



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA ELECTRÓNICA Y AUTOMATIZACIÓN

DISEÑO Y CONSTRUCCIÓN DE MANOS ROBÓTICAS
PROGRAMABLES APLICADAS AL ÁREA DEL
ENTRETENIMIENTO EN EL CAMPO MUSICAL

Trabajo de Titulación

Tipo: Proyecto Técnico

Presentado para optar al grado académico de:
INGENIERO EN ELECTRÓNICA Y AUTOMATIZACIÓN

AUTORES:

MARCO FABRICIO AGUIRRE SALAS

JEAN CARLOS MENESES RIVERA

Riobamba – Ecuador

2023



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA ELECTRÓNICA Y AUTOMATIZACIÓN

DISEÑO Y CONSTRUCCIÓN DE MANOS ROBÓTICAS
PROGRAMABLES APLICADAS AL ÁREA DEL
ENTRETENIMIENTO EN EL CAMPO MUSICAL.

Trabajo de Titulación

Tipo: Proyecto Técnico

Presentado para optar al grado académico de:

INGENIERO EN ELECTRÓNICA Y AUTOMATIZACIÓN

AUTORES: MARCO FABRICIO AGUIRRE SALAS

JEAN CARLOS MENESES RIVERA

DIRECTOR: ING. JOSÉ ENRIQUE GUERRA SALAZAR

Riobamba – Ecuador

2023

© 2023, Marco Fabricio Aguirre Salas, Jean Carlos Meneses Rivera

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Nosotros, Marco Fabricio Aguirre Salas y Jean Carlos Meneses Rivera, declaramos que el presente Trabajo de Titulación es de nuestra autoría y los resultados del mismo son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autores asumimos la responsabilidad legal y académica de los contenidos de este Trabajo de Titulación; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 01 de noviembre de 2023



Marco Fabricio Aguirre Salas
1804362059



Jean Carlos Meneses Rivera
0605435288

ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO
FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
CARRERA ELECTRÓNICA Y AUTOMATIZACIÓN

El Tribunal del Trabajo de Titulación certifica que: El Trabajo de Titulación; tipo: Proyecto Técnico, **DISEÑO Y CONSTRUCCIÓN DE MANOS ROBÓTICAS PROGRAMABLES APLICADAS AL ÁREA DEL ENTRETENIMIENTO EN EL CAMPO MUSICAL**, realizado por los señores: **MARCO FABRICIO AGUIRRE SALAS Y JEAN CARLOS MENESES RIVERA**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Titulación, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

FIRMA

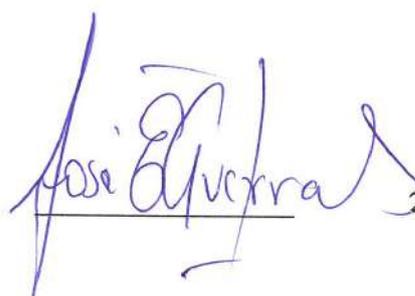
FECHA

Ing. Jorge Luis Hernández Ambato PhD
PRESIDENTE DEL TRIBUNAL



2023-11-01

Ing. José Enrique Guerra Salazar
DIRECTOR DEL TRABAJO DE TITULACIÓN



2023-11-01

Ing. José Luis Morales Gordon
ASESOR DEL TRABAJO DE TITULACIÓN



2023-11-01

DEDICATORIA

El presente Trabajo de Titulación está dedicado a mi madre Mercedes, por su constante esfuerzo y sacrificio que han sido necesarios para sacarme adelante y llegar a cumplir una meta en mi vida profesional, y a mi hermana Alexandra por brindarme su total apoyo que ha sido mi inspiración constante para salir adelante, esta tesis es un tributo a su fe inquebrantable en mí y a su amor incondicional.

Marco

El presente Trabajo de Titulación está dedicado a mis padres por haberme brindado su cariño y apoyo incondicional durante mi formación académica ya que ellos fueron los cimientos para alcanzar esta meta. A mis hermanos por brindarme ánimo para no decaer y lograr conseguir este logro.

