Sistema Robótico.....	8
2.6.1 Dinámica Directa.....	8
2.6.2 Dinámica Inversa.....	9
2.7 Hardware.....	9
2.7.1 Controlador Arduino Mega 2560 Mini Pro.....	9
2.7.2 Módulo NodeMCU ESP8266 Wifi	10
2.7.3 Módulo BLUETOOTH HC-05	10
2.7.4 Motor Nema 17 – Driver A4988	10
2.8 Software.....	11
2.8.1 Matlab.....	11
2.8.2 IDE Android Studio	11

2.8.3	IDE Arduino.....	11
2.8.4	EasyEDA.....	12
2.8.5	Ultimaker Cura.....	12
CAPÍTULO 3		13
3.1	Análisis Cinemático.....	13
3.1.1	Cinemática Directa.....	13
3.1.2	Cinemática Inversa.....	17
3.2	Análisis Dinámico.....	19
3.2.1	Dinámica Inversa.....	19
3.3	Planificador de Trayectoria.....	26
3.4	Código Fuente Android Studio	29
3.5	Código Fuente Arduino	31
3.6	Código fuente Módulo NodeMCU ESP8266	35
3.7	Impresión 3D Brazo Thor	36
3.8	Ensamble Brazo Thor	38
3.9	Circuito Electrónico	43
3.9.1	Diseño Circuito Electrónico	43
3.9.2	Grabado circuito electrónico	44
3.9.3	Grabado láser.....	45
3.9.4	Implementación circuito electrónico final.....	46
3.10	Ensamble Asistente Dental	48
CAPÍTULO 4		54
Pruebas y Resultados.....		54
4.1	Encuestas al personal odontológico.....	54
4.2	Pruebas de funcionamiento.....	58
4.2.1	Energización del Asistente Dental	58
4.2.2	Comunicación serial BLUETOOTH.....	58
4.2.3	Comunicación serial NodeMCU	59
4.3	Pruebas de funcionamiento en un ambiente controlado	60
4.3.1	Movimiento de las articulaciones.....	60
4.3.2	Respuesta de Alexa a los comandos.....	61

4.3.3	Respuesta del Asistente odontológico.....	62
4.4	Pruebas en ambiente no controlado (Consultorio odontológico).....	63
4.4.3	Respuesta a los comandos de voz	63
4.4.4	Respuesta asistente odontológico.....	64
CAPÍTULO 5	66
Conclusiones y Recomendaciones	66
Referencias	69
Anexos	71

ÍNDICE DE FIGURAS

Figura 2.1. Brazo Robótico THOR.	5
Figura 2.2. Movimientos Tridimensionales.	6
Figura 2.3. Dimensiones Arduino mega mini Pro.....	9
Figura 2.4. Esquema práctico general módulo A4988.....	11
Figura 3.1. Dimensiones y movimiento del brazo robótico THOR.	14
Figura 3.2. Programa en simulink para selección de motores.....	24
Figura 3.3 Perfil trapezoidal para la articulación 1.	25
Figura 3.4 Motor Nema 17.....	25
Figura 3.5. Animación punto 1 - punto 2.	27
Figura 3.6. Planificador de trayectoria Tiempo x Posición.....	28
Figura 3.7. Planificador de trayectoria Tiempo x Velocidad.	28
Figura 3.8. Planificador de trayectoria Tiempo x Aceleración.	29
Figura 3.9. Interfaz inicio Android Studio.	29
Figura 3.10. Interfaz Activar BLUETOOTH Android Studio.	30
Figura 3.11. Interfaz Dispositivos vinculados Android Studio.....	30
Figura 3.12. HMI Android Studio.....	31
Figura 3.13. Declaración pines Arduino.	31
Figura 3.14. Declaración de variables Arduino.	32
Figura 3.15. Declaración pines salida y puertos seriales Arduino.	32
Figura 3.16. Codificación de tiempo de operación Arduino.	33
Figura 3.17. Codificación lectura SeekBar Android Studio - Arduino.....	33
Figura 3.18. Codificación para guardar posiciones Arduino.	34
Figura 3.19. Codificación para reproducir los movimientos Arduino.	34
Figura 3.20. Codificación sentido motores horario - antihorario.....	35

Figura 3.21. Codificación módulo NodeMCU conexión Wifi.....	35
Figura 3.23. Modelado impresión 3D.	36
Figura 3.24. Impresora Anet-E12.....	37
Figura 3.25. Impresión pieza 3D.....	37
Figura 3.26. Piezas 3D terminadas.....	38
Figura 3.27. Ensamble base.	38
Figura 3.28. Ensamble rulimán y pieza rotativa de la base.....	39
Figura 3.29. Ensamble hombro.	39
Figura 3.30. Ensamble codo.....	40
Figura 3.31. Ensamble antebrazo.....	40
Figura 3.32. Ensamble muñeca.	41
Figura 3.34. Ensamble final hombro, codo, antebrazo y muñeca.	42
Figura 3.35. Ensamble brazo THOR Finalizado.....	42
Figura 3.36. Diagrama diseño PCB.....	43
Figura 3.37. Visualización 3D.	44
Figura 3.38. Impresión PCB.....	45
Figura 3.39. Transferencia impresión PCB.....	45
Figura 3.40. Interfaz software Grabado láser.....	46
Figura 3.41. Grabado láser.....	46
Figura 3.42. Espadines tipo Hembra.	47
Figura 3.43. Circuito electrónico final.	47
Figura 3.44. Mesa para el brazo robótico.....	48
Figura 3.45. Bomba succión.....	49
Figura 3.46. Fuentes de voltaje.	49
Figura 3.47. Caja electrónica.	50

Figura 3.48. Brazo mecánico.	50
Figura 3.49. Tomacorriente AC.	51
Figura 3.50. Amazon Alexa.	51
Figura 3.51. Acople manguera.	52
Figura 3.52. Montaje acople brazo.	52
Figura 3.53. Sistema final.	53
Figura 4.1. Pregunta 1.	54
Figura 4.2. Pregunta 2.	55
Figura 4.3. Pregunta 3.	55
Figura 4.4. Pregunta 4.	56
Figura 4.5. Pregunta 5.	56
Figura 4.6. Pregunta 6.	57
Figura 4.7. Solicitud aplicación.	58
Figura 4.8. Monitor serial comunicación NodeMcu.	59
Figura 4.9. Monitor serial comunicación BLUETOOTH.	59
Figura 4.10. Posición inicial.	60
Figura 4.11. Posición final de pruebas.	61
Figura 4.12. Engranés hombro.	61
Figura 4.13. Asistente odontológico.	62
Figura 4.14. Prueba taladro dental.	63
Figura 4.15. Consultorio dental.	64
Figura 4.16. Intervención asistente dental.	64

ÍNDICE TABLAS

Tabla 3.1 Parámetros D-H.....	13
Tabla 3.2 Velocidad lineal.	21
Tabla 3.3 Velocidad angular.	21
Tabla 3.4 Aceleración lineal.	22
Tabla 3.5 Aceleración angular.	23
Tabla 3.6 Parámetros dinámica inversa.	23
Tabla 3.7 Momentos para cada articulación	26
Tabla 3.8 Valores función planifica6	26
Tabla 3.9 Matriz transformación homogénea con puntos inicial y final.....	27
Tabla 3.10 Elementos electrónicos del circuito esquemático.	43
Tabla 4.1. Resultados de los comandos ejecutados.....	65

RESUMEN

Este documento presenta una revisión de los inicios de la robótica, revisando así su cronología, leyes fundamentales y su viabilidad en el campo de la medicina, para el desarrollo del asistente se utilizó diferentes tipos de software Android studio, IDE Arduino, etc.

Android Studio se utilizó para la interfaz donde el odontólogo puede interactuar desde cualquier dispositivo móvil con el brazo robótico mediante la conexión por BLUETOOTH. En la aplicación el usuario puede manipular el tiempo, las articulaciones, guardar movimientos y poner el brazo en una posición previamente establecida “HOME”. El accionamiento de las posiciones y del succionador se realizó mediante comandos de voz para lo cual se utilizó ALEXA.

Se modeló las piezas del brazo robótico en un software de impresión 3D, tomando en cuenta sus necesidades y así modificando parámetros tanto de diseño como de su misma composición, de igual forma se estudió mecánicamente el modelo del brazo robótico tanto en cinemática, dinámica y se realizó pruebas del planificador de trayectorias en el capítulo 3.

Para el análisis se realizaron pruebas en un ambiente controlado y no controlado encontrando así los resultados citados en ese apartado, a su vez se realizó una encuesta al personal de salud para adecuar el diseño a sus necesidades y finalmente se detalló las conclusiones obtenidas del trabajo final, en el capítulo 5.

Palabras claves: Robótica, Asistente dental, Alexa, Odontología, Reconocimiento de voz, impresión 3D, dinámica, cinemática.

ABSTRACT

This document presents a review of the beginning of robotics, thus reviewing its chronology, fundamental laws and its viability in the field of medicine, for the development of the assistant, different types of software were used Android studio, Arduino IDE, etc.

Android Studio was used for the interface where the dentist can interact from any mobile device with the robotic arm through the BLUETOOTH connection. In the application the user can manipulate the speed, the joints, save movements and put the arm in a previously established position "HOME".

The activation of the positions and the suction device was carried out by means of voice commands, for which ALEXA was used. The parts of the robotic arm were modeled in a 3D printing software, taking care of their needs and thus modifying both design and composition parameters, in the same way the robotic arm model was mechanically studied both in kinematics, dynamics and was tested the trajectory planner in Chapter 3.

For the analysis, tests were carried out in a controlled and uncontrolled environment, thus finding the results cited in that section, in turn, a survey was carried out on health personnel to adapt the design to their needs and finally the conclusions obtained from the final work were detailed in Chapter 5.

Keywords: Robotics, Dental assistant, Alexa, Dentistry, Voice recognition, 3D printing, dynamics, kinematics.

INTRODUCCIÓN

Este proyecto busca establecer agilidad en los procesos de succión por medio de la implementación de un brazo robótico serial, como asistente del especialista odontológico, mediante el análisis cinemático y dinámico del brazo robótico se busca encontrar el posicionamiento adecuado de los actuadores, a su vez tenemos que tomar en cuenta el sistema de control del brazo robótico para que cumpla una trayectoria específica a través de software. La configuración del posicionamiento del brazo robótico se llevará a cabo mediante una aplicación diseñada en Android Studio la cual nos permitirá guardar el movimiento que el odontólogo crea necesario. El asistente serial odontológico se accionará mediante comandos de voz los cuales fueron previamente definidos.

Se presenta la descripción del problema, la delimitación, los objetivos a los cuales se llegará y la justificación de porque surge el proyecto en el capítulo 1.

En el capítulo 2 se realiza un compendio de la información necesaria para el desarrollo y entendimiento del asistente dental, así como también características de los diferentes elementos que forman parte del proyecto técnico.

Se detalla el análisis del modelo matemático de la cinemática, dinámica, planificador de trayectorias del brazo robótico THOR en este el tercer capítulo. De igual manera el diseño, ensamblado y fabricación de la placa PCB, además de la programación en Android Studio y Arduino para la implementación del asistente serial odontológico.

Se obtiene en el capítulo 4, el análisis de resultados de las comunicaciones inalámbricas y la respuesta de actuadores en un ambiente controlado y no controlado.

Finalmente, en el capítulo 5 se detallan las conclusiones obtenidas tras las pruebas realizadas en el consultorio dental y se plantea las recomendaciones para el uso adecuado del asistente.

CAPÍTULO 1

Definición del Tema

En este capítulo se presenta la descripción del problema a resolver, la delimitación, los objetivos a los cuales se llegará y la justificación de porque surge el proyecto.

1.1 Descripción Del Problema

En una cirugía dental o procedimiento tradicional siempre existen complicaciones al momento de organizar el espacio de trabajo, es el caso especial de la clínica ECUDENTAL, pues la misma cuenta con un espacio reducido que dificulta la operatividad del cirujano y de la persona que le asiste.

Tomando en cuenta este problema los especialistas optan por trabajar solos y al momento de realizar la succión los odontólogos realizan maniobras con la manguera de succión incomodando o lastimando al paciente y en casos extremos la misma puede caer al suelo ocasionando que el instrumento quirúrgico se infecte.

1.2 Delimitación del Problema

La implementación del asistente odontológico permitirá que los especialistas de la clínica ECUDENTAL puedan realizar una intervención quirúrgica sin ayuda de otra persona optimizando el tiempo y la salubridad. Así como también aumentar la comodidad del paciente.

1.3 Objetivos Generales y Específicos

1.3.1 Objetivo General

Desarrollar un brazo robótico serial asistente para la succión de un proceso dental mediante el análisis cinemático y dinámico.

1.3.2 Objetivos Específicos

- Investigar el estado del arte para el establecimiento de la morfología del brazo robot necesario en la fase de diseño y simulación.
- Diseñar el mecanismo a través del análisis cinemático y dinámico del brazo robot para el dimensionamiento de los actuadores y su correcta implementación.
- Construir un brazo serial robótico para la colaboración del proceso de succión bucal en una intervención dental mediante una impresora 3D y PLA de alta calidad.
- Establecer el sistema de control y el planificador de trayectorias para que el brazo robot cumpla con las posiciones deseadas por el especialista al momento de realizar la succión analizando las dimensiones del espacio y su entorno de trabajo.
- Implementar un sistema de reconocimiento de 4 palabras específicas utilizando un módulo de reconocimiento de voz para que el brazo reaccione a los comandos del especialista.
- Realizar las pruebas de funcionamiento en un entorno controlado para la validación del comportamiento del brazo robot implementado.

1.4 Justificación

De acuerdo a (International Federation of Robotics, 2021) el uso de los robots industriales en fábricas de todo el mundo se está acelerando a un ritmo elevado: 126 robots por cada 10.000 empleados, casi el doble que hace cinco años (2015: 66 unidades). Por lo que se puede decir que en la actualidad la robótica es uno de los campos más importantes del mundo.

Las aplicaciones de la robótica en la medicina son extremadamente necesarias, desde la simulación de habilidades de decisión rápida, la manipulación mecánica y la versatilidad que representa el tener varios grados de libertad.

A pesar de la previa existencia de robots dentro de áreas médicas, se ha limitado únicamente a la creación de autómatas en otros campos médicos. Dejando de lado a los funcionarios odontológicos quienes son parte fundamental en la calidad de vida humana.

También el control de la contaminación es una medida obligatoria para la prevención de la infección del sitio quirúrgico. Consiste en la prevención de las unidades críticas, como: limpieza de pisos, paredes y equipamientos (Aline, Diogo, Assuca, Araujo, & Pinto, 2013). Aun tomando en cuenta estas medidas los problemas de higiene son recurrentes en la sala de operaciones y una de las necesidades más frecuentes que se presentan a diario es evitar la contaminación de los instrumentos quirúrgicos.

El presente proyecto de titulación tiene como fin ayudar a los odontólogos en el proceso de succión ya que cuando se realiza un proceso quirúrgico los odontólogos se ven obligados a utilizar sus dos manos, optando por poner la manguera de succión en la boca del paciente con maniobras dificultosas provocando incomodidad, en otros casos los especialistas optan por parar la intervención dando como resultado un mayor tiempo de operación. Con la ayuda del asistente se optimizará el tiempo de intervención quirúrgica y se evitará que la manguera pueda caer al suelo e infectarse al momento de realizar el proceso de succión manualmente.

CAPÍTULO 2

Este capítulo presenta información fundamental que permite la comprensión para el desarrollo del proyecto y las características de los diferentes elementos que forman parte de este.

Marco Teórico

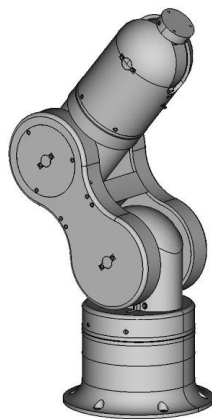
2.1 Robótica

Es una ciencia que reúne diferentes ramas tecnológicas, con el objetivo de diseñar máquinas robotizadas capaces de realizar una amplia variedad de tareas automatizadas (Revista de Robots, 2021), o de simular el comportamiento humano o animal.

2.2 Brazo Robótico Thor

Los brazos robóticos imitando la forma y funcionalidad del brazo humano, las partes de estos manipuladores o brazos son interconectadas a través de articulaciones que permiten movimientos rotacionales, traslacionales o desplazamientos lineales (Giraldo, Delgado, & Germán, 2006). Estos mecanismos ayudan a los usuarios a realizar funciones repetitivas, peligrosas o cansadas las cuales tienen un alto grado de dificultad reduciendo así el tiempo y mejorando la precisión de cada acción propuesta. Un ejemplo es el brazo robótico THOR mostrado en la Figura 1, los planos de fabricación se encuentran registrados con licencia Open Source, la cual permite que cualquier usuario pueda descargarlos para su libre uso.

Figura 2.1. Brazo Robótico THOR.



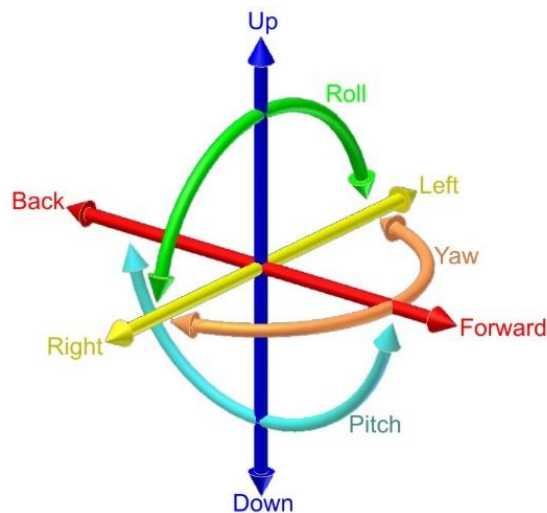
Fuente: (AngelLM, 2016).

El Brazo Robótico THOR consta de 6 grados de libertad y una altura de 625mm. Las cuales son accionadas mediante motores nema 17 de corriente continua (AngelLM, 2016).

2.3 Grados de Libertad

Los grados de libertad (GDL) determinan la capacidad de orientar la posición del elemento terminal. Frecuentemente los GDL de los robots coinciden con el número de las juntas que lo conforman. La elección de los grados de libertad necesaria de un brazo robótico viene determinada según la aplicación que va a realizar; un robot con 3 (GDL) es suficiente para posicionar elementos en planos horizontales. Sin embargo, en otras aplicaciones que se requiere orientar la herramienta terminal en el espacio tridimensional como se aprecia en la Figura 2.2 o acceder a posiciones complicadas se utiliza robots con 6 (GDL) (Barrientos, 2007).

Figura 2.2. Movimientos Tridimensionales.



Fuente: (Barrientos, 2007).

2.4 Cinemática de un Sistema Robótico

La cinemática de un manipulador robótico realiza el estudio de su movimiento con respecto a un sistema de referencia sin considerar las fuerzas que originan dicho movimiento.

2.4.1 Cinemática Directa

Consiste en determinar la posición y orientación del efector final del robot con respecto a un sistema de coordenadas de referencia (x-y-z- α - β - γ), a partir de valores conocidos como son las articulaciones y las dimensiones de los eslabones del robot (Vele, 2005). Una herramienta indispensable para la solución de la cinemática directa fue planteada por Denavit – Hartenberg en el año 1955, la cual consiste en establecer un sistema de coordenadas para cada eslabón de la cadena articulada y relacionar los parámetros geométricos con el sistema de referencia a la siguiente articulación. Esta relación se usa para describir una matriz de transformación homogénea T la cual nos sirve para encontrar la relación entre la posición y orientación del extremo del robot con respecto al sistema de referencia, esta matriz se encuentra en función de las coordenadas articulares (Giraldo et al., 2006).

Una matriz de transformación homogénea es la que transforma un vector de posición representado en coordenadas homogéneas desde un sistema de coordenadas hasta otro sistema de coordenadas.

Estructura de la matriz de transformación homogénea T:

$$T = \begin{bmatrix} \text{matriz de rotación} & \text{Vector de posición} \\ f & \text{escalado} \end{bmatrix} = \begin{bmatrix} R_{3 \times 3} & P_{3 \times 1} \\ f_{1 \times 3} & k_{1 \times 1} \end{bmatrix} = \begin{bmatrix} n_x & s_x & a_x & P_x \\ n_y & s_y & a_y & P_y \\ n_z & s_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (2.1)}$$

donde,

- R es la matriz de rotación.
- P es el vector posición del origen del sistema de coordenadas rotado con respecto al sistema de referencia.
- f es la transformación de perspectiva.
- k es el factor de escala global.

2.4.2 Cinemática Inversa

Consiste en encontrar la configuración que debe adoptar las coordenadas articulares del robot $q = (q_1, q_2, \dots, q_n)^T$ para que el efector final se oriente y posicione según una determinada posición espacial (Barrientos, 2007).

Para la solución de la cinemática inversa se tienen varios métodos tales como:

- **Matriz ampliada de Gauss-Jordán** consiste en encontrar la matriz inversa mediante el método de Gauss-Jordán.
- **Matriz adjunta y Denavit-Hartenberg** consiste en encontrar la matriz inversa a través de la matriz adjunta utilizando los parámetros de Denavit- Hartenberg.
- **Método Geométrico** consiste en encontrar relaciones geométricas en las que intervienen las coordenadas del efector final, coordenadas articulares, y las dimensiones físicas de sus elementos.
- **Método Analítico Algebraico** consiste en tomar como dato las variables y encontrar ecuaciones.

2.5 Planificador de Trayectorias

La planificación de trayectorias es una de las partes fundamentales en el correcto funcionamiento de un robot autónomo, el objetivo general de ésta es encontrar el camino más corto desde un estado inicial a uno final (Arcenegui, 2019) sin riesgo de colisión.

Existen dos maneras básicas para realizar el movimiento:

- **Control punto a punto:** Se debe suministrar puntos consecutivos para que el sistema de control automático pueda realizar la interpolación entre ellos, de tal manera que el autómatas pueda pasar por los puntos previamente configurados (Ramos, 2018).
- **Control de trayectoria continua:** Consiste en especificar el camino que debe unir los puntos mediante una trayectoria determinada (Ramos, 2018).

2.6 Dinámica de un Sistema Robótico

La dinámica estudia el movimiento acelerado y retardado de un manipulador desde la posición de reposo, mantener una velocidad constante del efector final hasta que finalmente se detenga; a través de los actuadores articulares, como son los motores eléctricos, actuadores hidráulicos y neumáticos, quiénes originan y aplican un complejo sistema de fuerzas y torques a los elementos y articulaciones del manipulador (Automatización, 2017).

2.6.1 Dinámica Directa

Consiste en el cálculo de la aceleración velocidad y posición de un robot a partir de la dinámica de pares o fuerzas impuestas por los accionamientos de potencia del robot, los efectos inerciales,

las fuerzas externas (carga útil, fuerzas de contacto o colisiones, etc.) y el sistema de control (Colorado, 2016).