Jean

AGRADECIMIENTO

Doy gracias a Dios por brindarme una buena experiencia en la Escuela Superior Politécnica de Chimborazo, y a su vez agradezco por llegar a convertirme en un profesional, agradezco a todos los ingenieros que me ayudaron en el proceso de formación, por llegar a compartirme sus conocimientos. Y de igual manera agradezco a mi madre y hermana por su apoyo y amor incondicional.

Marco

Agradezco a la Escuela Superior Politécnica de Chimborazo por haberme abierto las puertas para ser parte de ella y lograr formarme académicamente, a los ingenieros que han estado presentes durante este proceso impartiendo sus conocimientos y experiencias ayudando a convertirme en un profesional. Y a su vez agradezco a aquellos familiares y amigos los cuales han estado presentes brindándome su apoyo y ayuda.

Jean

ÍNDICE DE CONTENIDO

| | |
|------------------------|------|
| ÍNDICE DE TABLAS..... | x |
| ÍNDICE DE FIGURAS..... | xii |
| ÍNDICE DE ANEXOS..... | xv |
| RESUMEN..... | xvi |
| SUMMARY | xvii |
| INTRODUCCIÓN | 1 |

CAPÍTULO I

| | | |
|-----------|--------------------------------------|----------|
| 1. | DIAGNÓSTICO DEL PROBLEMA..... | 3 |
| 1.1 | PLANTEAMIENTO DEL PROBLEMA | 3 |
| 1.2 | JUSTIFICACIÓN TEÓRICA | 3 |
| 1.3 | JUSTIFICACIÓN APLICATIVA..... | 5 |
| 1.4 | OBJETIVOS..... | 6 |
| 1.5 | METODOLOGÍA | 6 |

CAPÍTULO II

| | | |
|-----------|---|-----------|
| 2. | MARCO TEÓRICO | 7 |
| 2.1 | EL PIANO | 7 |
| 2.2 | PROYECTOS DE ROBOTS PIANISTAS | 8 |
| 2.2.1 | <i>Diseño de controles y hardware de dedos robóticos para tocar el piano.....</i> | <i>8</i> |
| 2.2.2 | <i>Pianista robótico haciendo hablar a los pianos</i> | <i>9</i> |
| 2.2.3 | <i>El tercer pulgar robótico supernumerario para tareas musicales.....</i> | <i>10</i> |
| 2.2.4 | <i>Diseño de una mano robótica para tocar el piano</i> | <i>11</i> |
| 2.2.5 | <i>Brazo robótico para tocar el piano.....</i> | <i>11</i> |
| 2.2.6 | <i>Robot que lee música y toca el piano.....</i> | <i>12</i> |
| 2.2.7 | <i>Desarrollo de un novedoso robot de piano a dos manos</i> | <i>13</i> |
| 2.2.8 | <i>Robot educativo que interpreta melodías en piano.....</i> | <i>14</i> |

| | | |
|--------|--|----|
| 2.2.9 | <i>Desarrollo de una mano robótica con capacidad de manipulación</i> | 15 |
| 2.2.10 | <i>Análisis de los elementos utilizados en los proyectos de robot pianista</i> | 15 |
| 2.3 | ARQUITECTURA DE UN ROBOT PIANISTA EMPLEADA PARA EL PROTOTIPO | 16 |
| 2.4 | TARJETAS DE DESARROLLO..... | 17 |
| 2.4.1 | <i>Selección de tarjeta de desarrollo de hardware</i> | 18 |
| 2.5 | ACTUADORES | 19 |
| 2.5.1 | <i>Motor paso a paso</i> | 19 |
| 2.5.2 | <i>Servomotores</i> | 20 |
| 2.6 | SENSORES | 21 |
| 2.6.1 | <i>Sensor magnético</i> | 21 |
| 2.6.2 | <i>Selección del sensor</i> | 21 |
| 2.7 | ALIMENTACIÓN | 21 |
| 2.7.1 | <i>Fuente de Alimentación</i> | 21 |
| 2.7.2 | <i>Baterías</i> | 22 |
| 2.7.3 | <i>Selección de la alimentación</i> | 22 |
| 2.8 | APLICACIÓN MÓVIL | 23 |
| 2.9 | BASES DE DATOS EN LA NUBE | 24 |
| 2.9.1 | <i>Selección de base de datos en la nube</i> | 24 |

CAPÍTULO III

| | | |
|-----------|--|-----------|
| 3. | MARCO METODOLÓGICO | 27 |
| 3.1 | REQUERIMIENTOS DEL PROTOTIPO..... | 27 |
| 3.2 | CONCEPCIÓN DE ARQUITECTURA DEL SISTEMA | 28 |
| 3.3 | DIAGRAMA DE BLOQUES DEL SISTEMA | 30 |
| 3.4 | DISEÑO MECÁNICO DEL PROTOTIPO | 31 |
| 3.4.1 | <i>Diseño de las manos</i> | 31 |
| 3.4.2 | <i>Diseño del tornillo lineal</i> | 39 |
| 3.5 | DESCRIPCIÓN DE LOS ELEMENTOS <i>HARDWARE</i> DEL SISTEMA | 40 |
| 3.5.1 | <i>Tornillo lineal Fuyu FSL40</i> | 40 |
| 3.5.2 | <i>Fuente de poder T-250W 24V 10A</i> | 41 |
| 3.5.3 | <i>Fuente de alimentación RS25-5</i> | 41 |
| 3.5.4 | <i>Sensor magnético</i> | 42 |
| 3.5.5 | <i>LFD-01 9G robot servo 180°</i> | 43 |
| 3.5.6 | <i>Controlador para motor paso a paso DM542T</i> | 44 |

| | | |
|-------|--|----|
| 3.5.7 | <i>Microcontrolador ESP32</i> | 44 |
| 3.6 | ESQUEMA DE CONEXIÓN DEL PROTOTIPO..... | 45 |
| 3.6.1 | <i>Esquema de conexión de la placa principal</i> | 46 |
| 3.6.2 | <i>Conexión del controlador DM542T</i> | 50 |
| 3.6.3 | <i>Diseño de la estructura del prototipo</i> | 50 |
| 3.7 | PROGRAMACIÓN DEL PROTOTIPO | 53 |
| 3.8 | BASE DE DATOS EN FIREBASE | 58 |
| 3.9 | INTERFAZ DE INTERVENCIÓN CON EL USUARIO..... | 60 |

CAPITULO IV

| | | |
|-----------|--|-----------|
| 4. | PRUEBAS Y RESULTADOS | 63 |
| 4.1 | CONSIDERACIONES GENERALES | 63 |
| 4.2 | VALIDACIÓN DE SENSORES | 65 |
| 4.2.1 | <i>Validación del sensor magnético de la mano derecha</i> | 65 |
| 4.2.2 | <i>Validación del sensor magnético de la mano izquierda</i> | 67 |
| 4.3 | ESTABILIDAD DE SENSORES | 69 |
| 4.3.1 | <i>Estabilidad del sensor magnético de la mano derecha</i> | 69 |
| 4.3.2 | <i>Estabilidad del sensor magnético de la mano izquierda</i> | 70 |
| 4.4 | ESTABILIDAD DE LOS TIEMPOS DE REPRODUCCIÓN DE LAS CANCIONES..... | 71 |
| 4.4.1 | <i>Estabilidad de reproducción de la primera canción</i> | 71 |
| 4.4.2 | <i>Estabilidad de reproducción de la última canción</i> | 73 |
| 4.5 | PRECISIÓN DE DESPLAZAMIENTO DEL PROTOTIPO | 75 |
| 4.5.1 | <i>Precisión de desplazamiento de la mano derecha</i> | 75 |
| 4.5.2 | <i>Precisión de desplazamiento de la mano izquierda</i> | 78 |
| 4.6 | COMUNICACIÓN | 82 |
| 4.7 | CONSUMO ENERGÉTICO..... | 83 |
| 4.8 | FUNCIONAMIENTO GENERAL DEL PROTOTIPO..... | 84 |
| 4.9 | ANÁLISIS DE COSTO DEL PROTOTIPO..... | 85 |