2.6.2 Dinámica Inversa

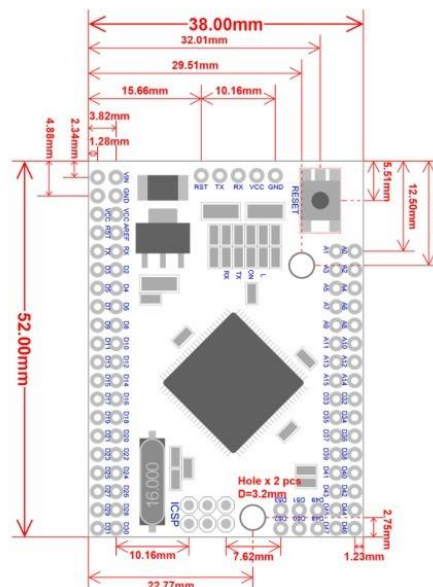
Consiste en el cálculo de las fuerzas y pares de reacción en las articulaciones requeridos por los accionamientos de potencia del manipulador, a partir de las fuerzas externas que se necesita mover (carga útil, fuerzas de contacto colisiones, etc.). La dinámica inversa esta direccionada al cálculo de los accionamientos de potencia requeridos y el diseño mecánico del robot, conforme sea la respuesta dinámica que se desea obtener (Automatización, 2017).

2.7 Hardware

2.7.1 Controlador Arduino Mega 2560 Mini Pro

El controlador Arduino mega 2560 mini pro tiene la característica de tener un tamaño reducido en comparación al Arduino mega 2560 R3 con un tamaño de 54mm x 38mm, la placa consta de un oscilador de cristal de 16Mhz, 54 pines de entrada y salida digital de los cuales 15 se pueden usar como salidas PWM, 16 como entradas análogas y 4 como UART (puertos serie de hardware), para la comunicación con la Pc se utiliza un conector Micro USB-UART CH340 (Mega, Atmega, Mega, & Pro, 2017).

Figura 2.3. Dimensiones Arduino mega mini Pro.



Fuente: (Mega et al., 2017).

2.7.2 Módulo NodeMCU ESP8266 Wifi

El módulo NodeMCU ESP8266 fue diseñado especialmente para proyectos IoT (Internet de las Cosas) y su programación se la puede realizar en diferentes lenguajes como: Arduino, Lua, MicroPython, C/C++, Scratch. La placa consta de un núcleo SoM ESP-12E, 10 pines de entrada y salida digitales, 2 pines UART para comunicación serial y conector Micro USB-UART CH340 (Candelario Elías Tutor & Luque Estepa, 2016).

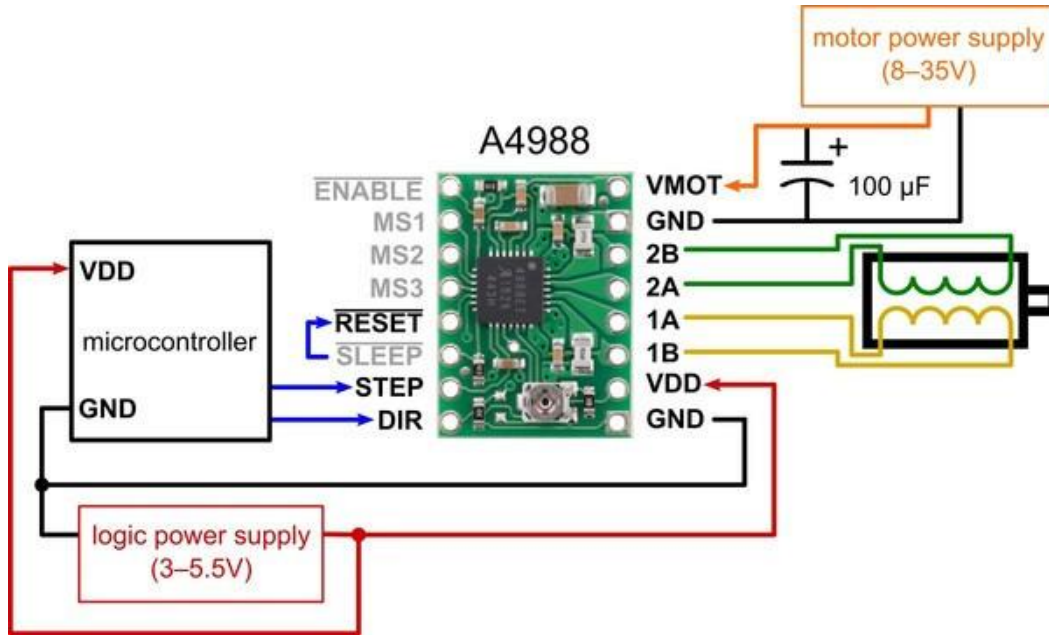
2.7.3 Módulo BLUETOOTH HC-05

El módulo permite la comunicación BLUETOOTH entre dos dispositivos un maestro y un esclavo. Este módulo viene configurado de fábrica para actuar como esclavo, es decir, viene preparado para realizar solicitudes de conexión, pero se lo puede configurar como módulo Maestro mediante comandos AT en el Software Arduino IDE. Consta de 6 pines (State, Tx, Rx, Vcc, Gnd y Key), la alimentación del módulo se la puede realizar con un voltaje entre 3.6V - 6V ya que contiene un regulador de 3.3V (Biendicho Lletí & Toledo Alarcón, 2015).

2.7.4 Motor Nema 17 – Driver A4988

Es un motor de tipo bipolar muy preciso con un ángulo de giro de 1.8° , por lo tanto, puede dividir cada revolución en 200 pasos. Es capaz de desarrollar una fuerza de 40 N.cm ya que cada bobinado soporta 0.4 [A] a 12[V] (Isaac, 2021). El control del Nema 17 se lo hace mediante el driver A4988 pues este se encarga de prever todas las protecciones necesarias de temperatura y corriente además de generar los pulsos necesarios para su correcto funcionamiento (García, 2020). Otra ventaja del driver es que solo necesitaremos 2 pines del controlador por lo que nos permite tener más puertos libres para otras funciones.

Figura 2.4. Esquema práctico general módulo A4988



Fuente: (García, 2020).

2.8 Software

2.8.1 Matlab

Es un lenguaje informático técnico (Gilat, 2006). El software es utilizado para análisis y procesamiento de datos, modelado y simulación, investigación, desarrollo de algoritmos y diseño de prototipos en universidades e institutos.

2.8.2 IDE Android Studio

Es un entorno el cual es utilizado para la creación de aplicaciones para Android la cual está basada en IntelliJ IDEA (Studio, 2021). Actualmente para la programación de las aplicaciones móviles en Android Studio se utiliza tres distintos lenguajes de programación los cuales son: Java, C++, y Kotlin.

2.8.3 IDE Arduino

Es una aplicación multiplataforma la cual se la puede utilizar en (Windows, Linux, macOS). Se utiliza para escribir, depurar, editar y grabar programas denominados sketches en el mundo de Arduino (ARDUINO, 2020).

2.8.4 EasyEDA

Es una herramienta online y gratuita la cual permite diseñar, simular circuitos electrónicos y PCB. Aprovechando el poder de la nube se puede acceder a bibliotecas de componentes Open Source los cuales son desarrollados por miles de ingenieros electrónicos (“Diseño Circuito Electrónico y Fabricación de PCB - EasyEDA,” 2017).

La ventaja de EasyEDA es la facilidad de modificar los proyectos ya que los mismos se encuentran en la nube y solo se necesita un ordenador y acceso a la Web.

2.8.5 Ultimaker Cura

Es una aplicación pensada para impresoras 3D, en la que se pueden alterar los parámetros de impresión y luego transformarlos a código G, que es el lenguaje encargado de llevar un diseño al mundo real. Ultimaker Cura es el software de impresión 3D más popular del mundo. Debido a su fácil uso puedes hacer impresiones con unos cuantos clics (“Ultimaker Cura: Software de Impresión 3D,” 2017).

CAPÍTULO 3

En este capítulo se detalla el análisis del modelo matemático de la cinemática, dinámica, planificador de trayectorias del brazo robótico THOR, De igual manera el diseño, ensamblado y fabricación de la placa PCB, además de la programación en Android Studio y Arduino para la implementación del Asistente serial odontológico.

3.1 Análisis Cinemático

3.1.1 Cinemática Directa

En la Tabla 3.1 se presentan los parámetros de Denavit-Hartenberg para el brazo robótico THOR con 6 grados de libertad tomando en cuenta a la posición del brazo robótico de la Figura 3.1.

Tabla 3.1 Parámetros D-H.

Articulación	θ_i	d	a	α
1	θ_1	202	0	$-\pi/2$
2	θ_2	0	160	0
3	θ_3	0	0	$\pi/2$
4	θ_4	195	0	$-\pi/2$
5	θ_5	0	67.15	$\pi/2$
6	θ_6	0	0	0

Parámetros D-H para el brazo con 6 GL, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Donde:

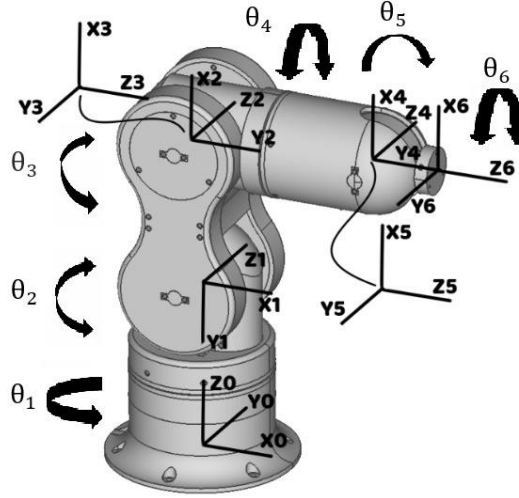
θ_i = Representa el movimiento de rotación del sistema de coordenadas entorno al eje “Z”.

d = Representa la distancia del movimiento de traslación del sistema de coordenadas a lo largo del eje “Z”.

a = Representa la distancia del movimiento de traslación del sistema de coordenadas a lo largo del eje “X”

α = Representa el movimiento de rotación del sistema de coordenadas entorno al eje “X”.

Figura 3.1. Dimensiones y movimiento del brazo robótico THOR.



Movimientos para los parámetros Denavit-Hartenberg, Fuente: (AngellM, 2016).

Para obtener las ecuaciones de posición de la cinemática directa, se procede a encontrar la matriz de transformación 0_6A como se muestra en la Ec. (3.1).

$$T_{(n,o,a)} = {}^0_6A \quad \text{Ec. (3.1)}$$

$${}^0_6A = {}^0_1A * {}^1_2A * {}^2_3A * {}^3_4A * {}^4_5A * {}^5_6A \quad \text{Ec. (3.2)}$$

Donde:

$${}^0_1A = R(Z, \theta_1) * T(Z, l_1) * R(X, -90) \quad \text{Ec. (3.3)}$$

$${}^1_2A = R(Z, \theta_2 + 90) * T(X, l_2) \quad \text{Ec. (3.4)}$$

$${}^2_3A = R(Z, \theta_3) * R(X, 90) \quad \text{Ec. (3.5)}$$

$${}^3_4A = R(Z, \theta_4) * T(Z, l_4) * R(X, -90) \quad \text{Ec. (3.6)}$$

$${}^4_5A = R(Z, \theta_5) * T(Z, l_5) * R(X, 90) \quad \text{Ec. (3.7)}$$

$${}^5_6A = R(Z, \theta_6) \quad \text{Ec. (3.8)}$$

A continuación, se procede a multiplicar las matrices de rotación y traslación obtenidas de la Tabla 3.1.

$${}^1_0A = \begin{bmatrix} \cos \theta_1 & 0 & -\sin \theta_1 & 0 \\ \sin \theta_1 & 0 & \cos \theta_1 & 0 \\ 0 & -1 & 0 & 202 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (3.9)}$$

$${}^2_1A = \begin{bmatrix} \cos \theta_2 & 0 & -\sin \theta_2 & 160 * \cos \theta_2 \\ \sin \theta_2 & \cos \theta_2 & 0 & 160 * \sin \theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (3.10)}$$

$${}^3_2A = \begin{bmatrix} \cos \theta_3 & 0 & \sin \theta_3 & 0 \\ \sin \theta_3 & 0 & -\cos \theta_3 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (3.11)}$$

$${}^4_3A = \begin{bmatrix} \cos \theta_4 & 0 & -\sin \theta_4 & 0 \\ \sin \theta_4 & 0 & \cos \theta_4 & 0 \\ 0 & -1 & 0 & 195 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (3.12)}$$

$${}^5_4A = \begin{bmatrix} \cos \theta_5 & 0 & \sin \theta_5 & 67 * \cos \theta_5 \\ \sin \theta_5 & 0 & -\cos \theta_5 & 67 * \sin \theta_5 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (3.13)}$$

$${}^6_5A = \begin{bmatrix} \cos \theta_6 & -\sin \theta_6 & 0 & 0 \\ \sin \theta_6 & \cos \theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (3.14)}$$

Por último, se reemplaza las matrices en la EC. 3.2, para ello se utilizó la siguiente nomenclatura.

$$\cos \theta_i = Ci$$

$$\sin \theta_i = Si$$

$$\cos(\theta_i + \theta_j) = C(i,j)$$

$$\sin(\theta_i + \theta_j) = S(i,j)$$

$$\cos(\theta_i - \theta_j) = C(-i,j)$$

$$\sin(\theta_i - \theta_j) = S(-i,j)$$

$${}^6_0A = \begin{bmatrix} a & d & g & j \\ b & e & h & k \\ c & f & i & l \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. (3.15)}$$

Donde:

$$a = -S_6 * (C_4 * S_1 - S_4 * (C_1 * S_2 * S_3 - C_1 * C_2 * C_3)) - C_6 * (C_5 * (S_1 * S_4 + C_4 * (C_1 * S_2 * S_3 - C_1 * C_2 * C_3)) + S_5 * (C_1 * C_2 * S_3 + C_1 * C_3 * S_2)).$$

$$b = S_6 * (C_1 * C_4 + S_4 * (S_1 * S_2 * S_3 - C_2 * C_3 * S_1)) + C_6 * (C_5 * (C_1 * S_4 - C_4 * (S_1 * S_2 * S_3 - C_2 * C_3 * S_1)) - S_5 * (C_2 * S_1 * S_3 + C_3 * S_1 * S_2)).$$

$$c = S(2,3) * S_4 * S_6 - C_6 * (C(2,3) * S_5 + S(2,3) * C_4 * C_5).$$

$$d = S_6 * (C_5 * (S_1 * S_4 + C_4 * (C_1 * S_2 * S_3 - C_1 * C_2 * C_3)) + S_5 * (C_1 * C_2 * S_3 + C_1 * C_3 * S_2)) - C_6 * (C_4 * S_1 - S_4 * (C_1 * S_2 * S_3 - C_1 * C_2 * C_3)).$$

$$e = C_6 * (C_1 * C_4 + S_4 * (S_1 * S_2 * S_3 - C_2 * C_3 * S_1)) - S_6 * (C_5 * (C_1 * S_4 - C_4 * (S_1 * S_2 * S_3 - C_2 * C_3 * S_1)) - S_5 * (C_2 * S_1 * S_3 + C_3 * S_1 * S_2)).$$

$$f = S_6 * (C_{2,3} * S_5 + S(2,3) * C_4 * C_5) + S_{2,3} * C_6 * S_4.$$

$$g = C_5 * (C_1 * C_2 * S_3 + C_1 * C_3 * S_2) - S_5 * (S_1 * S_4 + C_4 * (C_1 * S_2 * S_3 - C_1 * C_2 * C_3)).$$

$$h = S_5 * (C_1 * S_4 - C_4 * (S_1 * S_2 * S_3 - C_2 * C_3 * S_1)) + C_5 * (C_2 * S_1 * S_3 + C_3 * S_1 * S_2).$$

$$i = C(2,3) * C_5 - S(2,3) * C_4 * S_5.$$

$$j = 160 * C_1 * C_2 - 67 * C_5 * (S_1 * S_4 + C_4 * (C_1 * S_2 * S_3 - C_1 * C_2 * C_3)) - 67.15 * S_5 * (C_1 * C_2 * S_3 + C_1 * C_3 * S_2) + 195 * C_1 * C_2 * S_3 + 195 * C_1 * C_3 * S_2.$$

$$k = 160 * C2 * S1 + 67.15 * C5 * (C1 * S4 - C4 * (S1 * S2 * S3 - C2 * C3 * S1)) - 67.15 * S5 * (C2 * S1 * S3 + C3 * S1 * S2) + 195 * C2 * S1 * S3 + 195 * C3 * S1 * S2.$$

$$l = 195 * C2,3 - 160 * S2 - \frac{(67.15 * C(-4,5) * S(2,3))}{2} - \frac{(67.15 * C(4,5) * S(2,3))}{2} - 67.15 * C(2,3) * S5 + 202.$$

Finalmente, la matriz 0A se multiplica por el punto final EC.(3.16) dando como resultado el punto inicial (X, Y, Z).

$$P_0 = {}^0A * \begin{bmatrix} 0 \\ 0 \\ 9 \\ 1 \end{bmatrix} P_F \quad \text{Ec. (3.16)}$$

Reemplazando en la EC. (3.16) se obtiene las ecuaciones (X, Y, Z) del plano cartesiano:

$$X = 5 * C1 * (39 * S(2,3) + 32 * C2) - 9 * S5 * (S1 * S4 - \cos(t2, t3) * C1 * C4) - 67 * C5 * (S1 * S4 - \cos(t2, t3) * C1 * C4) + 9 * S(2,3) * C1 * C5 - 67 * S(2,3) * C1 * S \quad \text{Ec. (3.17)}$$

$$Y = 67 * C5 * (C1 * S4 + \cos(t2, t3) * C4 * S1) + 9 * S5 * (C1 * S4 + \cos(t2, t3) * C4 * S1) + 5 * S1 * (39 * S(2,3) + 32 * C2) + 9 * S(2,3) * C5 * S1 - 67 * S(2,3) * S1 * S5 \quad \text{Ec. (3.18)}$$

$$Z = 195 * \cos(t2, t3) - 160 * S2 + 9 * \cos(t2, t3) * C5 - 67 * \cos(t2, t3) * S5 - 67 * S(2,3) * C4 * C5 - 9 * S(2,3) * C4 * S5 + 202 \quad \text{Ec. (3.19)}$$

3.1.2 Cinemática Inversa

La cinemática inversa se obtiene a través del método de Denavit – Hartenberg el cual permite obtener los valores articulares de θ_i a partir de una posición específica del plano cartesiano.

Como se muestra en la Ec. (3.20) a Ec. (3.24).

$$({}^1_0A)^{-1} * \begin{bmatrix} n_x & o_x & a_x & X \\ n_y & o_y & a_y & Y \\ n_z & o_z & a_z & Z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^2_1A * {}^3_2A * {}^4_3A * {}^5_4A * {}^6_5A$$

$$\text{Ec. (3.20)}$$

$$({}_1^2A)^{-1} * ({}_0^1A)^{-1} * T_{(n,o,a)} = {}_2^3A * {}_3^4A * {}_4^5A * {}_5^6A \quad \text{Ec. (3.21)}$$

$$({}_2^3A)^{-1} * ({}_1^2A)^{-1} * ({}_0^1A)^{-1} * T_{(n,o,a)} = {}_3^4A * {}_4^5A * {}_5^6A \quad \text{Ec. (3.22)}$$

$$({}_3^4A)^{-1} * ({}_2^3A)^{-1} * ({}_1^2A)^{-1} * ({}_0^1A)^{-1} * T_{(n,o,a)} = {}_4^5A * {}_5^6A \quad \text{Ec. (3.23)}$$

$$({}_4^5A)^{-1} * ({}_3^4A)^{-1} * ({}_2^3A)^{-1} * ({}_1^2A)^{-1} * ({}_0^1A)^{-1} * T_{(n,o,a)} = {}_5^6A \quad \text{Ec. (3.24)}$$

A continuación, se procede a encontrar las ecuaciones finales para cada uno de los movimientos articulares θ_i .

$$\theta_1 = \tan^{-1} \left(\frac{-Y * (67.15 * C5 * S4)}{X} \right) \quad \text{Ec. (3.25)}$$

$$\theta_2 = \tan^{-1} \left(\frac{(X * C1 + Y * S1) * (160 + (195 * S3 - 67.15 * S3 * S5 + 67.15 * C3 * C4 * C5))}{Z + 202} \right) \quad \text{Ec. (3.26)}$$

$$\theta_3 = \sin^{-1} \left(\frac{-160 + C2 * (X * C1 + Y * S1) - S2 * (Z + 202) - 67.15 * C3 * C4 * C5}{195 - 67.15 * S5} \right) \quad \text{Ec. (3.27)}$$

$$\theta_4 = \sin^{-1} \left(- \frac{\tan \theta_1 + Y/X}{67.15 * C4} \right) \quad \text{Ec. (3.28)}$$

$$\theta_5 = \cos^{-1} \left(- \frac{\tan \theta_1 + Y/X}{67.15 * S4} \right) \quad \text{Ec. (3.29)}$$

$$\theta_6 = \tan^{-1} \left(\frac{X * (C1 * (C2 * (C5 * S3 + C3 * C4 * S5) + S2 * (C3 * C5 - C4 * S3 * S5)) - S1 * S4 * S5) + 202 * S2 * (C5 * S3 + C3 * C4 * S5)}{Y * (S1 * (C2 * (C5 * S3 + C3 * C4 * S5)) + C1 * S4 * S5) + Z * (C2 * (C3 * C5 - C4 * S3 * S5) - S2 * (C5 * S3 + C3 * C4 * S5))} \right) \quad \text{Ec. (3.30)}$$

3.2 Análisis Dinámico

3.2.1 Dinámica Inversa

El modelo dinámico inverso se obtiene mediante el método de Newton-Euler el cual calcula el par motriz en cada una de las articulaciones.

Este método consiste en definir la cinemática hacia adelante calculando la aceleración lineal, aceleración angular, velocidad lineal, velocidad angular tomando como punto de referencia la base hasta el efector final. El cálculo recursivo de la dinámica hacia atrás consiste en obtener la fuerza, el momento y el torque necesarios para cada una de las articulaciones comenzando desde la pinza (incluyendo la carga útil) hasta la base del robot.