| | |
|---------------------------|-----------|
| CONCLUSIONES | 87 |
|---------------------------|-----------|

| | |
|------------------------------|-----------|
| RECOMENDACIONES | 89 |
|------------------------------|-----------|

BIBLIOGRAFÍA

ANEXOS

ÍNDICE DE TABLAS

| | |
|--|----|
| Tabla 2-1. Comparativa de tarjetas de desarrollo..... | 18 |
| Tabla 2-2. Comparativa entre motores paso a paso lineales | 19 |
| Tabla 2-3. Comparativa entre la familia de servomotores | 20 |
| Tabla 2-4. Comparativa entre sensores de distancia | 21 |
| Tabla 2-5. Comparativa entre fuente de alimentación y baterías | 23 |
| Tabla 2-6. Sistemas operativos para móviles a nivel mundial | 23 |
| Tabla 2-7. Sistemas operativos para móviles a nivel de Ecuador | 24 |
| Tabla 2-8. Comparativa de base de datos en la nube | 25 |
| Tabla 3-1. Características técnicas del tornillo lineal Fuyu FSL40..... | 40 |
| Tabla 3-2. Especificaciones de la fuente de alimentación T-250W 24 V 10 A | 41 |
| Tabla 3-3. Especificaciones de la fuente de alimentación RS25-5 | 42 |
| Tabla 3-4. Especificaciones del sensor magnético..... | 43 |
| Tabla 3-5. Especificaciones del servomotor LFD-01..... | 43 |
| Tabla 3-6. Especificaciones del controlador DM542T | 44 |
| Tabla 3-7. Características técnicas del microcontrolador ESP32..... | 45 |
| Tabla 3-8. Medidas que constituye la estructura del prototipo | 53 |
| Tabla 3-9. Librerías usadas en la programación del módulo maestro..... | 54 |
| Tabla 3-10. Librerías usadas en la programación del módulo esclavo ESP32..... | 55 |
| Tabla 3-11. Distribución de número de pasos para la creación de las notas musicales | 58 |
| Tabla 4-1. Parámetros para la toma de muestras..... | 65 |
| Tabla 4-2. Criterios de valoración para los resultados | 65 |
| Tabla 4-3. Error absoluto y relativo del sensor magnético de la mano derecha..... | 66 |
| Tabla 4-4. Error absoluto y relativo del sensor magnético de la mano izquierda | 68 |
| Tabla 4-5. Prueba de estabilidad del sensor magnético de la mano derecha..... | 69 |
| Tabla 4-6. Prueba de estabilidad del sensor magnético de la mano izquierda | 71 |
| Tabla 4-7. Prueba de estabilidad del tiempo de reproducción de la primera canción | 72 |
| Tabla 4-8. Prueba de estabilidad del tiempo de reproducción de la última canción | 74 |
| Tabla 4-9. Error absoluto y relativo del dedo guía de la mano derecha en la tecla inicial..... | 76 |
| Tabla 4-10. Error absoluto y relativo del dedo guía de la mano derecha en la tecla final | 77 |
| Tabla 4-11. Error absoluto y relativo del dedo guía de la mano izquierda en la tecla inicial | 79 |
| Tabla 4-12. Error absoluto y relativo del dedo guía de la mano izquierda en la tecla final | 81 |
| Tabla 4-13. Prueba de comunicación | 82 |
| Tabla 4-14. Consumo energético | 84 |