Para obtener la velocidad angular y velocidad lineal se requiere hallar la matriz jacobiana a partir de la Ec. (3.31).

$$Jv = \vec{Z}_{i-1} \times (\vec{r}_6 \times \vec{r}_i)$$

Ec.(3.31)

De donde se obtiene que:

$$Z_0 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} Z_1 = \begin{pmatrix} -\sin \theta_1 \\ \cos \theta_1 \\ 0 \end{pmatrix} Z_2 = \begin{pmatrix} -\sin \theta_1 \\ \cos \theta_1 \\ 0 \end{pmatrix} Z_3 = \begin{pmatrix} \sin(\theta_2 + \theta_3) \cos \theta_1 \\ \sin(\theta_2 + \theta_3) \sin \theta_1 \\ 0 \end{pmatrix}$$

$$Z_4 = \begin{pmatrix} S4 * ((C1 * S2 * S3) - (C1 * C2 * C3) - C4 * S1) \\ C1 * C4 + S4 * (S1 * S2 * S3 - C2 * C3 * S1) \\ S(2,3) * S4 \end{pmatrix}$$

Z_5

$$= \begin{pmatrix} C5 * ((C1 * C2 * S3) + (C1 * C3 * S2)) - S5 * (S1 * S4 + C4 * (C1 * S2 * S3 - C1 * C2 * C3)) \\ S5 * (C1 * S4 - C4 * (S1 * S2 * S3 - C2 * C3 * S1)) + C5 * (C2 * S1 * S3 + C3 * S1 * S2) \\ C(2,3) * C5 - S(2,3) * C4 * C5 \end{pmatrix}$$

$$Z_6 = \begin{pmatrix} C5 * (C1 * C2 * S3 + C1 * C3 * S2) - S5 * (S1 * S4 + C4 * C1 * S2 * S3 - C1 * C2 * C3) \\ S5 * (C1 * S4 - C4 * (S1 * S2 * S3 - C2 * C3 * S1)) + C5 * (C2 * S1 * S3 + C3 * S1 * S2) \\ C(2,3) * C5 - S(2,3) * C4 * S5 \end{pmatrix}$$

De la misma manera se obtienen los vectores de posición:

$$\vec{r}_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \vec{r}_1 = \begin{pmatrix} 0 \\ 0 \\ 202 \end{pmatrix} \vec{r}_2 = \begin{pmatrix} 160 * C1 * C2 \\ 160 * C2 * S1 \\ 202 - 160 * S2 \end{pmatrix} \vec{r}_3 = \begin{pmatrix} 160 * C1 * C2 \\ 160 * C2 * S1 \\ 202 - 160 * S2 \end{pmatrix}$$

$$\vec{r}_4 = \begin{pmatrix} 5 * C1 * (39 * S(2,3) + 32 * C2) \\ 5 * S1 * (39 * S(2,3) + 32 * C2) \\ 195 * C(2,3) - 160 * S2 + 202 \end{pmatrix}$$

$$\vec{r}_5 = \begin{pmatrix} 160 * C1 * C2 - 67.15 * C5 * (S1 * S4 + C4 * (C1 * S2 * S3 - C1 * C2 * C3)) - 67.15 * S5 \\ 160 * C2 * S1 + 67.15 * C5 * (C1 * S4 - C4 * (S1 * S2 * S3 - C2 * C3 * S1)) - 67.15 * S5 \\ 195 * C(2,3) - 160 * S2 - (67.15 * C(4,5)) \end{pmatrix}$$

$$\begin{pmatrix} * (C1 * C2 * S3 + C1 * C3 * S2) + 195 * C1 * C2 * S3 + 195 * C1 * C3 * S2 \\ * (C2 * S1 * S3 + C3 * S1 * S2) + 195 * C2 * S1 * S3 + 195 * C3 * S1 * S2 \\ * \frac{S2,3}{2} - (67.15 * C(4,5)) * \frac{S2,3}{2} - 67.15 * C(2,3) * S5 + 202 \end{pmatrix}$$

$$\vec{r}_6 = \begin{pmatrix} 160 * C1 * C2 - 67.15 * C5 * (S1 * S4 + C4 * (C1 * S2 * S3 - C1 * C2 * C3)) - 67.15 * S5 \\ 160 * C2 * S1 + 67.15 * C5 * (C1 * S4 - C4 * (S1 * S2 * S3 - C2 * C3 * S1)) - 67.15 * S5 \\ 195 * C(2,3) - 160 * S2 - (67.15 * C - 4,5) \end{pmatrix}$$

$$\begin{pmatrix} * (C1 * C2 * S3 + C1 * C3 * S2) + 195 * C1 * C2 * S3 + 195 * C1 * C3 * S2 \\ * (C2 * S1 * S3 + C3 * S1 * S2) + 195 * C2 * S1 * S3 + 195 * C3 * S1 * S2 \\ * \frac{S2,3}{2} - (67.15 * C(4,5)) * \frac{S2,3}{2} - 67.15 * C(2,3) * S5 + 202 \end{pmatrix}$$

Utilizando la Ec. (3.31) se puede hallar la matriz jacobiana y posteriormente encontrar las ecuaciones de velocidad y aceleración.

Tabla 3.2 Velocidad lineal.

$\dot{\theta}_1$	\dot{x}	$67.15 * S(2,3) * S1 * S5 - 5 * S1 * (39 * S(2,3) + 32 * C2) - 67.15 * C5 * (C1 * S4 + C(2,3) * C4 * S1).$
	\dot{y}	$5 * C1 * (39 * S(2,3) + 32 * C2) - 67.15 * C5 * (S1 * S4 - C(2,3) * C1 * C4) - 67.15 * S(2,3) * C1 * S5.$
	\dot{z}	0.
$\dot{\theta}_2$	\dot{x}	$-C1 * (160 * S2 - 195 * C(2,3) + 67.15 * C(2,3) * S5 + 67.15 * S(2,3) * C4 * C5.)$
	\dot{y}	$-S1 * (160 * S2 - 195 * C(2,3) + 67.15 * C(2,3) * S5 + 67.15 * S(2,3) * C4 * C5.)$
	\dot{z}	$67.15 * S(2,3) * S5 - 160 * C2 - 195 * S(2,3) - 67.15 * C(2,3) * C4 * C5.$
$\dot{\theta}_3$	\dot{x}	$-C1 * (67.15 * C(2,3) * S5 - 195 * C(2,3) + 67.15 * S(2,3) * C4 * C5.)$
	\dot{y}	$-S1 * (67.15 * C(2,3) * S5 - 195 * C(2,3) + 67.15 * S(2,3) * C4 * C5.)$
	\dot{z}	$67.15 * S(2,3) * S5 - 195 * S(2,3) - 67.15 * C(2,3) * C4 * C5.$
$\dot{\theta}_4$	\dot{x}	$-67.15 * C5 * (C4 * S1 + C(2,3) * C1 * S4).$
	\dot{y}	$67.15 * C5 * (C1 * C4 - C(2,3) * S1 * S4).$
	\dot{z}	$67.15 * S(2,3) * C5 * S4.$
$\dot{\theta}_5$	\dot{x}	$67.15 * S5 * (S1 * S4 - C(2,3) * C1 * C4) - 67.15 * S(2,3) * C1 * C5.$
	\dot{y}	$-67.15 * S5 * (C1 * S4 + C(2,3) * C4 * S1) - 67.15 * S(2,3) * C5 * S1.$
	\dot{z}	$67.15 * S(2,3) * C4 * S5 - 67.15 * C(2,3) * C5.$
$\dot{\theta}_6$	\dot{x}	0.
	\dot{y}	0.
	\dot{z}	0.

Ecuaciones de velocidad lineal, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Tabla 3.3 Velocidad angular.

$\dot{\theta}_1$	\dot{w}_x	0.
	\dot{w}_y	0.
	\dot{w}_z	1.
$\dot{\theta}_2$	\dot{w}_x	$-S1.$
	\dot{w}_y	$C1.$
	\dot{w}_z	0.
$\dot{\theta}_3$	\dot{w}_x	$-S1.$
	\dot{w}_y	$C1.$
	\dot{w}_z	0.
$\dot{\theta}_4$	\dot{w}_x	$S(2,3) * C1.$
	\dot{w}_y	$S(2,3) * S1.$
	\dot{w}_z	$C(2,3).$
$\dot{\theta}_5$	\dot{w}_x	$-C4 * S1 - C(2,3) * C1 * S4.$
	\dot{w}_y	$C1 * C4 - C(2,3) * S1 * S4.$
	\dot{w}_z	$S(2,3) * S4.$
$\dot{\theta}_6$	\dot{w}_x	$S(2,3) * C1 * C5 - S5 * S1 * S4 - C(2,3) * C1 * C4.$
	\dot{w}_y	$S5 * C1 * S4 + C(2,3) * C4 * S1 + S(2,3) * C5 * S1.$
	\dot{w}_z	$C(2,3) * C5 - S(2,3) * C4 * S5.$

Ecuaciones de velocidad angular, Autores: Jonathan Quilumbaquin, Isaac Reyes.

De igual forma se obtiene el vector de aceleración a partir de la Ec. (3.32).

$$\vec{a}_i = \left[\frac{\partial \vec{V}_1}{\partial \theta_1}, \frac{\partial \vec{V}_2}{\partial \theta_2}, \frac{\partial \vec{V}_3}{\partial \theta_3}, \frac{\partial \vec{V}_4}{\partial \theta_4}, \frac{\partial \vec{V}_5}{\partial \theta_5}, \frac{\partial \vec{V}_6}{\partial \theta_6} \right]$$

Ec.(3.32)

Tabla 3.4 Aceleración lineal.

$\ddot{\theta}_1$	\ddot{x}	$67.15 * C5 * S1 * S4 - C(2,3) * C1 * C4 - 5 * C1 * 39 * S(2,3) + 32 * C2 + 67.15 * S(2,3) * C1 * S5.$
	\ddot{y}	$67.15 * S2,3 * S1 * S5 - 5 * S1 * 39 * S(2,3) + 32 * C2 - 67.15 * C5 * C1 * S4 + C(2,3) * C4 * S1$
	\ddot{z}	0.
$\ddot{\theta}_2$	\ddot{x}	$-C1 * 195 * S(2,3) + 160 * C2 - 67.15 * S(2,3) * S5 + 67.15 * C(2,3) * C4 * C5.$
	\ddot{y}	$-S1 * (195 * S(2,3) + 160 * C2 - 67.15 * S(2,3) * S5 + 67.15 * C(2,3) * C4 * C5.)$
	\ddot{z}	$160 * S2 - 195 * C(2,3) + 67.15 * C(2,3) * S5 + 67.15 * S(2,3) * C4 * C5.$
$\ddot{\theta}_3$	\ddot{x}	$-C1 * (67.15 * C(2,3) * S5 - 195 * C(2,3) + 67.15 * S(2,3) * C4 * C5.)$
	\ddot{y}	$-S1 * (67.15 * C(2,3) * S5 - 195 * C(2,3) + 67.15 * S(2,3) * C4 * C5.)$
	\ddot{z}	$67.15 * S(2,3) * S5 - 195 * S(2,3) - 67.15 * C(2,3) * C4 * C5.$
$\ddot{\theta}_4$	\ddot{x}	$67.15 * C5 * (S1 * S4 - C(2,3) * C1 * C4).$
	\ddot{y}	$-67.15 * C5 * (C1 * S4 + C(2,3) * C4 * S1).$
	\ddot{z}	$67.15 * S(2,3) * C5 * C4.$
$\ddot{\theta}_5$	\ddot{x}	$67.15 * C5 * (S1 * S4 - C(2,3) * C1 * C4) + 67.15 * S(2,3) * C1 * S5.$
	\ddot{y}	$67.15 * S(2,3) * S1 * S5 - 67.15 * C5 * (C1 * C5 * C1 * S4) * C(2,3) * C4 * S1.$
	\ddot{z}	$67.15 * C(2,3) * S5 + 67.15 * S(2,3) * C5 * C4$
$\ddot{\theta}_6$	\ddot{x}	0.
	\ddot{y}	0.
	\ddot{z}	0.

Ecuaciones de aceleración lineal, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Tabla 3.5 Aceleración angular.

$\ddot{\theta}_1$	$\ddot{w}x$	0.
	$\ddot{w}y$	0.
	$\ddot{w}z$	0.
$\ddot{\theta}_2$	$\ddot{w}x$	$-C1.$
	$\ddot{w}y$	$-S1.$
	$\ddot{w}z$	0.
$\ddot{\theta}_3$	$\ddot{w}x$	$-C1.$
	$\ddot{w}y$	$-S1.$
	$\ddot{w}z$	0.
$\ddot{\theta}_4$	$\ddot{w}x$	$S(2,3) * C1.$
	$\ddot{w}y$	$S(2,3) * S1.$
	$\ddot{w}z$	$C(2,3).$
$\ddot{\theta}_5$	$\ddot{w}x$	$-C4 * S1 - C(2,3) * C1 * S4.$
	$\ddot{w}y$	$C1 * C4 - C(2,3) * S1 * S4.$
	$\ddot{w}z$	$S(2,3) * S4.$
$\ddot{\theta}_6$	$\ddot{w}x$	0.
	$\ddot{w}y$	0.
	$\ddot{w}z$	1.

Ecuaciones de aceleración angular, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Una vez se han encontrado las ecuaciones se procede a la selección de motores en base a la dinámica inversa para esto es necesario simular en Matlab en el modelo que se muestra en la Figura 3.2 este modelo se obtuvo de Mathworks (Manurung, 2022).

Los parámetros utilizados para la simulación de la dinámica inversa se presentan en la Tabla 3.6. para el perfil trapezoidal con el método de desacoplo cinemático.

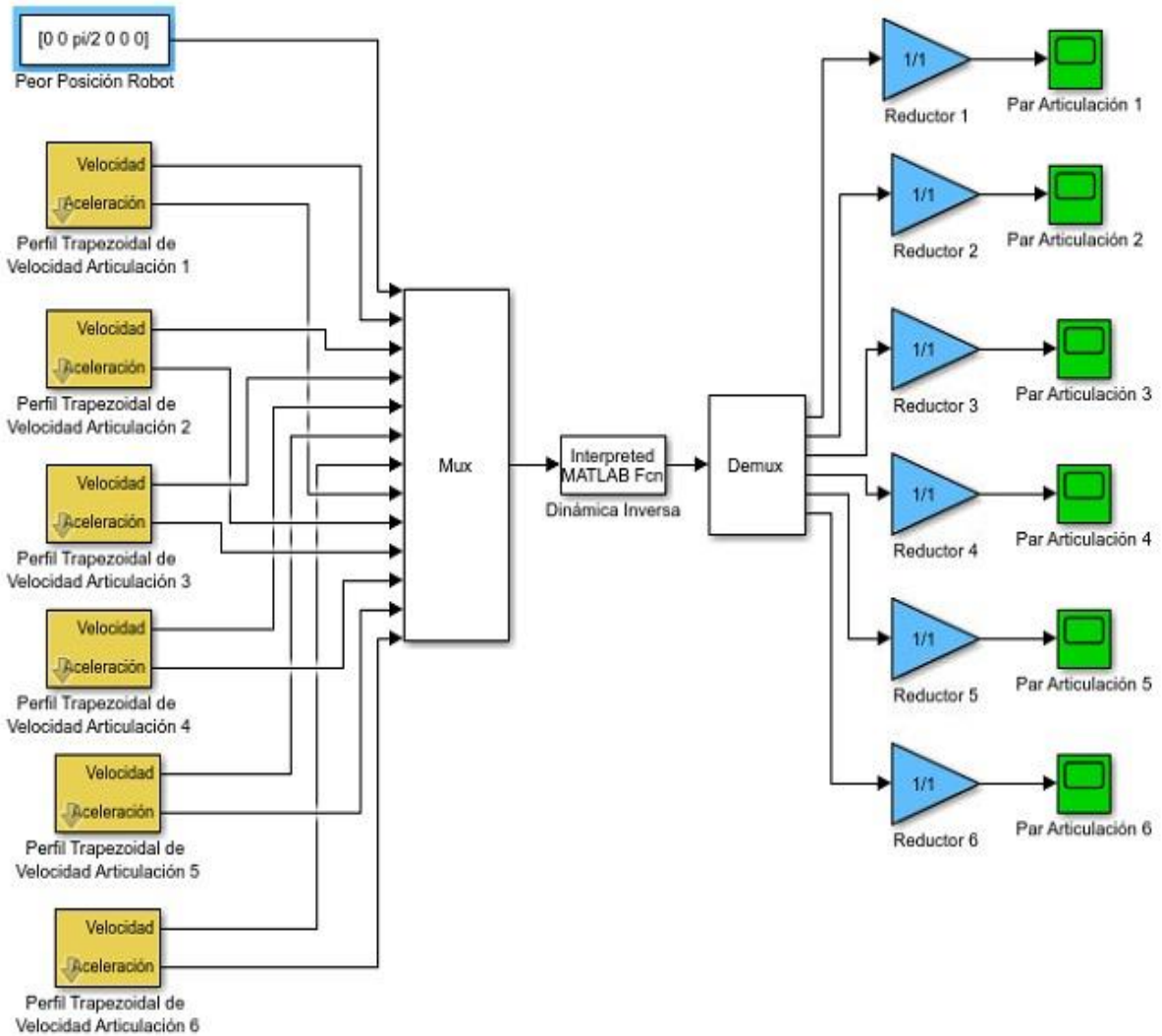
Tabla 3.6 Parámetros dinámica inversa.

Articulaciones	Velocidad máxima	Tiempo aceleración	Tiempo constante	Tiempo deceleración
1	$\pi/3$	0.1	0.4	0.1
2	$-\pi/3$	0.1	0.4	0.1
3	$-\pi/3$	0.1	0.4	0.1
4	$\pi/3$	0.1	0.4	0.1
5	$-\pi/3$	0.1	0.4	0.1
6	$\pi/3$	0.1	0.4	0.1

Parámetros de velocidad y tiempo utilizados para la dinámica inversa, Autores: Jonathan

Quilumbaquin, Isaac Reyes.

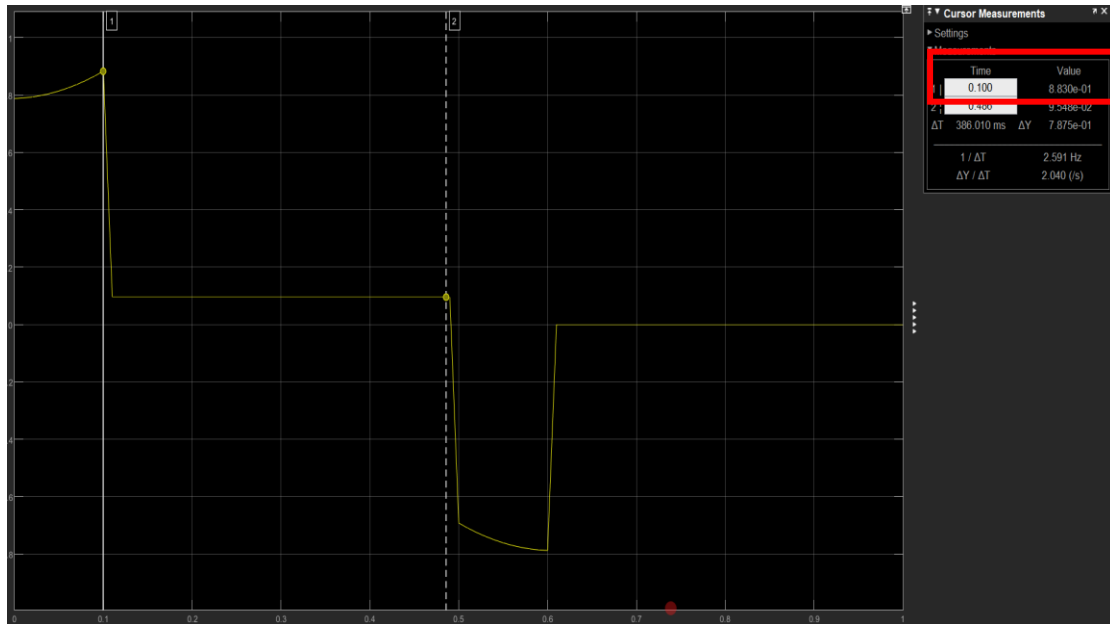
Figura 3.2. Programa en simulink para selección de motores.



Función para la dinámica inversa, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Para la selección de motores se consultó en una tienda virtual con catálogos de los motores a pasos más recientes (Figura 3.4), es necesario sobredimensionar un poco el motor que se va a adquirir conforme a sus datos de par pico y par nominal obtenidos de los perfiles trapezoidales, tal como se muestra en la Figura 3.3.

Figura 3.3 Perfil trapezoidal para la articulación 1.



Par pico y par nominal de la articulación 1, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Figura 3.4 Motor Nema 17.



Motor articulación 1, Fuente (STEPPERONLINE, 2022).

Se simuló para cada una de las articulaciones como en la Figura 3.3 y la Figura 3.4 obteniendo así los valores detallados en la Tabla 3.7. Los perfiles trapezoidales de las articulaciones (2, 3, 4, 5,6) se muestran en el Anexo 1.

Tabla 3.7 Momentos para cada articulación

Articulaciones	Par Pico	Par Nominal	Par del motor seleccionado
Articulación 1	0.88 Nm	0.099 Nm	1.2 Nm
Articulación 2	0.19 Nm	0.012 Nm	0.26 Nm
Articulación 3	0.22 Nm	0.082 Nm	0.26 Nm
Articulación 4	0.153 Nm	0.08 Nm	0.26 Nm
Articulación 5	0.088 Nm	0.023 Nm	0.04 Nm
Articulación 6	0.048 Nm	0.012 Nm	0.04 Nm

Valor par de cada articulación, Autores: Jonathan Quilumbaquin, Isaac Reyes.

3.3 Planificador de Trayectoria

Para generar el planificador de trayectorias se utilizará la función en Matlab `planifica6(p1, p2, n, s, a, codo, muñeca, npuntos)`. (Anexo 2) A esta función se le introduce coordenadas cartesianas del punto de inicio como del punto final, la matriz de orientación `n s a`, los parámetros de codo y muñeca para la configuración del brazo robótico y el número de puntos intermedios.

Tabla 3.8 Valores función `planifica6`

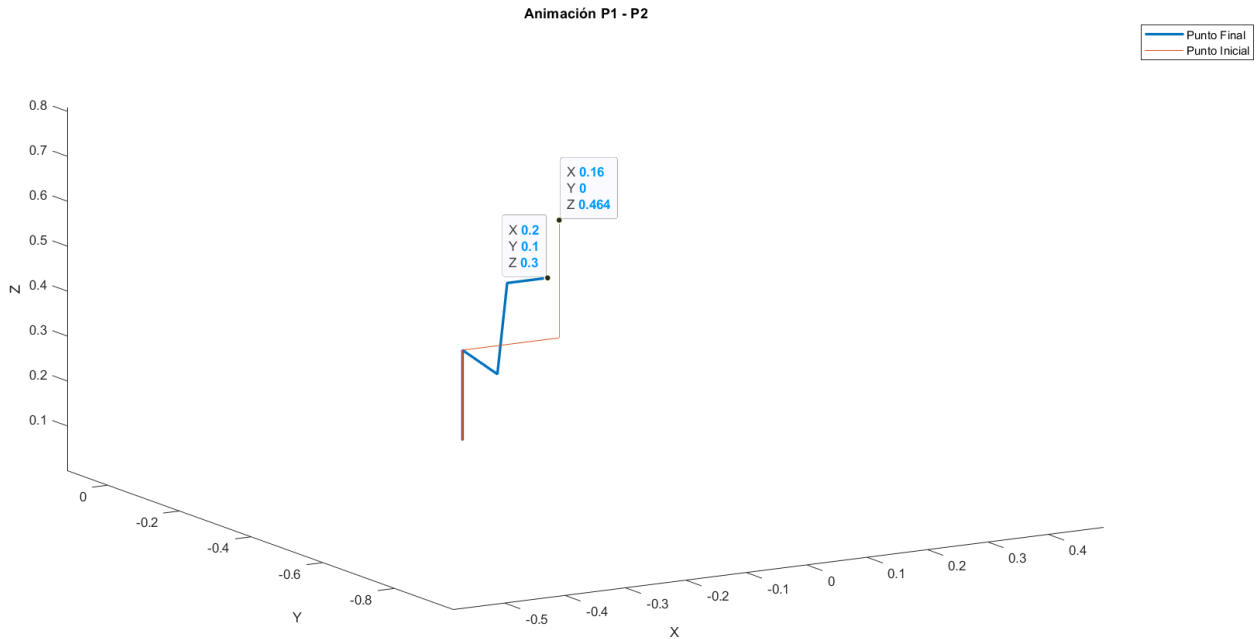
p1	[0.16 0 0.464]'
p2	[0.2 0.1 0.3]'
n	[0 0 1]'
s	[0 1 0]'
a	[1 0 0]'
Codo	-1
Muñeca	-1
N° puntos	100

Valores de la función en Matlab `planifica6(p1, p2, n, s, a, codo, muñeca, npuntos)`, Autores:

Jonathan Quilumbaquin, Isaac Reyes.

En el Anexo 3 se muestra la función `animación(mat_q)`, la cual nos permite animar el movimiento del brazo desde el punto inicial P1 al punto final P2 como se muestra en la Figura 3.5.

Figura 3.5. Animación punto 1 - punto 2.



Función animación (mat_q), Autores: Jonathan Quilumbaquin, Isaac Reyes.

A continuación, se procede a ingresar la matriz de transformación homogénea con los puntos inicial ($T1 = [0 \ 1 \ 0 \ 0.16; 1 \ 0 \ 0 \ 0; 0 \ 0 \ 1 \ 0.464; 0 \ 0 \ 0 \ 1]$;) y final ($T2 = [0 \ 1 \ 0 \ 0.2; 1 \ 0 \ 0 \ 0.1; 0 \ 0 \ 1 \ 0.3; 0 \ 0 \ 0 \ 1]$;)).

Tabla 3.9 Matriz transformación homogénea con puntos inicial y final.

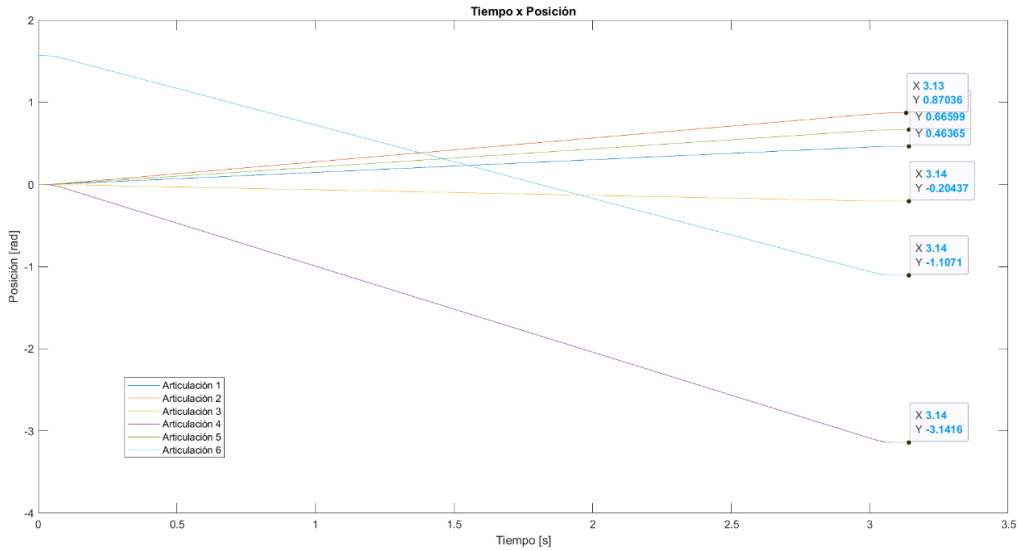
Matriz de transformación homogénea punto inicial (T1)	Matriz de transformación homogénea punto final (T2)
$\begin{bmatrix} 0 & 1 & 0 & 0.16 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0.464 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 & 1 & 0 & 0.2 \\ 1 & 0 & 0 & 0.1 \\ 0 & 0 & 1 & 0.3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$

Valores de la matriz homogénea utilizados para obtener las gráficas de la posición, velocidad y aceleración, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Posteriormente, se genera la cinemática inversa de los puntos T1 y T2 con la función ($q1 = \text{inversekinematic6}(T1, -1, 1)$; y $q2 = \text{inversekinematic6}(T2, -1, 1)$;)). La función $[t, \text{pos}, \text{vel}, \text{ace}] = \text{planificador}(q1, q2)$ (Anexo 4), permite obtener los valores del tiempo, posición, velocidad

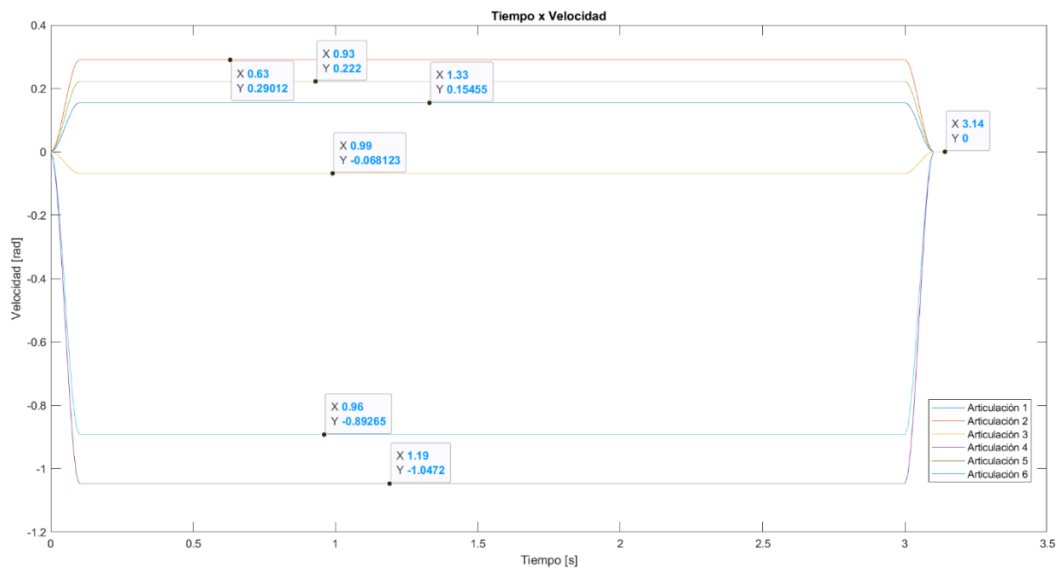
y aceleración la cual nos permitirá graficar el tiempo x posición Figura 3.6, tiempo x velocidad Figura 3.7 y tiempo x aceleración Figura 3.8.

Figura 3.6. Planificador de trayectoria Tiempo x Posición.



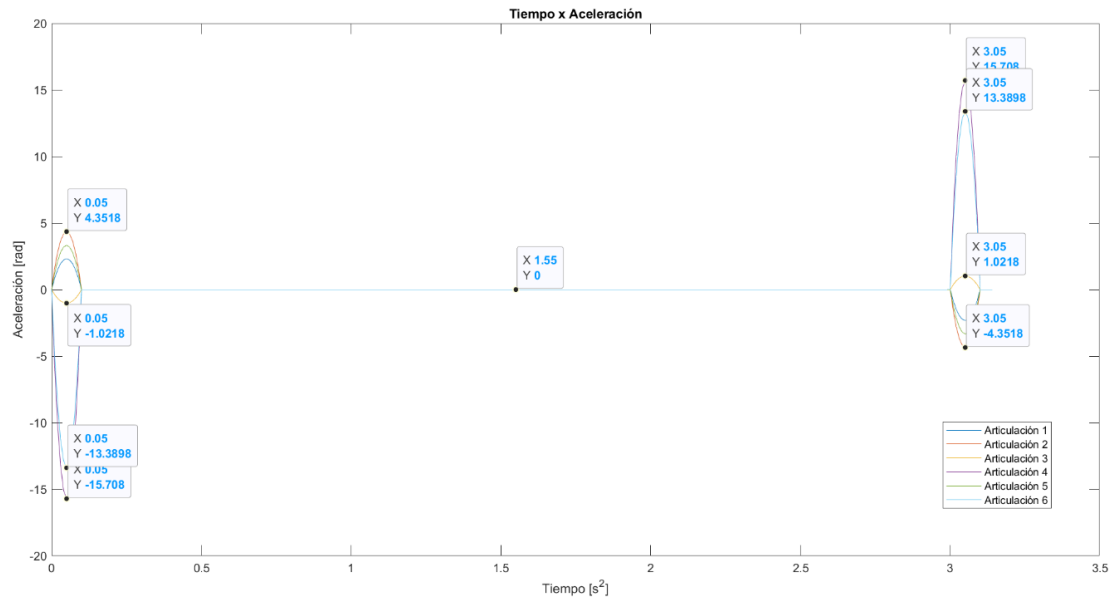
Gráfica tiempo x posición utilizando plot (t, pos), Autores: Jonathan Quilumbaquin, Isaac Reyes.

Figura 3.7. Planificador de trayectoria Tiempo x Velocidad.



Gráfica tiempo x velocidad utilizando plot (t, vel), Autores: Jonathan Quilumbaquin, Isaac Reyes.

Figura 3.8. Planificador de trayectoria Tiempo x Aceleración.



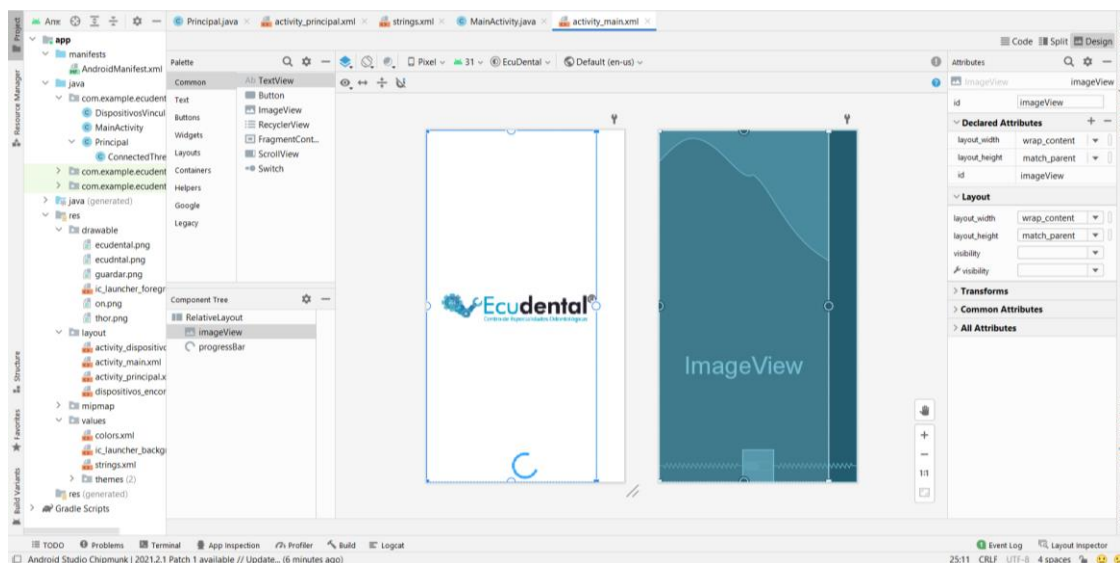
Gráfica tiempo x aceleración utilizando plot (t, acel), Autores: Jonathan Quilumbaquin, Isaac Reyes.

3.4 Código Fuente Android Studio

Con la ayuda de funciones previamente creadas resultó menos complejo la creación del entorno gráfico y funciones lógicas como las descritas a continuación.

En la Figura 3.9 se aprecia el entorno de creación de la APK.

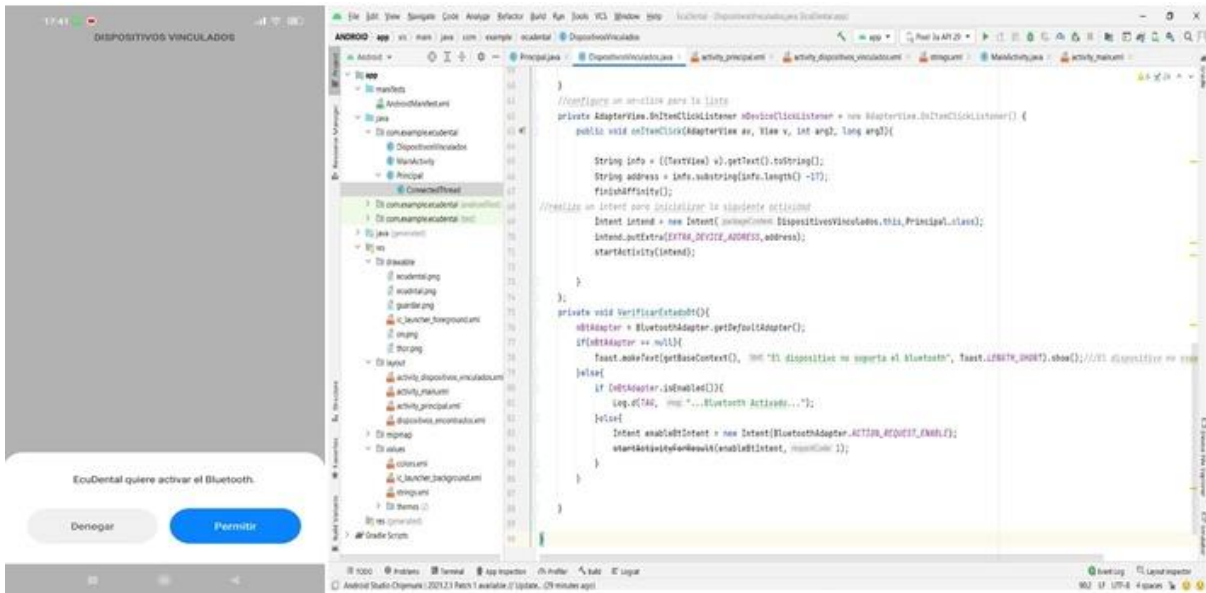
Figura 3.9. Interfaz inicio Android Studio.



Interfaz APK Android Studio, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.10 se muestra la codificación y la interfaz en el celular la cual permitirá la activación del BLUETOOTH para la correcta conexión.

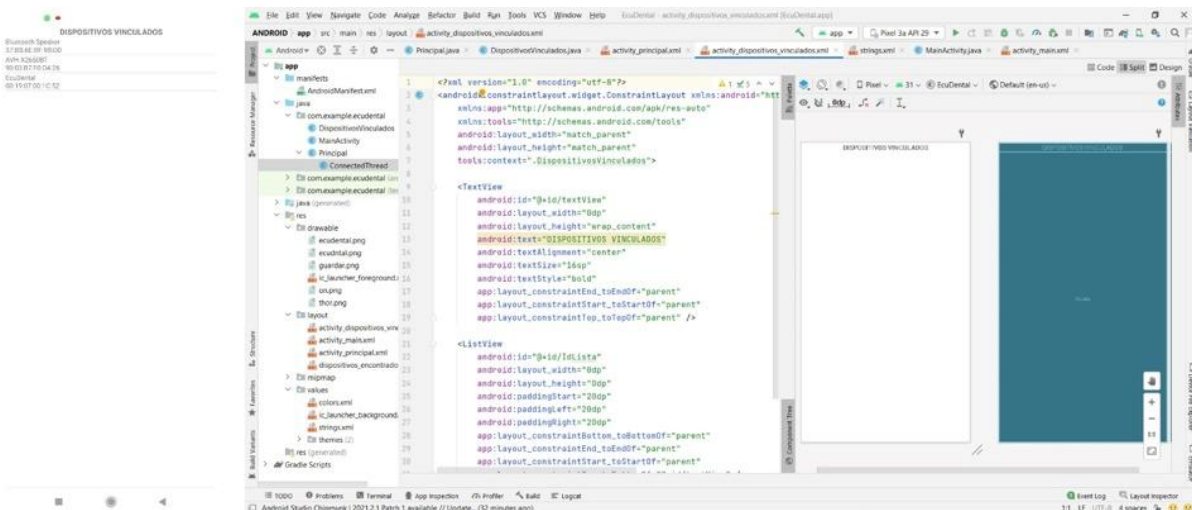
Figura 3.10. Interfaz Activar BLUETOOTH Android Studio.



Codificación activar BLUETOOTH, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.11 se muestra la codificación en Android Studio y los dispositivos vinculados que se encuentran en el teléfono.

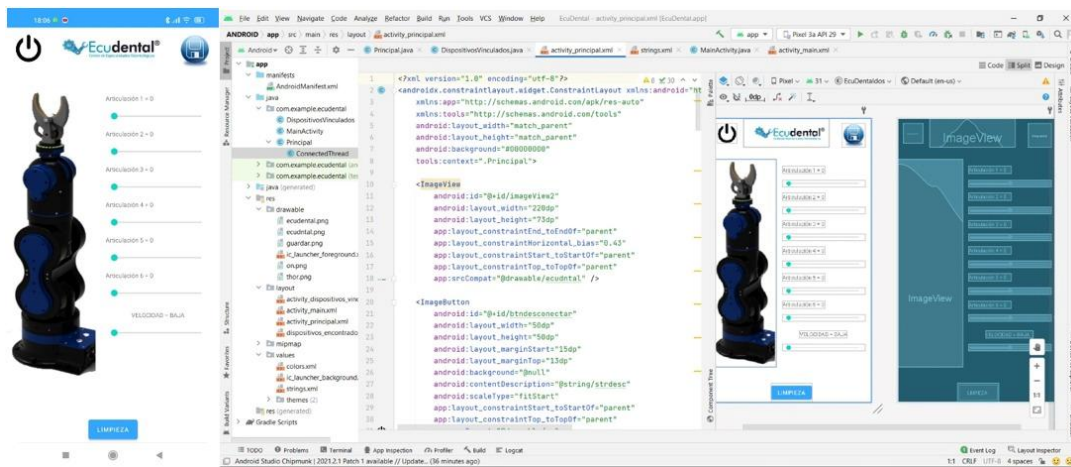
Figura 3.11. Interfaz Dispositivos vinculados Android Studio.



Codificación dispositivos vinculados, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.12 se muestra la codificación y el HMI para el control del asistente odontológico.

Figura 3.12. HMI Android Studio.



Codificación HMI, Autores: Jonathan Quilumbaquin, Isaac Reyes.

3.5 Código Fuente Arduino

Mediante el IDE de Arduino se realizó la programación de los microcontroladores para la comunicación y uso de estos; cabe recalcar que se repite el proceso de codificación para todas las articulaciones.

En la Figura 3.13 se declaran las salidas de la bomba succionadora, pasos y dirección de los motores Nema 17.

Figura 3.13. Declaración pines Arduino.

```
//DECLARACION PINES
int bomba=32;

//motor articulación 6
int steps6 = 31; //Posición del pin contador de pasos
int direccion6 = 29; //Posición del pin dirección

//motor articulación 5/2//PROBLEM
int steps52 = 27; //Posición del pin contador de pasos
int direccion52 = 25; //Posición del pin dirección

//motor articulación 5/1
int steps5 = 23; //Posición del pin contador de pasos
int direccion5 = 21; //Posición del pin dirección

//motor articulación 4
int steps4 = 17; //Posición del pin contador de pasos
int direccion4 = 15; //Posición del pin dirección
```

Declaración pines Arduino pasos y dirección, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.14 se define las variables tiempo de respuesta, número de pasos, movimientos guardados y los grados obtenidos de Android Studio.

Figura 3.14. Declaración de variables Arduino.

```
//ARTICULACIÓN 6
int pasos6 = 0;
String Grados6;
int GradosNumero6;
int movGuardado6 = 0;

//ARTICULACION 5
int pasos5 = 0;
int pasos52 = 0;
String Grados5;
int GradosNumero5;
int movGuardado5 = 0;

//ARTICULACION 4
int pasos4 = 0;
String Grados4;
int GradosNumero4;
int movGuardado4 = 0;
```

Inicialización variables tipo entero y String, Autores: Jonathan Quilumbaquin, Isaac Reyes.

La declaración de los pines de salida y la velocidad de los puertos seriales que se utilizaran para la comunicación entre módulo BLUETOOTH, NodeMCU y Arduino se observa en la Figura 3.15.

Figura 3.15. Declaración pines salida y puertos seriales Arduino.

```
void setup() {
  Serial.begin(9600);

  Serial1.begin(115200);

  pinMode(steps52, OUTPUT);
  pinMode(direccion52, OUTPUT);

  pinMode(steps6, OUTPUT);
  pinMode(direccion6, OUTPUT);
}
```

Declaración pines de salida para módulo A4988 y puertos seriales, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.16 se observa la inicialización del puerto serial y el tiempo de respuesta a los cuales va a actuar el asistente odontológico.

Figura 3.16. Codificación de tiempo de operación Arduino.

```

if (a == '0'){ //TIEMPO DE RESPUESTA BAJO
TIEMPO = 4000;
}
if (a == '1'){ //TIEMPO DE RESPUESTA MEDIO
TIEMPO = 3000;
}
if (a == '2'){ //TIEMPO DE RESPUESTA ALTO
TIEMPO = 2000;
}

```

Tiempo de respuesta Alto, Medio y Bajo, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.17 se observa la lectura de los seekbar de Android Studio los cuales nos permitirá el movimiento de las distintas articulaciones.

Figura 3.17. Codificación lectura SeekBar Android Studio - Arduino.

```

if (a == 'z'){ |
delay(10); // tiempo de retardo entre dato y dato
while (Serial.available()){
char b = Serial.read(); //guarda el dato de la seekbar en b
Grados6 = Grados6 + b; // suma los valores de la seekbar a grados
}
GradosNumero6 = Grados6.toInt();// convierte la cadena en entero

while (GradosNumero6>pasos6) //Equivale al número de vueltas (200 es 360° grados) o micropasos
{
pasos_der6();
pasos6 = pasos6 +1;
delayMicroseconds(VELOCIDAD); // Regula la velocidad, cuanto mas bajo mas velocidad.
}

while (GradosNumero6<pasos6) //Equivale al numero de vueltas (200 es 360° grados) o micropasos
{
pasos_izq6();
pasos6 = pasos6 - 1;
delayMicroseconds(VELOCIDAD); // Regula la velocidad, cuanto mas bajo mas velocidad.
}
Grados6= ""; // borra los datos de la seekbar
}//// FIN lectura SEEKBAR ARTICULACION 6 ///////////////////////////////////////////////////

```

Lectura Seekbar y movimiento de motores, Autores: Jonathan Quilumbaquin, Isaac Reyes.

La codificación para guardar las posiciones de cada una de las articulaciones se muestra en la Figura 3.18.

Figura 3.18. Codificación para guardar posiciones Arduino.

```
if (a == 'A') {
  movGuardado6 = GradosNumero6;
  movGuardado5 = GradosNumero5;
  movGuardado4 = GradosNumero4;
  movGuardado3 = GradosNumero3;
  movGuardado2 = GradosNumero2;
  movGuardado1 = GradosNumero1;
}
```

Guarda la posición de cada una de las articulaciones independientemente, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.19 se observa la comunicación serial1 la cual es la encargada de comunicarse con el módulo NodeMCU y a su vez la reproducción de las posiciones previamente guardadas.

Figura 3.19. Codificación para reproducir los movimientos Arduino.

```
if (Serial1.available () > 0) {

  char reproducir = Serial1.read();
  Serial1.println(reproducir);
  if (reproducir == 'H') {
    ////////// ARTICULACION 6 //////////
    while (movGuardado6 > pasos6) {
      pasos_der6();
      pasos6 = pasos6 + 1;
      delayMicroseconds(VELOCIDAD);
    }
    while (movGuardado6 < pasos6) {
      pasos_izq6();
      pasos6 = pasos6 - 1;
      delayMicroseconds(VELOCIDAD);
    }////////// FIN ARTICULACION 6 //////////
  }
}
```

Reproducción de los movimientos mediante voz, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Las funciones creadas para los movimientos de los motores en sentido horario y antihorario se muestran en la Figura 3.20.

Figura 3.20. Codificación sentido motores horario - antihorario.

```
void pasos_der6() {
  digitalWrite(direccion6, HIGH);
  digitalWrite(steps6, HIGH); // This LOW to HIGH change is what creates the
  digitalWrite(steps6, LOW); // al A4988 de avanzar una vez por cada pulso de energia.
}
void pasos_izq6() {
  digitalWrite(direccion6, LOW);
  digitalWrite(steps6, LOW); // LOW to HIGH hace que el motor avance ya que da la orden
  digitalWrite(steps6, HIGH); // al A4988 de avanzar una vez por cada pulso de energia.
}
```

Funciones creadas para el sentido de los motores, Autores: Jonathan Quilumbaquin, Isaac Reyes.

3.6 Código fuente Módulo NodeMCU ESP8266

Para realizar la comunicación entre Arduino y el módulo NodeMCU ESP8266 se debe establecer la red y la contraseña (Figura 3.21 A) a su vez la velocidad de comunicación de 115200 baudios (Figura 3.21 B) y en la (Figura 3.21 C) se observa el protocolo de comunicación.

Figura 3.21. Codificación módulo NodeMCU conexión Wifi.

```
const char* ssid = "ALTEC-GAMING";
const char* password = "Alt65u2c";
A

void setup() {
  Serial.begin(115200);
  B

  void loop(){
    ConectarWifi();
    espAlexa.loop();
    delay(1);}

    //////////////// FUNCION PARA CONECTAR EL
    void ConectarWifi() {
      if(WiFi.status() != WL_CONNECTED) {
        WiFi.mode(WIFI_STA);
        WiFi.begin(ssid, password);
        Serial.println("");
        Serial.println("Connecting to WiFi");
        while(WiFi.status() != WL_CONNECTED) {
          digitalWrite(D0, HIGH);
          delay(500);
          Serial.print(".");}
        Serial.print("Connected to ");
        Serial.println(ssid);
        Serial.print("IP address: ");
        Serial.println(WiFi.localIP());
        digitalWrite(D0, LOW);}}
    C
```

Figura A Contraseña y SSID, Figura B velocidad de comunicación y Figura C protocolo de comunicación de la red, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.22 se emiten los datos hacia el Arduino mega mediante la comunicación serial, esta función es la encargada de reproducir las posiciones del brazo y activar el succionador de saliva.

Figura 3.22. Codificación para envío de datos NodeMCU ESP8266.

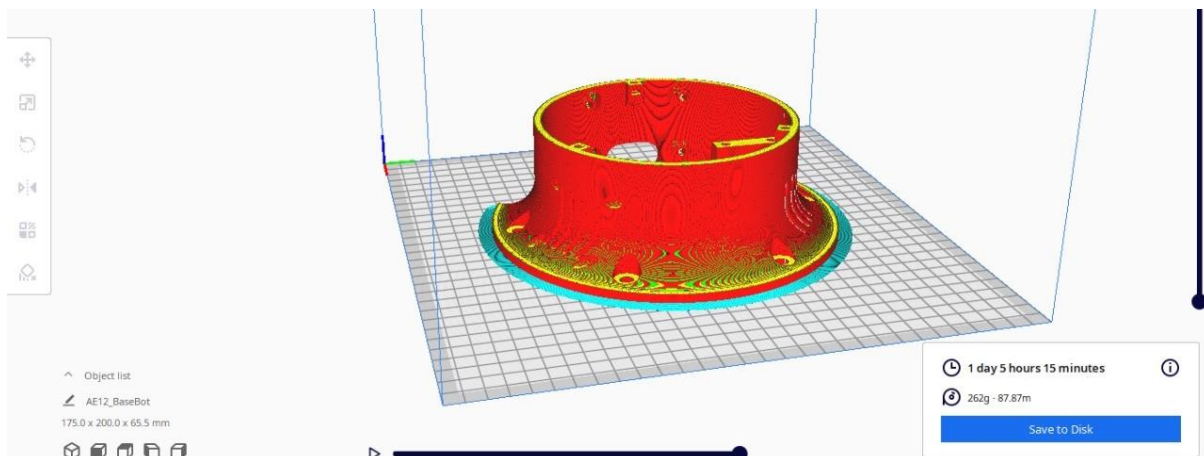
```
//////////////////// FUNCION PARA EL PIN D1 ////////////////////// POSICION INICIAL
void Funcion_D1(uint8_t brightness){
  if(brightness){Serial.write("H");
  }
}
//////////////////// FUNCION PARA EL PIN D2 ////////////////////// SUCCION
void Funcion_D2(uint8_t brightness){
  if(brightness){Serial.write("I");
  }
}
```

Envío datos para reproducir movimientos, Autores: Jonathan Quilumbaquin, Isaac Reyes.

3.7 Impresión 3D Brazo Thor

El diseño de este modelo consta de un total de 51 archivos stl, que son utilizados para la impresión del brazo robótico Thor, se debe tomar en cuenta la demanda de cada pieza y su función final para así modificar el relleno y el material a utilizar para la impresión del autómata como se muestra en la Figura 3.23.

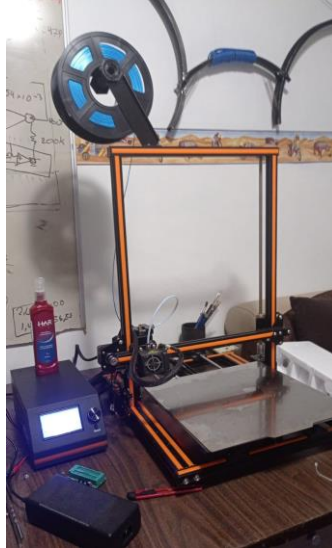
Figura 3.23. Modelado impresión 3D.



Modelado Ultimaker Cura, Autores: Jonathan Quilumbaquin, Isaac Reyes.

La impresora 3D que se utilizará es la Anet-E12 mostrada en la Figura 3.22, con una velocidad de 100 mm/s y una precisión de 0.1 mm, el tamaño máximo de la impresión es de 300*300*400mm.

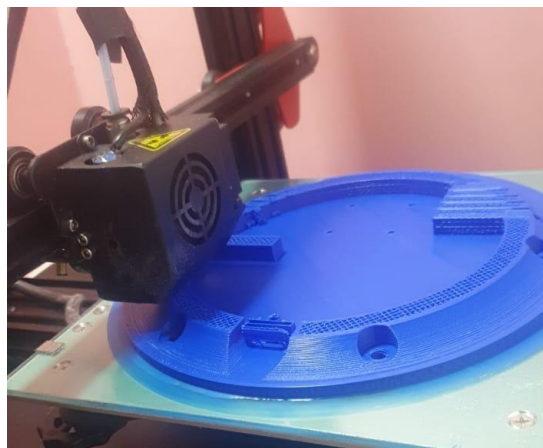
Figura 3.24. Impresora Anet-E12.



Impresora utilizada para las piezas, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Una vez seleccionado el material y el diseño 3D realizado para la base del brazo robótico THOR se procederá a dar inicio con su impresión como se muestra en la Figura 3.25.

Figura 3.25. Impresión pieza 3D.



Impresión 3D de la base del brazo robótico, Autores: Jonathan Quilumbaquin, Isaac Reyes.

3.8 Ensamble Brazo Thor

En la Figura 3.26 se observa las piezas terminadas y listas para ensamblar.

Figura 3.26. Piezas 3D terminadas.



Piezas listas para comenzar el ensamble, Autores: Jonathan Quilumbaquin, Isaac Reyes.

El armado del brazo se lo debe realizar por partes comenzado por la base Figura 3.27 para seguir con los movimientos de la muñeca, antebrazo, codo y hombro una vez todas las piezas estén armadas y funcionales se ensambla el brazo completo.

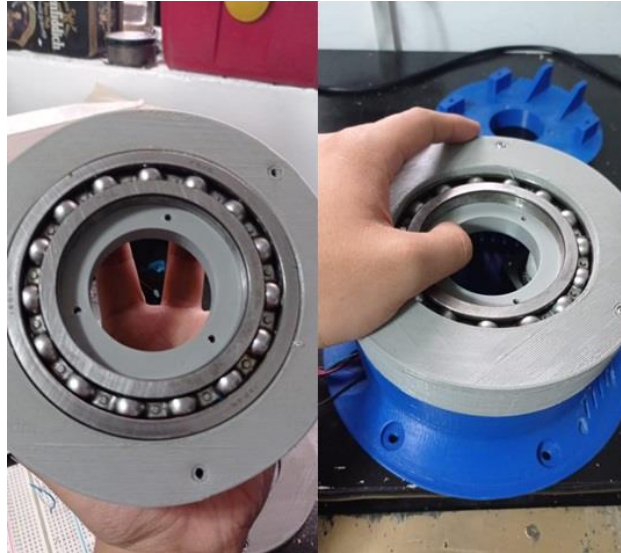
Figura 3.27. Ensamble base.



Fijación del motor y engrane articulación 1, Autores: Jonathan Quilumbaquin, Isaac Reyes.

El rulimán utilizado es el 16014ZZ el cual consta de una dimensión de 70x110x13 mm, el cual se sitúa justo entre la base y la parte giratoria; se coloca a presión para posteriormente ajustar con pernos a la base como se indica en la Figura 3.28.

Figura 3.28. Ensamble rulimán y pieza rotativa de la base.



Articulación 1, Autores: Jonathan Quilumbaquin, Isaac Reyes.

El hombro (articulación 2) es la responsable de toda la fuerza del brazo robótico, para su movimiento se emplean 2 motores de alto torque y engranes. Junto a esta articulación van los laterales en uno de ellos se pasan los cables y el otro contiene los engranes y rodamientos para ejercer el movimiento como se indica en la Figura 3.29.

Figura 3.29. Ensamble hombro.



Articulación 2, Autores: Jonathan Quilumbaquin, Isaac Reyes.

El codo (articulación 3) se mueve mediante una correa dentada que está acoplada a un engrane y un motor que genera el movimiento final como se muestra en la Figura 3.30.

Figura 3.30. Ensamble codo.



Articulación 3, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.31 se muestra como el engrane del antebrazo (articulación 4) debe ser bastante robusto porque lleva todo el peso de la manguera succionadora y los dos motores de la articulación 5-6.

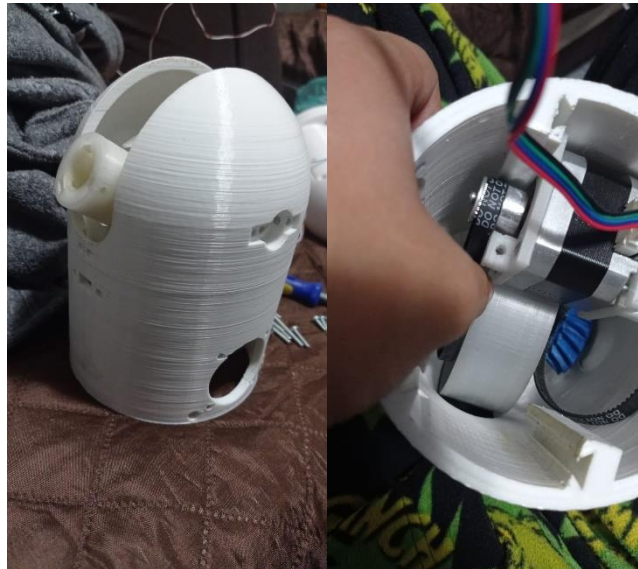
Figura 3.31. Ensamble antebrazo.



Articulación 4, Autores: Jonathan Quilumbaquin, Isaac Reyes.

La muñeca (articulación 5-6) consta de 2 movimientos (Pitch - Roll) los cuales son accionados por 2 motores; para el funcionamiento del movimiento Pitch (Figura 3.32 A) ambos motores giran al mismo sentido y para el movimiento Roll (Figura 3.32 B) los motores giran en un sentido contrario.

Figura 3.32. Ensamble muñeca.



A

B

Movimiento Pitch y Roll de la muñeca, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Una vez que todas las piezas se han armado por separado se procede a juntarlas para que trabajen adecuadamente, colocando los pernos y tuercas correspondientes como se muestra en la Figura 3.33.

Figura 3.33. Ensamble hombro, codo, antebrazo y muñeca.



Ensamble articulación 2, 3, 4, 5 y 6 Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.34 se observa el ensamble final del Hombro, codo, antebrazo y los movimientos de la muñeca (Pitch - Roll).

Figura 3.34. Ensamble final hombro, codo, antebrazo y muñeca.



Ensamble final articulación 2, 3, 4, 5 y 6 Autores: Jonathan Quilumbaquin, Isaac Reyes.

Para finalizar con el ensamble se procede a unir las piezas previamente mencionadas, con la base como se muestra en la Figura 3.35.

Figura 3.35. Ensamble brazo THOR Finalizado.



Ensamble brazo finalizado, Autores: Jonathan Quilumbaquin, Isaac Reyes.

3.9 Circuito Electrónico

3.9.1 Diseño Circuito Electrónico

La electrónica requerida para este proyecto fue previamente implementada en un protoboard, una vez que todo se probó se pudo empezar con el diseño de la placa del circuito impreso, para esto se empleó el software EasyEDA.

El diseño esquemático del circuito consta de los siguientes componentes:

Tabla 3.10 Elementos electrónicos del circuito esquemático.

Descripción	Cantidad
Driver A4988	4
Capacitor 10 μ F 50v	7
Driver DRV 8825	3
Conectores 4 pines	7
Bornera	1
Módulo BLUETOOTH HC-05	1
Arduino Mega PRO-2560	1
Módulo NodeMCU	1
Dip Switch 4	1

Elementos utilizados para el diseño del circuito esquemático, Autores: Jonathan Quilumbaquin, Isaac Reyes.

El esquema electrónico final se muestra en el Anexo 6 de igual forma en la Figura 3.36 se observa el ruteo y diseño PCB de tamaño 114x200 mm el cual será utilizado para el grabado en baquelita.

Figura 3.36. Diagrama diseño PCB.

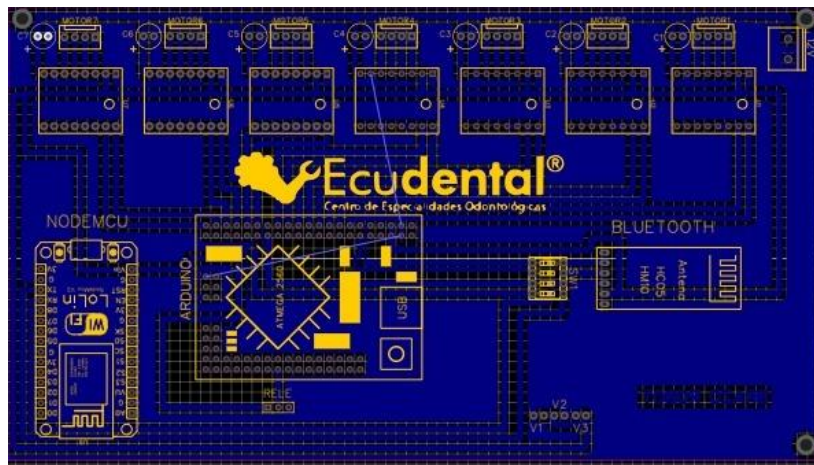


Diagrama para grabado PCB finalizada Autores: Jonathan Quilumbaquin, Isaac Reyes.

Es importante conocer que corriente va a pasar por las pistas para dimensionar el grosor de esta, si solo se requiere transmitir datos la pista puede ser muy delgada, pero si se requiere manejar corrientes más altas es necesario hacer más gruesa la pista. EasyEDA nos permite generar un archivo en 3D como se observa en la Figura 3.37 el cual nos permite visualizar como quedaría nuestro circuito ya implementado.

Figura 3.37. Visualización 3D.

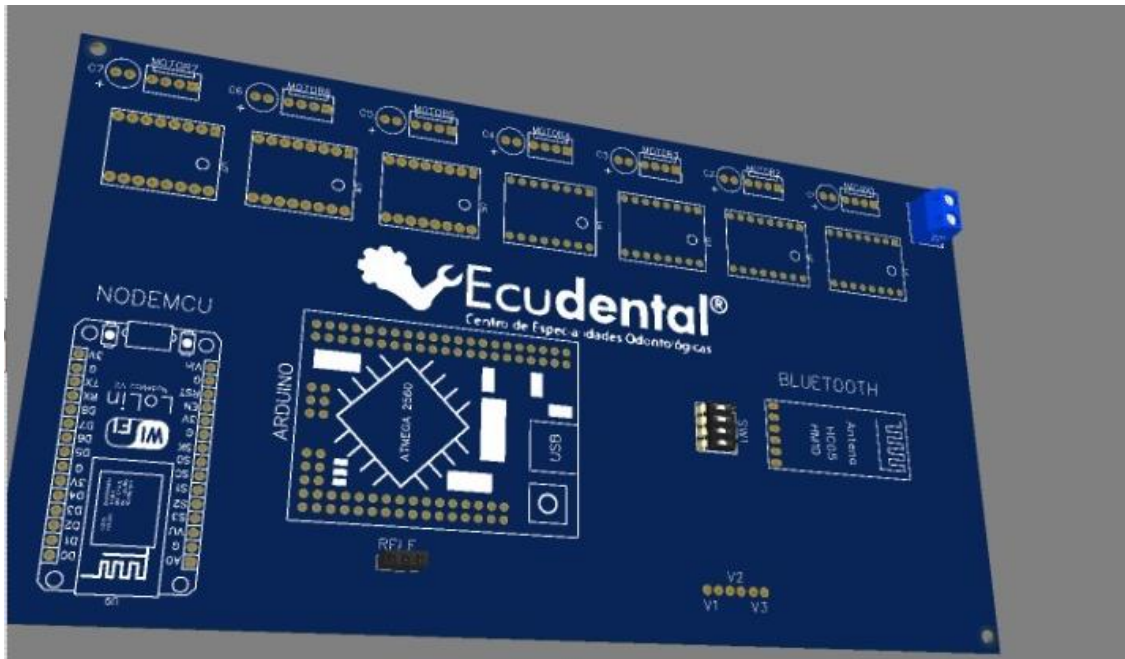


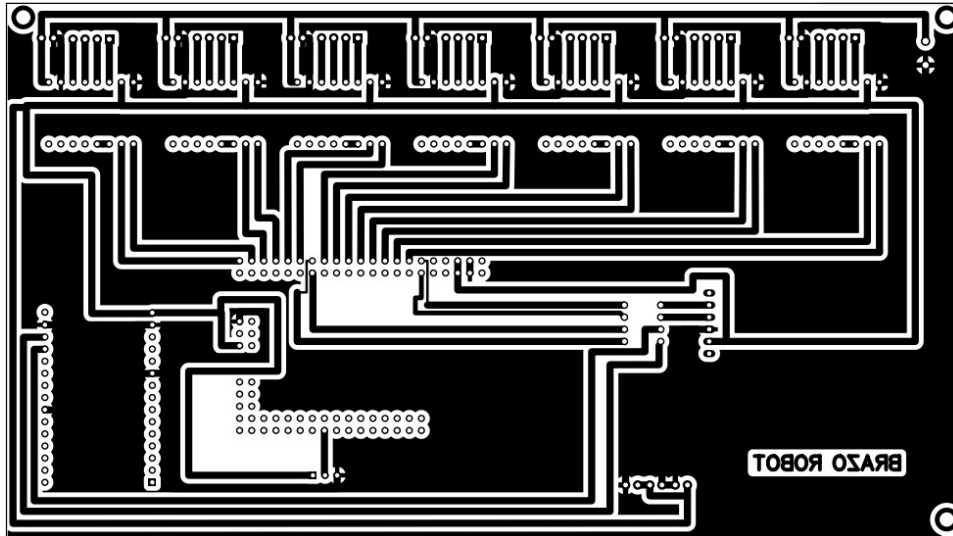
Diagrama 3D, Autores: Jonathan Quilumbaquin, Isaac Reyes.

3.9.2 Grabado circuito electrónico

El grabado del circuito se realiza mediante el método de transferencia por acetona, es fundamental que la impresión Figura 3.38, se haga en una impresora láser en papel de revista de 15g, la calidad de la impresión se debe fijar en la más alta posible para evitar la pérdida de pistas pequeñas.

Se debe tomar en cuenta que la baquelita previamente debe ser limpiada con alcohol y evitar el contacto del papel con los dedos.

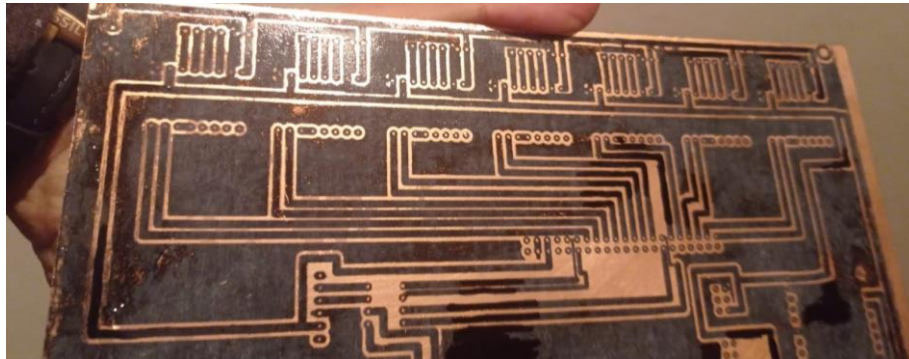
Figura 3.38. Impresión PCB.



Impresión PCB, Autores: Jonathan Quilumbaquin, Isaac Reyes.

A continuación, en la Figura 3.39, se procede a transferir la impresión a la baquelita con el uso de acetona y corregir con marcador los puntos que no se transfirieron de manera correcta.

Figura 3.39. Transferencia impresión PCB.

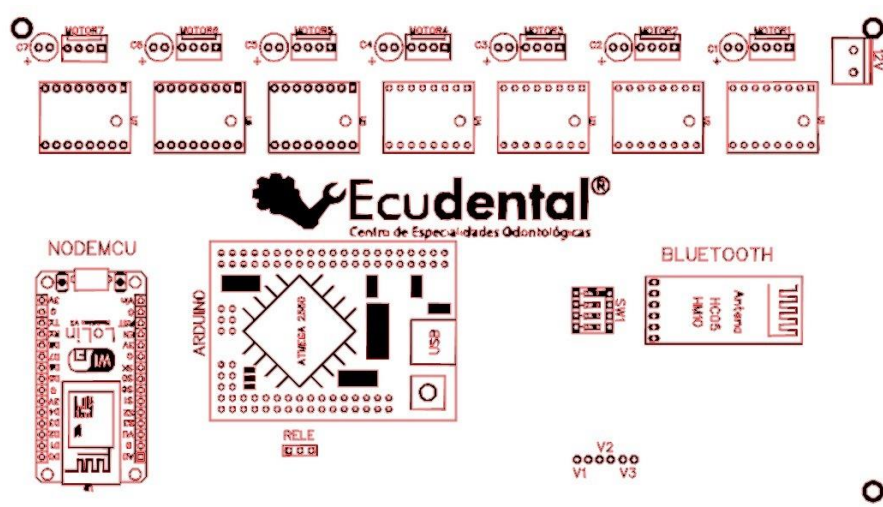


Transferencia impresión PCB - baquelita, Autores: Jonathan Quilumbaquin, Isaac Reyes.

3.9.3 Grabado láser

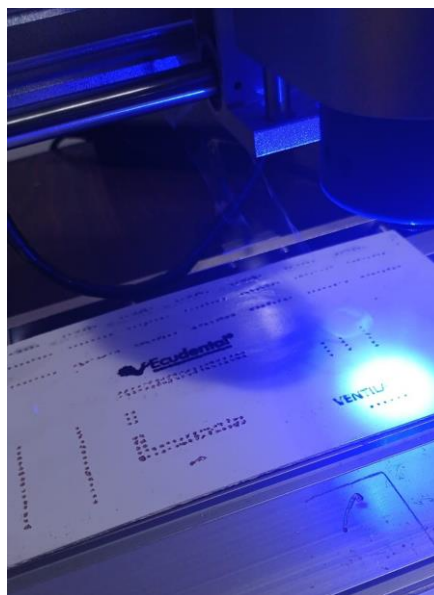
Para la capa superior se utilizó grabado laser en un CNC 3018max, con un láser cuya potencia es de 5 mW, el software utilizado es LaserGRLB que es de libre uso. En él se configura el área a trabajar y la potencia del láser en la Figura 3.40 se muestra resultado del grabado en el software y en la 3.41 el terminado.

Figura 3.40. Interfaz software Grabado láser.



Configuración grabado láser componentes baquelita, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Figura 3.41. Grabado láser.

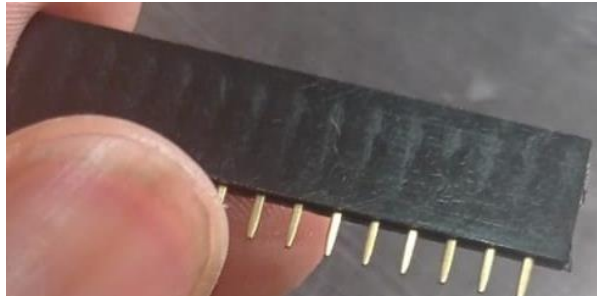


Serigrafía parte superior, Autores: Jonathan Quilumbaquin, Isaac Reyes.

3.9.4 Implementación circuito electrónico final

Una vez la placa electrónica está culminada es momento de montar los elementos electrónicos, se recomienda utilizar espadines tipo hembra Figura 3.42, para que en un futuro se facilite el mantenimiento preventivo y correctivo del asistente dental.

Figura 3.42. Espadines tipo Hembra.



Espadines utilizados para drivers, Arduino mega y Esp8266, Autores: Jonathan Quilumbaquin, Isaac Reyes.

La suelda ideal para este proceso debe tener proporciones de 60 % estaño y 40 % plomo, la temperatura adecuada para estas condiciones es de 380 °C.

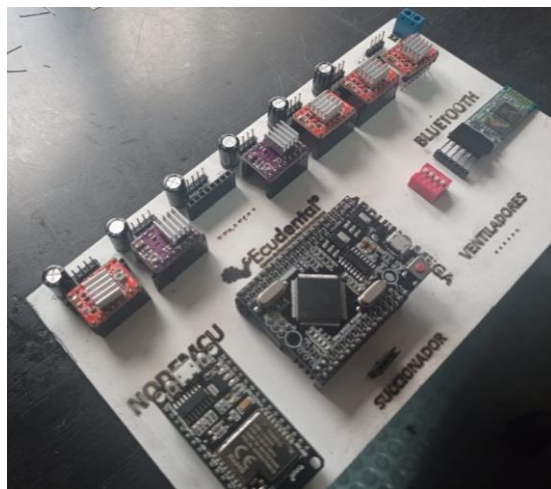
Una vez establecida las condiciones de estaño y temperatura se procede a soldar los componentes.

Se debe evitar en el proceso de soldadura los siguientes puntos:

- Suelda fría.
- No conductividad entre los puntos.
- Sueldas débiles.
- Cortocircuitos entre pistas.
- No adherencia de estaño en el cobre.

Para finalizar con la placa electrónica se debe montar los drivers y microcontroladores, alineándolos de manera uniforme para que se vean estéticamente bien.

Figura 3.43. Circuito electrónico final.



Montaje elementos electrónicos, Autores: Jonathan Quilumbaquin, Isaac Reyes.

3.10 Ensamble Asistente Dental

Después de culminar los procesos anteriores 3.8 y 3.9 se procede con el ensamble del asistente dental.

Compuesto de varias piezas tales como:

- Mesa.
- Bomba succionadora.
- Fuente voltaje.
- Circuito de control.
- Brazo mecánico.
- Tomacorriente AC.
- Amazon Alexa.
- Caja de componentes electrónicos.
- Acople manguera succionadora.

Estos se detallarán a continuación.

Es importante que el brazo robótico tenga un soporte el cual permita que el mismo se dirija a realizar el proceso odontológico establecido. Debido a sus condiciones físicas (tamaño y peso) es muy fácil incluir a la mesa en el consultorio dental.

El diseño de la mesa (Figura 3.44 A) se realizó en el software AutoCAD tomando en cuenta las dimensiones del brazo robótico, la caja de componentes electrónicos y la bomba de succión. En la (Figura 3.44 B) se puede ver la mesa finalizada.

Figura 3.44. Mesa para el brazo robótico.

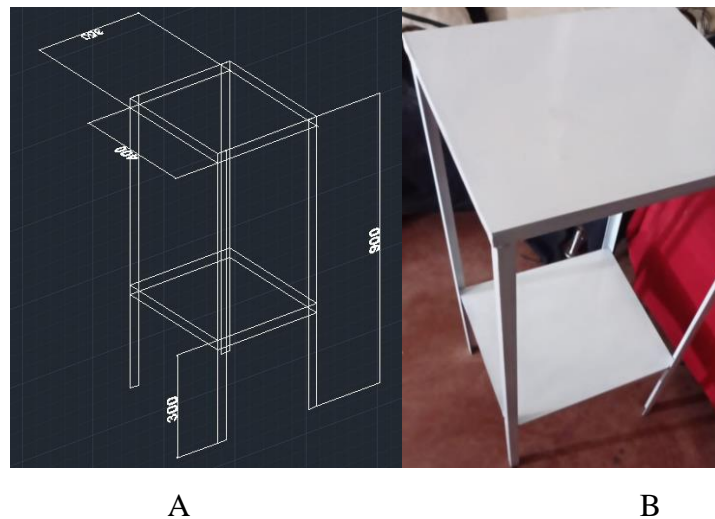


Figura A dimensiones, Figura B mesa finalizada, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.45 se muestra la bomba de succión la cual funciona a 110V, tiene una capacidad de 1000ml y un rango de presión de succión de 700 hPa a 1060hPa. Además, tiene una manguera con un largo de 2m que es suficiente para el alcance del autómeta a la boca del paciente.

El peso de la bomba ayuda a dar un centro de gravedad y estabilidad óptimo a la mesa.

Figura 3.45. Bomba succión.



Implementación bomba succión, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Para la alimentación del proyecto entero se utilizó una fuente conmutada de 12V y 10A y un adaptador de 5V para la parte lógica como se muestra en la Figura 3.46, de esta manera garantizamos que ambas fuentes trabajen por separado y no causen ningún inconveniente al momento de actuar la parte de control y comunicaciones con la parte de fuerza.

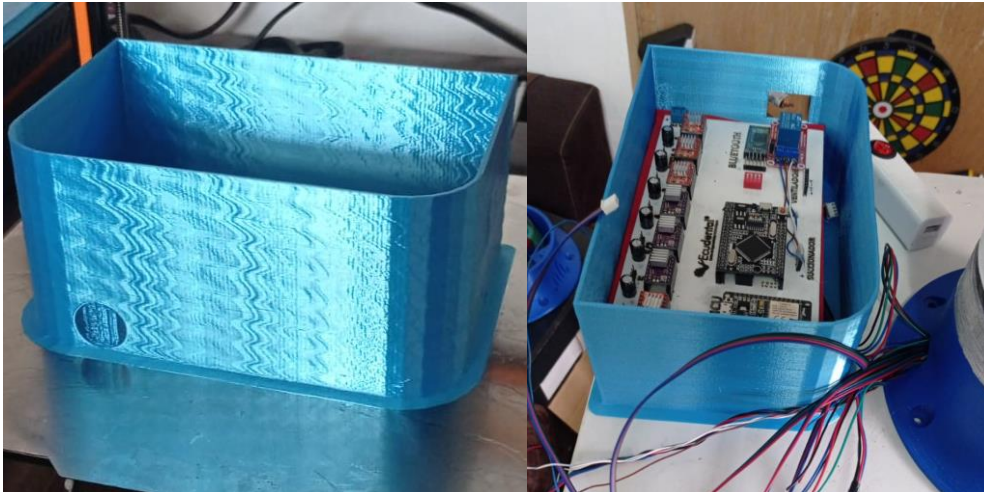
Figura 3.46. Fuentes de voltaje.



Fuentes de voltaje utilizados para el control del brazo robótico, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.47 se muestra la fijación de la caja electrónica la cual contiene la fuente de 12V, 5V y el circuito electrónico final.

Figura 3.47. Caja electrónica.



Montaje de fuentes de voltaje y circuito electrónico, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.48 se muestra la sujeción del brazo a la mesa para lo cual se utilizó pernos y tuercas de diámetro m5.

Figura 3.48. Brazo mecánico.



Sujeción brazo mecánico a la mesa, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Se realizó una conexión adicional de 110V para el dispositivo de reconocimiento de voz y a su vez que sea útil para realizar cualquier conexión de otro equipo médico que se ocupe en el proceso dental como se muestra en la Figura 3.49.

Figura 3.49. Tomacorriente AC.



Sujeción tomacorriente a la mesa, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 3.50 se aprecia el dispositivo de reconocimiento de voz de Amazon este es el que ayuda al proceso de succión y movimiento del asistente dental y está colocado en una zona cercana al operario para un óptimo trabajo.

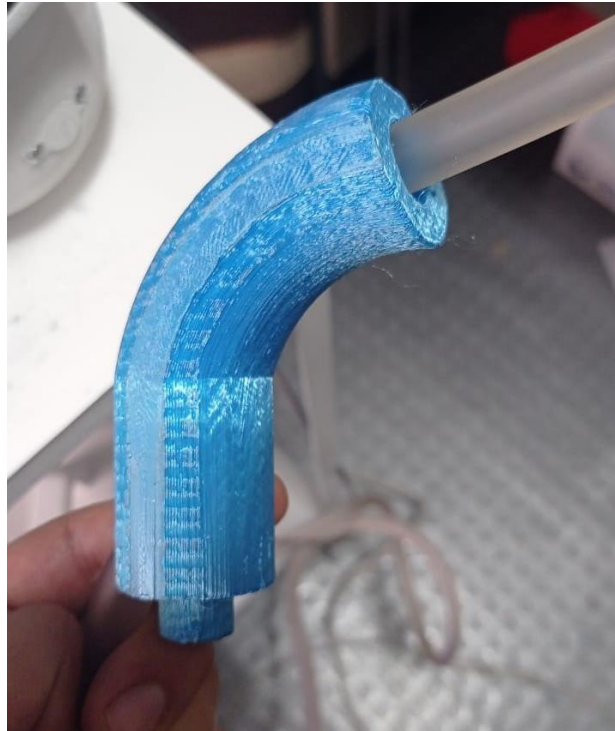
Figura 3.50. Amazon Alexa.



Amazon Alexa incluida en el prototipo, Autores: Jonathan Quilumbaquin, Isaac Reyes.

La finalidad del acople es sujetar la manguera de succión Figura 3.51, este fue previamente diseñado en AutoCAD y las dimensiones son de 9 cm de alto x 3cm de diámetro externo y 1.5 cm de diámetro interno.

Figura 3.51. Acople manguera.



Acople manguera succión, Autores: Jonathan Quilumbaquin, Isaac Reyes.

El montaje del acople en el brazo se lo realiza mediante pernos los cuales van a funcionar como prisioneros evitando que se mueva la pieza como se muestra en la Figura 3.52.

Figura 3.52. Montaje acople brazo.



Sujeción mediante pernos, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Finalmente, en la Figura 3.53 se muestra todo el sistema acoplado.

Figura 3.53. Sistema final.



Sistema final terminado, Autores: Jonathan Quilumbaquin, Isaac Reyes

CAPÍTULO 4

Pruebas y Resultados

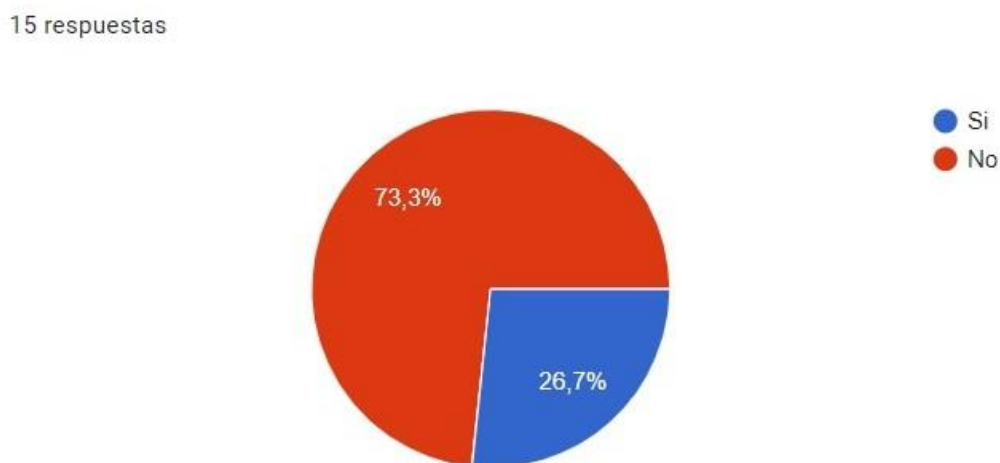
En este capítulo se detalla el análisis y los resultados de las encuestas realizadas al personal odontológico, de la misma manera las pruebas de comunicación serial entre el (Brazo robótico, BLUETOOTH y NodeMCU); a su vez la respuesta generada por los comandos de voz en un ambiente controlado y no controlado.

4.1 Encuestas al personal odontológico

Para determinar la viabilidad del proyecto se ejecutó una encuesta con el fin de conocer la existencia de este tipo de proyectos aplicados en el sector odontológico. Como resultado preliminar, no existen trabajos previos que corroboren con la ejecución del proyecto en mención; por lo que la encuesta realizada a expertos odontológicos permitió analizar el porcentaje de aceptación del asistente dental. La encuesta se realizó el 23 de mayo del 2021 a una población de 15 personas dando como resultado las siguientes gráficas.

La Figura 4.1 responde a la pregunta. ¿Está usted familiarizado con la robótica? El 73.3% de encuestados no está familiarizado con la robótica el otro 26.7% si está familiarizado.

Figura 4.1. Pregunta 1.

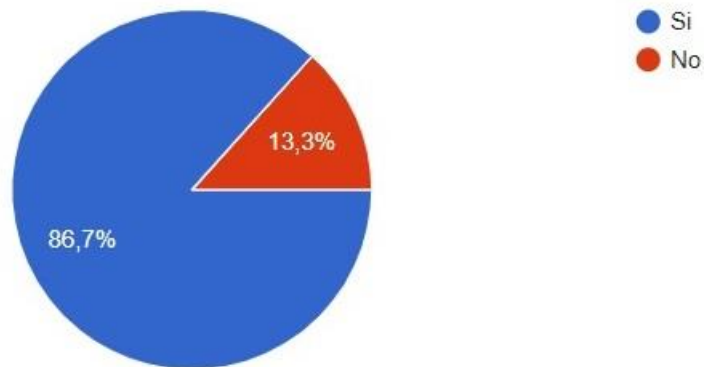


Respuesta a la pregunta 1, Autores: Jonathan Quilumbaquin, Isaac Reyes.

La Figura 4.2 responde a la pregunta ¿Considera usted confortable el tener dentro de un quirófano un asistente robótico que ayude en el proceso de succión que funcione mediante comandos de voz? El 86.7% respondió que si mientras que el 13.3% que no.

Figura 4.2. Pregunta 2.

15 respuestas

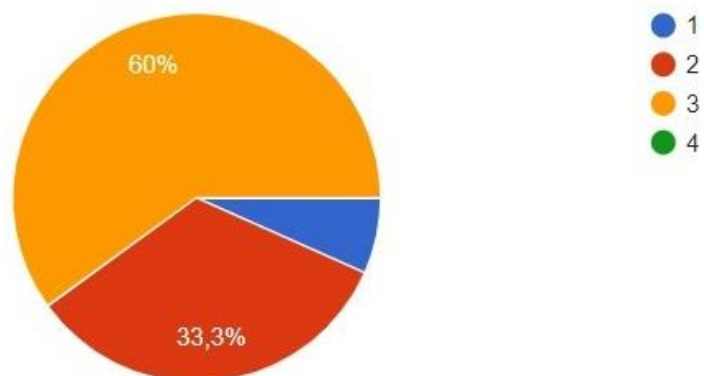


Respuesta a la pregunta 2, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Idealmente cuantas personas deben estar en una sala quirúrgica (sin contar el paciente), fue la pregunta número 3 y su resultado se presenta en la Figura 4.3. El 60% de encuestados respondió que es ideal trabajar con 3 personas, el 33.3% con 2 personas y el 7.7% respondió que solo el cirujano debe estar presente en las intervenciones quirúrgicas.

Figura 4.3. Pregunta 3.

15 respuestas

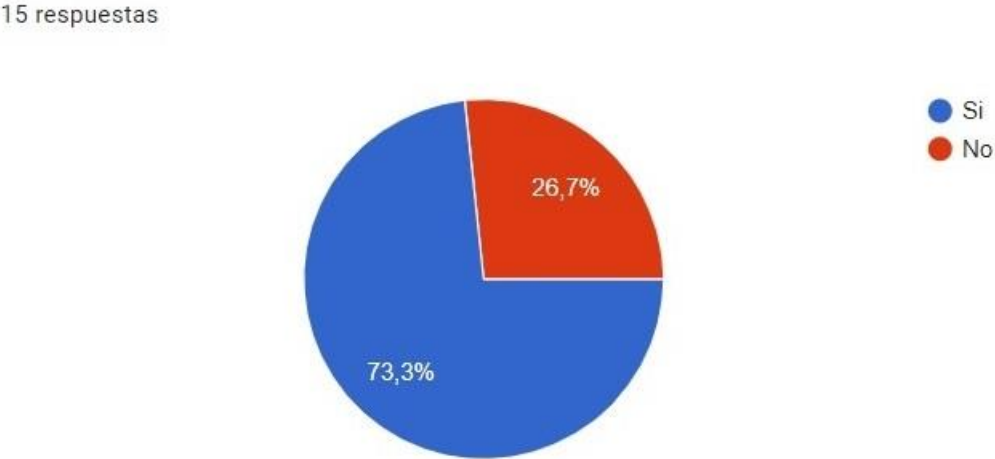


Color azul 1 persona, color rojo 2 personas, color naranja 3 personas, color verde 4 personas,

Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la pregunta 4 se consultó: Considera fastidioso el tener que cambiar de baterías a este robot; El 73.3% respondió que es fastidioso cambiar las baterías del robot mientras que para el 26.7% no lo es, este resultado se presenta en la Figura 4.4.

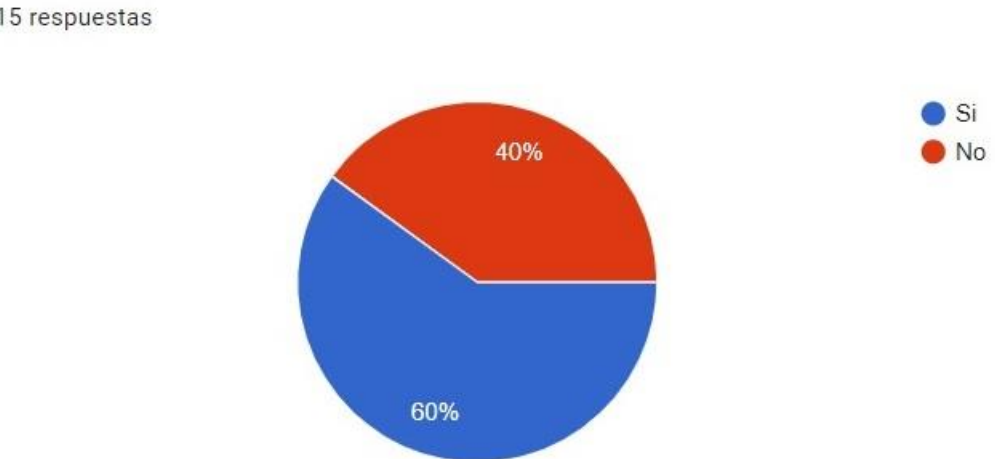
Figura 4.4. Pregunta 4.



Respuesta a la pregunta 4, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Cree que la vida del paciente puede tener riesgo al utilizar esta tecnología, fue la pregunta número 5 y en la Figura 4.5, el 60% de encuestados respondió que si tiene riesgo el otro 40% respondió que no.

Figura 4.5. Pregunta 5.

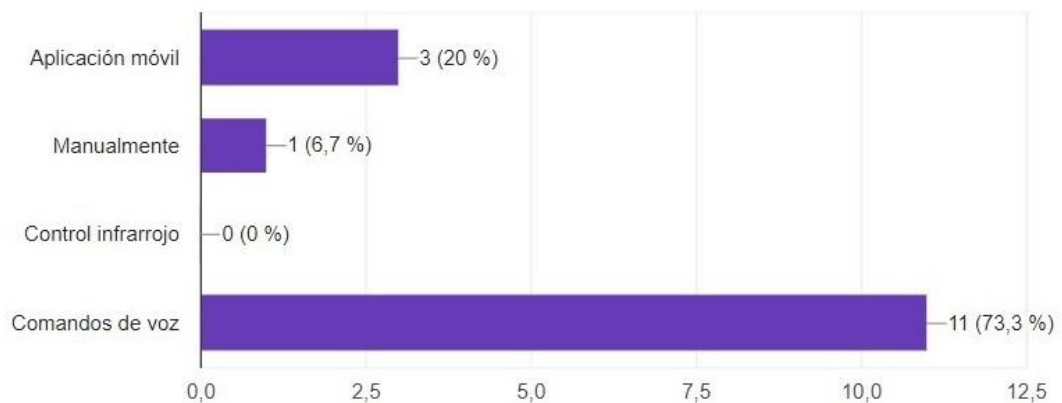


Respuesta a la pregunta 5, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la última pregunta se consultó: Como considera más eficiente la utilización y manipulación del mismo. El 73.3% de encuestados respondió que la manipulación del asistente debe ser mediante comandos de voz, el 20% mediante una aplicación móvil y el 6.7% mediante control manual, esta última respuesta se la puede evidenciar en la Figura 4.6.

Figura 4.6. Pregunta 6.

15 respuestas



Respuesta a la pregunta 6, Autores: Jonathan Quilumbaquin, Isaac Reyes.

La encuesta realizada a expertos en el área odontológica demuestra que el proyecto tiene una buena acogida ya que 13 de 15 odontólogos respondieron que es muy conveniente el uso del asistente robótico controlado mediante voz dentro del quirófano, teniendo como ventaja el tiempo, la comodidad y la salubridad del paciente.

Tras haber encuestado a profesionales de la rama se pudo obtener resultados precisos que apuntan y filtran las respuestas para orientarnos mejor en la investigación, cada una de las preguntas fue formulada para indagar temas específicos y desconocidos, este método se empleó ya que no existen previas investigaciones del mismo, como tal el realizar la encuesta dejó claro que:

- Es factible implementar este prototipo,
- No se debe incluir baterías ya que resulta molesto cambiarlas, por ello se opta conectar el prototipo a la red eléctrica.
- La mejor forma de manipular el prototipo es mediante comandos de voz ya que implica mayor facilidad al momento del manejo del mismo. Haciendo del proceso de succión algo más cómodo y eficiente.

4.2 Pruebas de funcionamiento

4.2.1 Energización del Asistente Dental

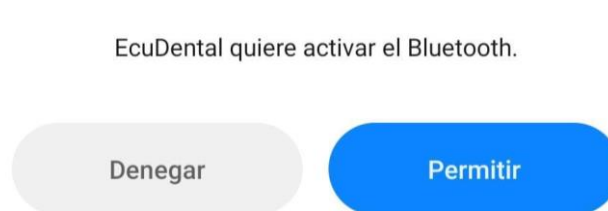
Al energizar el asistente odontológico, se debe comprobar que se encienda los indicadores de la fuente de voltaje 12V y 5V, módulo BLUETOOTH, Arduino mega mini Pro, módulo NodeMCU y las articulaciones se pongan rígidas es importante revisar está energización para un óptimo funcionamiento del asistente. (Anexo 5).

4.2.2 Comunicación serial BLUETOOTH

Se requiere implementar un sistema de comunicación maestro - esclavo para el movimiento de las articulaciones del brazo robótico.

La comunicación serial entre el dispositivo móvil (maestro) y el módulo BLUETOOTH (esclavo) se realiza a la velocidad estándar de (9600 Baudios). En la Figura 4.7 se muestra la solicitud de la aplicación para encender el BLUETOOTH del dispositivo móvil; iniciando la comunicación serial.

Figura 4.7. Solicitud aplicación.

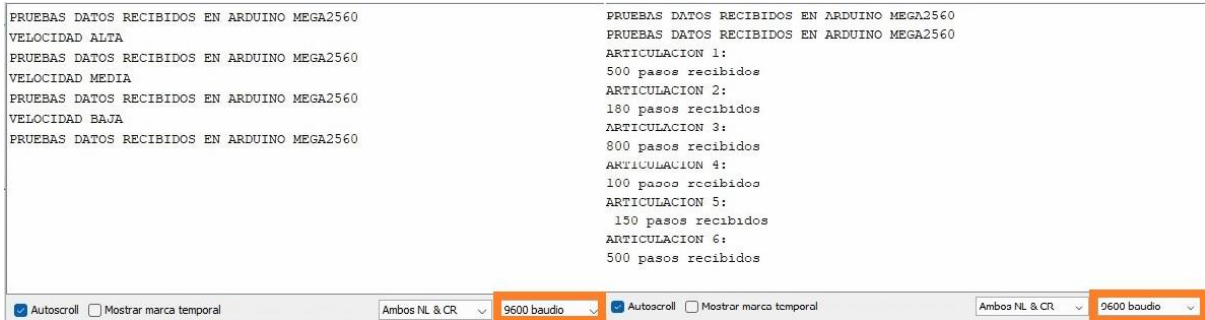


Solicitud activar BLUETOOTH aplicación EcuDental, Autores: Jonathan Quilumbaquin, Isaac Reyes.

La transmisión utilizada en el proyecto técnico es la Half-dúplex ya que solo permite transmitir o recibir datos, una sola función a la vez.

En el monitor serial del Ide de Arduino se observa la velocidad (baja, media o alta) (Figura 4.8 A) a la que va a responder el asistente dental y el posicionamiento (Figura 4.8 B) de todas las articulaciones (pasos de los motores).

Figura 4.8. Monitor serial comunicación NodeMcu.



A

B

Figura A velocidad, Figura B posicionamiento, Autores: Jonathan Quilumbaquin, Isaac Reyes.

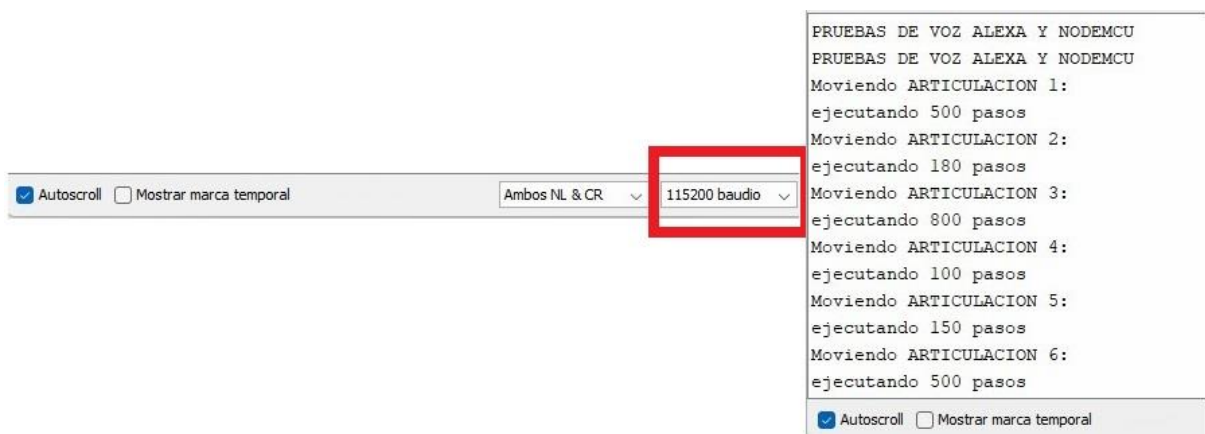
4.2.3 Comunicación serial NodeMCU

Se realizaron varias pruebas de comunicación con el módulo NodeMCU a diferentes velocidades como 9600, 28800, 57600 y 115200 (baudios) siendo está ultima la más adecuada (Figura 4.9 A), debido a que la recepción de datos debe ser alta y precisa.

Las pruebas realizadas con una velocidad de comunicación menor tuvieron pérdida de datos y como consecuencia una mala respuesta del asistente odontológico.

En la (Figura 4.9 B) se muestra la reproducción de los movimientos guardados mediante la aplicación Ecdental.

Figura 4.9. Monitor serial comunicación BLUETOOTH.



A

B

Figura A velocidad de comunicación, Figura B reproducción de movimientos, Autores: Jonathan Quilumbaquin, Isaac Reyes.

4.3 Pruebas de funcionamiento en un ambiente controlado

Estas pruebas se las realizaron una vez ensamblado todo el prototipo con el fin de determinar posibles errores de construcción y funcionamiento.

4.3.1 Movimiento de las articulaciones

Para realizar la comprobación de movimientos de las articulaciones es necesario probarlas por separado, para ello se utiliza la comunicación de la aplicación móvil con el brazo robótico la cual envía los movimientos requeridos por el usuario.

Se requiere ser meticuloso en cada una de estas pruebas y estar atento a que no exista pérdida de pasos, por las siguientes razones:

- Calibración errónea del driver.
- Falla mecánica de engranes.

En la Figura 4.10 se muestra la posición donde se empezarán las pruebas.

Figura 4.10. Posición inicial.



Posición de inicio para pruebas, Autores: Jonathan Quilumbaquin, Isaac Reyes.

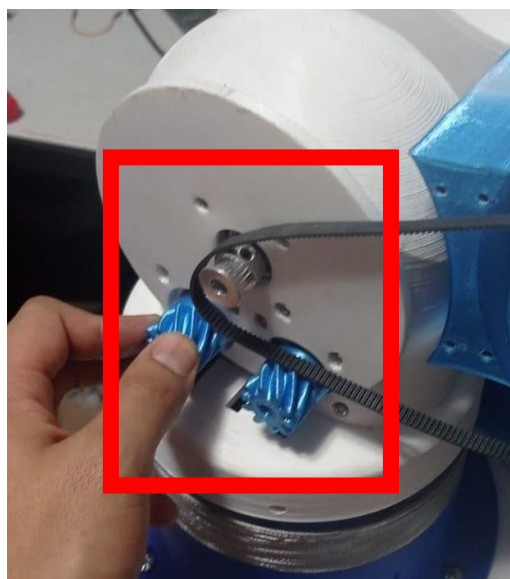
Al finalizar las pruebas Figura 4.11, se observó que utilizando un bajo tiempo de respuesta en el movimiento se obtiene mayor precisión en su funcionamiento.

Figura 4.11. Posición final de pruebas.



Movimiento brazo robótico, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Figura 4.12. Engranajes hombro.



En esta figura se aprecia los engranes sobrecalentados, Autores: Jonathan Quilumbaquin, Isaac Reyes.

4.3.2 Respuesta de Alexa a los comandos

Los comandos de voz utilizados con Alexa fueron analizados según el proceso que va a realizar el odontólogo. Para ello se utilizaron palabras fáciles y concretas tales como:

- Inicio: Regresa el brazo a la posición original.
- Succionar: Mueve el brazo a la posición guardada y enciende la bomba de succión.
- Finalizar: Apaga la bomba de succión y se ubica en una posición cercana al proceso de succión.
- Seguro: Se retira completamente del área de trabajo con una velocidad mayor y segura a las establecidas en la aplicación.

Se realizaron 56 pruebas de los comandos en una habitación cerrada (sin ruido externo), las hicieron hombres y mujeres, obteniendo un 89.28% de eficiencia, es decir reconoció correctamente 50 palabras de las 56 probadas, con esto se demuestra que no importa el tono de voz al momento de utilizar el asistente.

4.3.3 Respuesta del Asistente odontológico

Una vez comprobados los puntos 4.3.1 y 4.3.2 se puede realizar la prueba en conjunto. Obteniendo como resultado que el asistente guarda los movimientos de la aplicación y responde a los comandos de voz de manera eficiente, la velocidad de respuesta del prototipo dependerá de la calidad y de la conexión a internet.

Figura 4.13. Asistente odontológico.



Asistente odontológico en funcionamiento, Autores: Jonathan Quilumbaquin, Isaac Reyes.

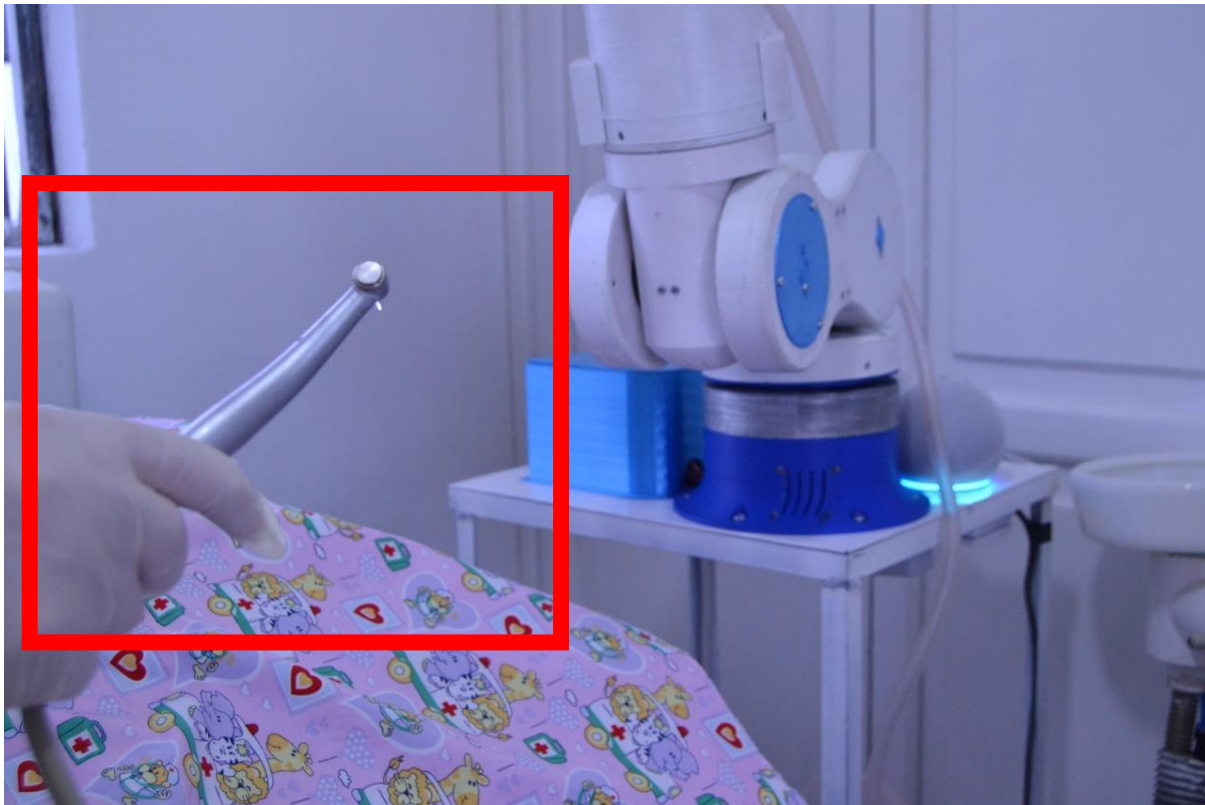
4.4 Pruebas en ambiente no controlado (Consultorio odontológico)

Se considera como ambiente no controlado el área de trabajo dental; se realizaron pruebas en un consultorio cercano ya que no se pudo probar directamente en la clínica Ecudental debido a condiciones de fuerza mayor y con la autorización de la clínica en mención.

4.4.3 Respuesta a los comandos de voz

En un ambiente no controlado los comandos de voz se pueden ver afectados por factores externos tales como: instrumentos odontológicos y ruido generado en el consultorio (sonido dispositivo móvil, voz del paciente, etc.); se realizaron las mismas 56 pruebas del punto 4.3.2 pero esta vez con un taladro dental encendido como el de la Figura 4.14 obteniendo un resultado porcentual de precisión del 82.14%.

Figura 4.14. Prueba taladro dental.



Prueba en ambiente no controlado, Autores: Jonathan Quilumbaquin, Isaac Reyes.

4.4.4 Respuesta asistente odontológico

Para efectuar esta prueba se realizó un proceso de limpieza dental utilizando todas las herramientas requeridas como se muestra en la Figura 4.15.

Figura 4.15. Consultorio dental.



Área de trabajo equipada con asistente dental, Autores: Jonathan Quilumbaquin, Isaac Reyes.

En la Figura 4.16 se muestra el proceso odontológico utilizando el asistente dental en un paciente.

Figura 4.16. Intervención asistente dental.



Proceso de succión asistente dental, Autores: Jonathan Quilumbaquin, Isaac Reyes.

Una vez realizada la prueba se obtuvieron los siguientes resultados:

- Es fundamental dimensionar bien el área de trabajo ya que se debe evitar que el brazo robótico tenga cualquier obstrucción en su trayectoria, esto puede causar que durante su funcionamiento choque con algún objeto externo impidiendo así su correcta operación.
- El asistente dental tuvo un retardo de respuesta de 3s con relación al ambiente controlado, esto depende directamente de la conexión a internet que se tuvo en la clínica odontológica.
- El asistente dental responde mejor cuando existe poco ruido en la sala quirúrgica, en este caso el compresor se encontraba dentro del consultorio provocando más ruido de lo normal por lo que el odontólogo se vió con la necesidad de alzar la voz.

En la tabla 4.1 se resumen los resultados de los comandos ejecutados en el proceso odontológico con el prototipo THOR.

Tabla 4.1. Resultados de los comandos ejecutados.

Comando Persona	Inicio	Succionar	Finalizar	Seguro
1	✓	✓	X	✓
2	✓	✓	✓	X
3	X	✓	✓	✓
4	✓	✓	✓	✓
5	X	✓	✓	✓
6	✓	✓	✓	✓
7	✓	✓	✓	✓
8	X	✓	✓	✓
9	X	X	✓	✓
10	✓	✓	✓	✓
11	✓	✓	X	✓
12	✓	✓	X	✓
13	✓	✓	✓	✓
14	✓	✓	✓	X

✓ Palabra reconocida, X Palabra no reconocida, Autores: Jonathan Quilumbaquin, Isaac

Reyes.

CAPÍTULO 5

Conclusiones y Recomendaciones

En este capítulo se detallan las conclusiones obtenidas tras las pruebas realizadas en la clínica dental y se plantean las recomendaciones para el uso adecuado del asistente.

Conclusiones:

- La funcionalidad de un brazo robótico en la actualidad es de gran ayuda ya que estos permiten reducir el tiempo y aumentar la precisión del trabajo, además son útiles para realizar tareas simultáneas, tal es el caso del asistente odontológico implementado en este proyecto, el cual facilita al odontólogo el proceso de succión y reduce personal del cubículo.
- Después de haber realizado el análisis de la dinámica para la selección de los motores, se concluye que es importante siempre determinar una posición inicial en la cual el actuador (Brazo robótico) permanezca en una posición de reposo que no genere mucho esfuerzo para el par de retención de los motores antes de que empiece a funcionar, de esta manera se garantizará un mayor tiempo de vida útil del prototipo.
- La producción de este prototipo se puede reducir significativamente en cuanto a tiempo y costos utilizando moldes plásticos, con esto optimizaríamos el tiempo de fabricación y el costo del PLA de alta calidad, para otros componentes del prototipo como: la base, el contenedor de circuito electrónico y el acople de la manguera de succión.
- Debido a que el esfuerzo mecánico de los motores es significativo por el tiempo de operación del prototipo, en este diseño se concluye que los engranes mostrados en la Figura 4.13 se impriman en otro material que soporten temperaturas superiores a las del PLA (200°C) estos pueden ser de fibra de vidrio o grilon.
- Se optó por utilizar Amazon Alexa como el medio de interacción entre el brazo robótico y el odontólogo, debido a su compatibilidad con distintos módulos de control y por su rápida respuesta a los comandos de voz, ya que las pruebas con el módulo V.3 presentaron problemas al momento de grabar las voces además de un retardo significativo al procesarlas.
- Las 4 palabras específicas mencionadas en el literal 4.3.2, fueron pensadas de manera que sean cortas y no llamen a otra función de Alexa, esto hubiese provocado una confusión en el módulo de reconocimiento de voz tal es el caso de S.O.S., emergencia o parar. Así como para que el odontólogo realice su trabajo de una manera más rápida y eficaz.
- Según los resultados obtenidos en los apartados 4.3.1 y 4.3.2 se puede decir que la reacción del asistente dental variará mucho su funcionamiento según las condiciones que la provea el entorno de trabajo, la calidad del internet y la ubicación de equipos de trabajo; tal es el

caso de la conexión de red ya que en el consultorio dental el router está a una distancia de 10m provocando retardo en la ejecución de los comandos de voz y su ancho de banda era menor a 20Mbps.

Recomendaciones:

- En el prototipo desarrollado, teniendo en cuenta que es el primero en su clase, a manera de mejorar su funcionalidad se sugiere cambiar el brazo mecánico por uno más liviano y de mayor alcance ya que para el proceso no necesita mover demasiado peso incluso se puede abaratar costos al momento de construcción.
- La conexión a internet juega un rol fundamental para la velocidad de respuesta del asistente, por lo que se recomienda tener un mínimo de 20 Mbps y que el router no este mas allá de los 5m.
- Se recomienda utilizar llantas con freno ya que resulta complicado el transporte del asistente dental debido a su peso.
- Para su correcto funcionamiento se debe realizar mantenimiento preventivo cada 2 meses y cada 6 meses mantenimiento correctivo, evitando así daños irreversibles. Así como la limpieza y desinfección de la manguera del asistente al finalizar cada tratamiento
- Se recomienda llamar a un técnico para la configuración del asistente a la conexión de internet.
- Se recomienda previamente ajustar el nivel de succión de la bomba escogiendo una fuerza ideal para el trabajo requerido

Referencias

- Aline, M., Diogo, A., Assucao, M., Araujo, P., & Pinto, P. (2013). Importancia de la protección de la mesa de instrumentos quirúrgicos en la contaminación intraoperatoria de cirugías limpias. *Revista Latino-Americana de Enfermagem*, 21(1), 1–8.
- AngelLM. (2016, February 12). Thor - Open Source, 3D printable robotic arm. Retrieved June 22, 2022, from www.thingiverse.com website: <https://hackaday.io/project/12989-thor>
- Arcenegui, R. R. (2019). Planificación de Trayectorias para Robots Autónomos en Entornos Industriales. *Universidad de Sevilla*, 49.
- ARDUINO. (2020). *Software de Arduino*. 1. Retrieved from <https://arduino.cl/programacion/Automatización>, I. (2017). *Dinámica de Robots*. (Parte II), 1–29. Retrieved from <https://nbio.umh.es/files/2012/04/practica3.pdf>
- Barrientos, A. (2007). *Fundamentos de robótica*. Retrieved from <http://www.sidalc.net/cgi-bin/wxis.exe/?IsisScript=SUV.xis&method=post&formato=2&cantidad=1&expresion=mfn=009858>
- Biendicho Lletí, F., & Toledo Alarcón, F. J. (2015). Comunicación Bluetooth entre Arduino UNO y Android aplicado a un detector de mentiras. *Universidad Politécnica de Valencia*, 50.
- Candelario Elías Tutor, J., & Luque Estepa, A. (2016). *Implementación de WPS en el firmware NodeMCU para el ESP8266*. 85.
- Colorado, M. R. (2016). Cinemática y dinámica de robots manipuladores.
- Diseño circuito electrónico y fabricación de PCB - EasyEDA. (2017, March 18). Retrieved January 19, 2022, from https://easyeda.com/feather/Diseno_circuito_electronico_y_fabricacion_de_PCB-980f079572fa4bce9572511f21189138
- García, V. (2020, March 5). DESCRIPCIÓN DEL DRIVER A4988 – Electrónica Práctica Aplicada. Retrieved December 28, 2021, from <https://www.diarioelectronicohoy.com/blog/descripcion-del-driver-a4988>
- Gilat, A. (2006). *Matlab: una introducción con ejemplos prácticos*. Reverté.
- Giraldo, L., Delgado, E., & Germán, C. (2006). Cinemática Inversa de un Brazo Robot Utilizando Algoritmos Genéticos. *Revista Avances En Sistemas e Informática*, 3(1), 29–34.

International Federation of Robotics. (2021). Robot Density nearly Doubled globally - International Federation of Robotics. Retrieved March 22, 2022, from <https://ifr.org/ifr-press-releases/news/robot-density-nearly-doubled-globally>

Isaac. (2021). Nema 17: todo sobre el motor paso a paso compatible con Arduino. Retrieved June 22, 2022, from <https://www.hwlibre.com/nema-17/?fbclid=IwAR2YTLgOpIA1eXEcQY0sZQ-gVYCN2CCcpTVZzwbXy3vM5V6KGm4XXGf4dc0>

Manurung, A. (2022). Inverse dynamics with recursive Newton-Euler.

Mega, A., Atmega, A., Mega, A., & Pro, M. (2017). *EnM Industry Mega2650PRO Datasheet Mega 2650 PRO mini 5V ATMEGA 2650-16AU Development Board EnM Industry*. 1–7.

Ramos, S. (2018). Planificación de Trayectorias. Retrieved March 22, 2022, from <http://nbio.umh.es/files/2012/04/practica5.pdf>

Revista de Robots. (2021). ¿Qué es la ROBÓTICA?, tipos y muchos ejemplos en 2020. Retrieved March 22, 2022, from Revista de Robots website: <https://revistaderobots.com/robots-y-robotica/que-es-la-robotica/>

STEPPERONLINE. (2022, February 14). Nema 17 Bipolar STEPPERONLINE. Retrieved July 25, 2022, from <https://www.omc-stepperonline.com/nema-17-bipolar-1-8deg-26ncm-36-8oz-in-0-4a-12v-42x42x34mm-4-wires-300mm-length-cable-17hs13-0404s?tracking=5c5b1e7294397>

Studio, A. (2021, May 17). Introducción a Android Studio. Retrieved January 19, 2022, from <https://developer.android.com/studio/intro?hl=es-419>

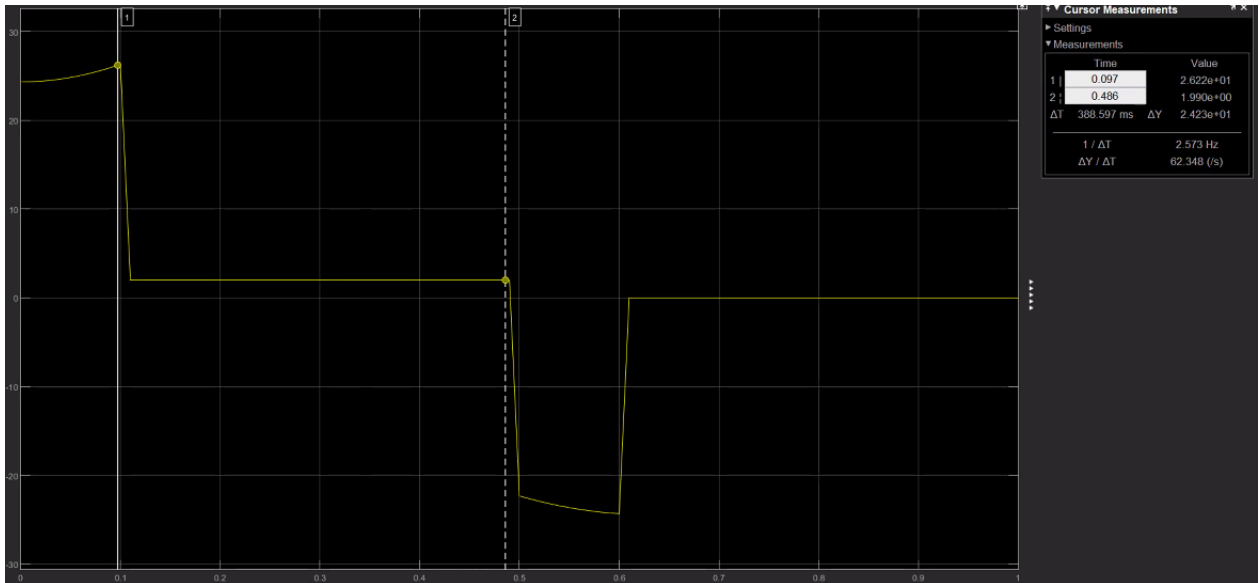
Ultimaker Cura: software de impresión 3D. (2017). Retrieved May 30, 2022, from <https://ultimaker.com/es/software/ultimaker-cura>

Vele, O. (2005). *Cinemática inversa de un robot bípedo*. (May 2006), 1–20.

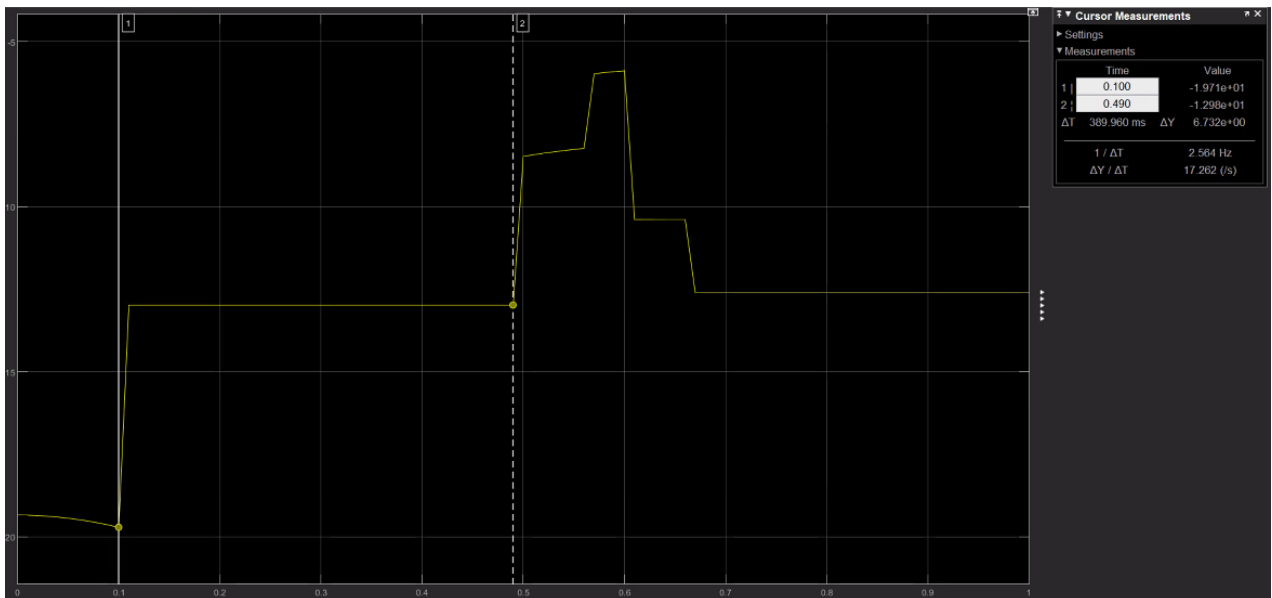
Anexos

Anexo1.

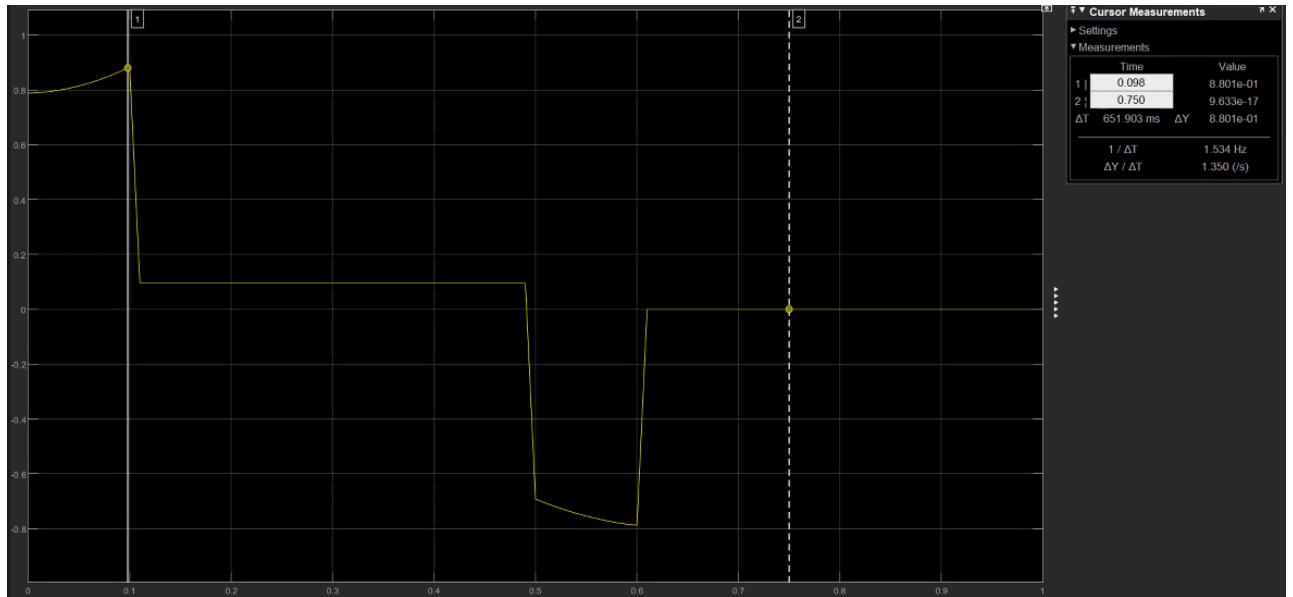
Perfil trapezoidal de la articulación 2.



Perfil trapezoidal de la articulación 3.



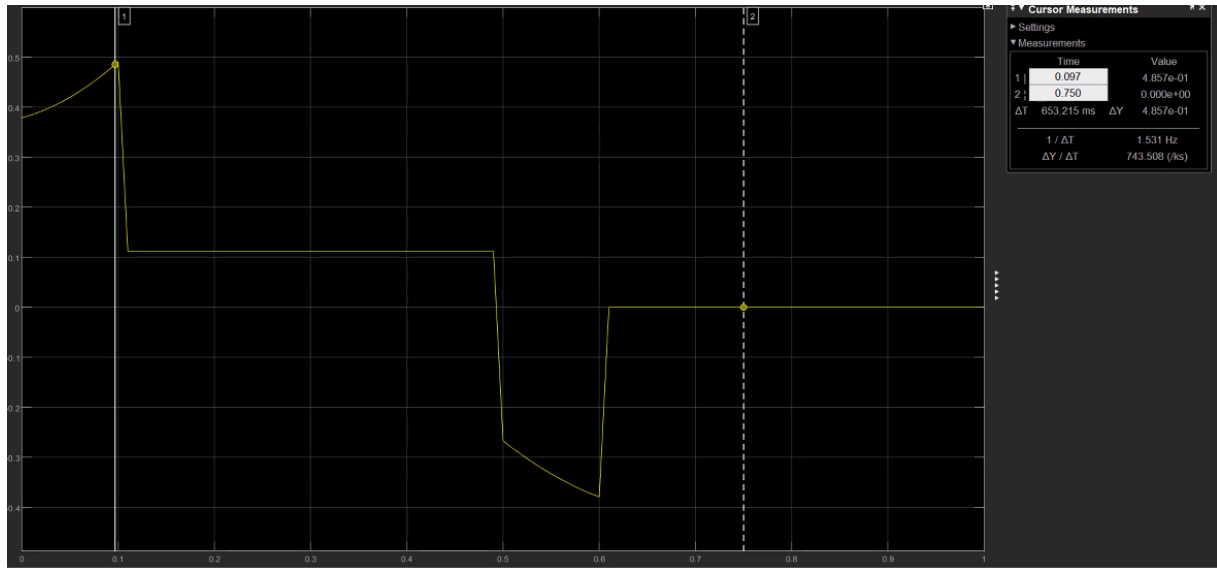
Perfil trapezoidal de la articulación 4.



Perfil trapezoidal de la articulación 5.



Perfil trapezoidal de la articulación 6.



Anexo 2. Codificación Matlab Planificador de trayectorias.

```
function mat_q = planifica6(p1, p2, n, s, a, codo, muneca, npuntos)

% Cálculo del vector unitario
u = p2-p1;
mu = sqrt(u(1)^2+u(2)^2+u(3)^2);
u = (1/mu)*u;

% Cálculo de la distancia entre puntos
d = mu/(npuntos+1);

for i=0:(npuntos+1)
    % Cálculo de la posición cartesiana actual de la mano del manipulador
    p = p1+(i*d)*u;
    T = [n s a p];
    % Cálculo de las coordenadas articulares
    q = inversekinematic6(T,codo,muneca);
    mat_q(:,i+1) = q;
end
```

Anexo 3. Animación punto inicial – punto final.

```
function animacion6(mat_q)

% Parámetros Denavit-Hartenberg del robot
d = [0.202 0 0 0.195 0 0.067];
a = [0 0.160 0 0 0 0 ];
alfa = [-pi/2 0 pi/2 -pi/2 pi/2 0 ];

% Vector de posición (x, y, z) del sistema de coordenadas de referencia
x0 = 0; y0 = 0; z0 = 0;
```

```

% Se dibuja el sistema de coordenadas de referencia. Se asigna el modo XOR
para borrar
% sólo el robot dibujado anteriormente. Se utiliza un grosor de línea de 2
unidades
p = plot3(x0,y0,z0,'EraseMode','xor','LineWidth',2);
% Se asigna una rejilla a los ejes
grid;
% Se establecen los límites de los ejes
axis([-1 1 -1 1 0 1.5]);
% Mantiene el gráfico actual
hold on;

% Número de columnas de la matriz
n = size(mat_q,2);

% Se dibuja la disposición del robot correspondiente a cada columna
for i=1:n

    % Variables articulares del brazo robot
    teta1 = mat_q(1,i);
    teta2 = mat_q(2,i);
    teta3 = mat_q(3,i);
    teta4 = mat_q(4,i);
    teta5 = mat_q(5,i);
    teta6 = mat_q(6,i);

    % Matrices de transformación homogénea entre sistemas de coordenadas
consecutivos
    A01 = denavit(teta1, d(1), a(1), alfa(1));
    A12 = denavit(teta2, d(2), a(2), alfa(2));
    A23 = denavit(teta3, d(3), a(3), alfa(3));
    A34 = denavit(teta4, d(4), a(4), alfa(4));
    A45 = denavit(teta5, d(5), a(5), alfa(5));
    A56 = denavit(teta6, d(6), a(6), alfa(6));

    % Matrices de transformación del primer sistema al correspondiente
    A02 = A01 * A12;
    A03 = A02 * A23;
    A04 = A03 * A34;
    A05 = A04 * A45;
    A06 = A05 * A56;

    % Vector de posición (x, y, z) de cada sistema de coordenadas
    x1 = A01(1,4); y1 = A01(2,4); z1 = A01(3,4);
    x2 = A02(1,4); y2 = A02(2,4); z2 = A02(3,4);
    x3 = A03(1,4); y3 = A03(2,4); z3 = A03(3,4);
    x4 = A04(1,4); y4 = A04(2,4); z4 = A04(3,4);
    x5 = A05(1,4); y5 = A05(2,4); z5 = A05(3,4);
    x6 = A06(1,4); y6 = A06(2,4); z6 = A06(3,4);

    % Se dibuja el robot
    x = [x0 x1 x2 x3 x4 x5 x6];
    y = [y0 y1 y2 y3 y4 y5 y6];
    z = [z0 z1 z2 z3 z4 z5 z6];
    set(p,'XData',x,'YData',y,'ZData',z);
    % Se fuerza a MATLAB a actualizar la pantalla
    drawnow;
    pause(0.1); end

```


Anexo 4. Valores Tiempo, Posición, velocidad y aceleración.

```
function [t, pos_plan, vel_plan, ace_plan] = planificador(q1,q2)

%*****parámetros de los
accionamientos*****

%-----
% Especificaciones de los tiempos de arranque y frenado de cada motor.
%-----

tmotor = 0.1*ones(6,2);

%-----
%Velocidades Maxima de cada motor.
%-----

velmax = [1.0472;1.0472;1.0472;1.0472;1.0472;1.0472];

%*****planificador
coordinado*****

%-----
% Inicialización de los vectores posicion - Velocidad - aceleracion.
%-----

q = zeros(6,1);
q0 = [q1 q q];
qf = [q2 q q];

%-----
% Sincronización de los motores para que realizen un movimiento coordinado
%-----

[velo2,tmaximo]=sincronizador(q0,qf,velmax);

%-----
%Inicialización de la escala de tiempo y las matrices.
%-----

t = 0:0.01:(tmaximo+0.15);

% +0.15 se suma con el fin de aumentar el intervalo de tiempo y muestrear
% todo el intervalo de frenado de la articulación, asumiendo las
% aproximaciones realizadas en la función SINCRONIZADOR.

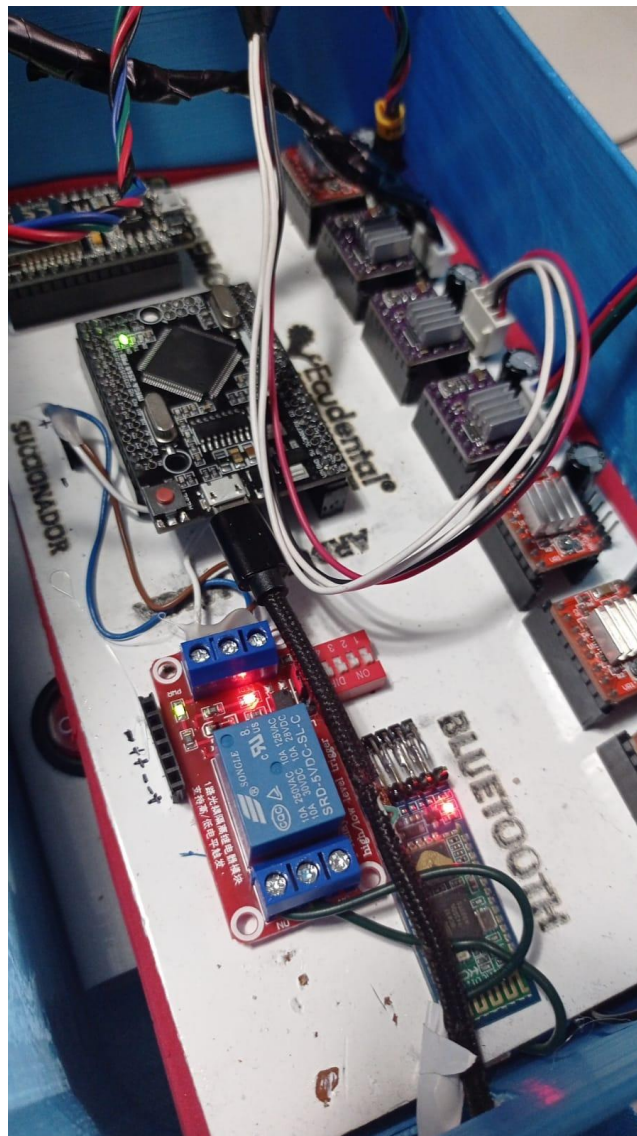
ini=zeros(length(t),1);
pos_plan(:,1)=ini;
vel_plan(:,1)=ini;
ace_plan(:,1)=ini;

%-----
% Cálculo de los coeficientes de los polinomios y evaluación de los
% polinomios de interpolación.
%-----

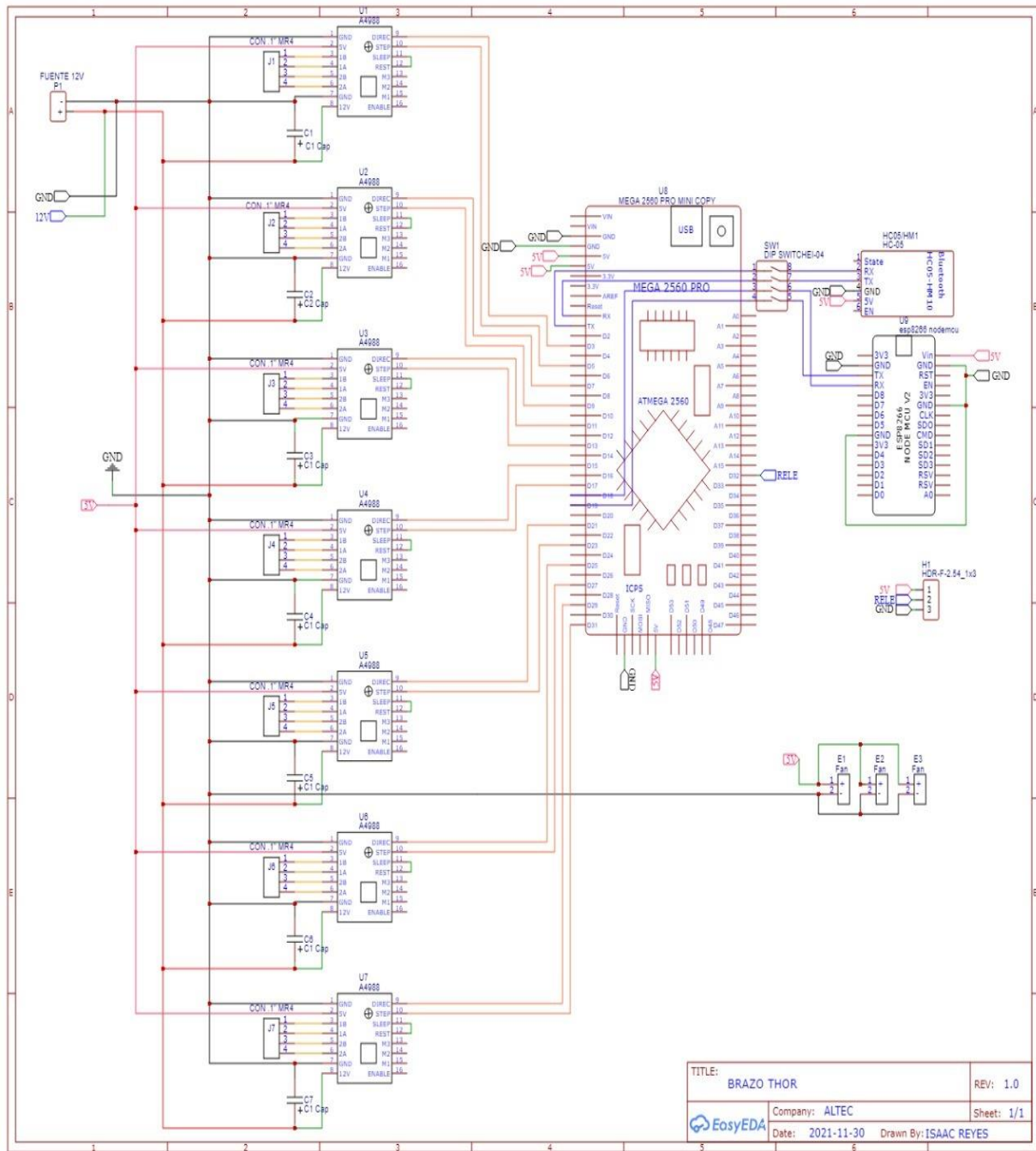
for i = 1:6
```

```
[caso,A,tt] = calculocoef(i,velo2,q0,qf,tmotor);  
posi=evalpos(t,tt,caso,A);  
pos_plan(:,i)=posi';  
ve=evalvel(t,tt,caso,A);  
vel_plan(:,i)=ve';  
ace=evalacel(t,tt,caso,A);  
ace_plan(:,i)=ace';  
end;  
  
return
```

Anexo 5. Prueba de energización asistente dental.



Anexo 6. Esquema electrónico.



Anexo 7. Manual de usuario.

MANUAL DE USUARIO



Asistente Dental

El proceso para la ejecución del asistente dental es bastante simple como enlazarse al BLUETOOTH “Ecdental” y colocar la clave “Ecdental2022”, previo a estos pasos se debe tener la app instalada y posicionar al brazo robótico en un punto cero cómo se detalla en las imágenes.

DISPOSITIVOS VINCULADOS

Bluetooth Speaker 37:BB:6E:BF:9B:DD
AVH-X2650BT 90:03:B7:F0:D4:26
BZK A4:77:58:02:C8:27
EcuDental 00:19:07:00:1C:52



Una vez establecida la posición inicial del asistente basta con manipular las articulaciones en la posición que necesitemos, posteriormente damos al botón guardar.



- Articulación 1 = 0
- Articulación 2 = 0
- Articulación 3 = 0
- Articulación 4 = 0
- Articulación 5 = 0
- Articulación 6 = 0
- VELOCIDAD = BAJA

Una vez guardadas las posiciones basta con llamar con los siguientes comandos de voz:

- **Inicio:** Regresa el brazo a las posiciones donde inició.
- **Succionar:** Mueve el brazo a la posición guardada y enciende la bomba de succión.
- **Finalizar:** Apaga la bomba de succión y va a la posición inicial.
- **Seguro:** Se retira completamente del área de trabajo.



MANUAL TÉCNICO



Asistente Dental

Arduino mega 2560 PRO

Microcontrolador	ATmega2560
Voltaje de entrada recomendado	7-12 V
Voltaje de trabajo	5 V
USB	Micro USB
Pines digitales I/O	54
Pines PWM I/O	15
Pines analógicos	16
Corriente DC por pines	20 mA
Flash Memory	256 KB
SRAM	8 KB
EEPROM	4 KB
Longitud	52mm
Ancho	38mm
Peso	25 g
Frecuencia del reloj	16Mhz

NodeMCU

Procesador	ESP8266 @ 80MHz (3.3V) (ESP-12E)
Flash Memory	4 MB
SoM	ESP-12E (Ai-Thinker)
SoC	ESP8266 (Espressif)
Frecuencia de reloj	80 MHZ/160MHZ
RAM	96 KB
Pines Digitales GPIO	17
Dimensiones	49*26*12 mm

Asistente robótico dental

Voltaje de operación	110 V AC
Frecuencia	60 Hz.
Número de motores	7
Desplazamiento	XYZ
Potencia	115w
Rango de Presión bomba succionadora	700hPa-1060hPa
Capacidad bomba succionadora	1000ml
Dimensiones	35x40x90