Tabla 4-15. Análisis de costo del prototipo..... 85

ÍNDICE DE FIGURAS

| | |
|---|----|
| Ilustración 2-1. Nombres de las teclas de un piano | 7 |
| Ilustración 2-2. Diseño de controles y hardware de dedos robóticos para tocar el piano..... | 8 |
| Ilustración 2-3. Pianista robótico haciendo hablar a los pianos..... | 9 |
| Ilustración 2-4. El tercer pulgar robótico supernumerario para tareas musicales..... | 10 |
| Ilustración 2-5. Diseño de una mano robótica para tocar el piano | 11 |
| Ilustración 2-6. Brazo robótico para tocar el piano | 12 |
| Ilustración 2-7. Robot que lee música y toca el piano..... | 13 |
| Ilustración 2-8. Desarrollo de un novedoso robot de piano a dos manos | 13 |
| Ilustración 2-9. Robot educativo de bajo costo, interpreta melodías en piano | 14 |
| Ilustración 2-10. Desarrollo de una mano robótica con capacidad de manipulación | 15 |
| Ilustración 2-11. Arquitectura general de un robot pianista | 16 |
| Ilustración 2-12. Arquitectura del prototipo de robot pianista | 17 |
| Ilustración 3-1. Concepción de la arquitectura general del prototipo de robot pianista | 29 |
| Ilustración 3-2. Diagrama de bloques del prototipo. | 31 |
| Ilustración 3-3. Estructura y dimensiones de los dedos de la mano derecha..... | 31 |
| Ilustración 3-4. Creación de los 4 dedos para el ensamblaje de la mano derecha..... | 32 |
| Ilustración 3-5. Base para los dedos y servomotores..... | 32 |
| Ilustración 3-6. Sujetadores para cada uno de los dedos | 33 |
| Ilustración 3-7. Sujetador entre el servo y el dedo | 33 |
| Ilustración 3-8. Sujetador para los servomotores parte frontal y posterior..... | 33 |
| Ilustración 3-9. Base para cubrir los servomotores | 34 |
| Ilustración 3-10. Piezas de sujeción | 34 |
| Ilustración 3-11. Ensamble de los dedos y la base | 34 |
| Ilustración 3-12. Ensamble de los sujetadores de los servomotores..... | 35 |
| Ilustración 3-13. Ensamble completo de la mano derecha | 35 |
| Ilustración 3-14. Estructura de los dedos para la mano izquierda | 36 |
| Ilustración 3-15. Creación de los 4 dedos para el ensamblaje de la mano izquierda..... | 36 |
| Ilustración 3-16. Base para los dedos y servomotores de la mano izquierda | 36 |
| Ilustración 3-17. Sujetadores para cada uno de los dedos | 37 |
| Ilustración 3-18. Sujetador entre el servo y el dedo | 37 |
| Ilustración 3-19. Sujetador para los servomotores parte frontal y posterior..... | 37 |
| Ilustración 3-20. Base para cubrir los servomotores | 38 |
| Ilustración 3-21. Piezas de sujeción | 38 |

| | |
|--|----|
| Ilustración 3-22. Ensamble de los dedos y la base | 38 |
| Ilustración 3-23. Ensamble de los sujetadores de los servomotores..... | 39 |
| Ilustración 3-24. Ensamble completo de la mano izquierda..... | 39 |
| Ilustración 3-25. Modelo 3D del tornillo lineal..... | 39 |
| Ilustración 3-26. Tornillo de bola de mesa Fuyu FSL40..... | 40 |
| Ilustración 3-27. Fuente de alimentación T-250W 24 V 10A | 41 |
| Ilustración 3-28. Fuente de alimentación RS25-5 | 42 |
| Ilustración 3-29. Sensor magnético | 42 |
| Ilustración 3-30. Servomotor LFD-01..... | 43 |
| Ilustración 3-31. Controlador DM542T..... | 44 |
| Ilustración 3-32. Microcontrolador ESP32..... | 45 |
| Ilustración 3-33. Distribución por bloques de los elementos hardware..... | 46 |
| Ilustración 3-34. Esquema de conexión de la placa principal | 48 |
| Ilustración 3-35. Placa PCB..... | 49 |
| Ilustración 3-36. Modelo en 3D de la placa..... | 49 |
| Ilustración 3-37. Placa PCB implementada físicamente..... | 50 |
| Ilustración 3-38. Dimensiones de la estructura del prototipo primera parte..... | 51 |
| Ilustración 3-39. Dimensiones de la estructura del prototipo segunda parte..... | 52 |
| Ilustración 3-40. Diseño en 3D de la estructura | 52 |
| Ilustración 3-41. Diagrama de flujo del prototipo | 57 |
| Ilustración 3-42. Información que contiene la base de datos..... | 59 |
| Ilustración 3-43. Envío de la información en tipo String | 59 |
| Ilustración 3-44. Pantalla principal de la aplicación móvil | 60 |
| Ilustración 3-45. Pantalla del modo manual de la aplicación móvil..... | 61 |
| Ilustración 3-46. Pantalla del modo piano de la aplicación móvil..... | 61 |
| Ilustración 3-47. Pantalla del modo piano de la aplicación móvil..... | 62 |
| Ilustración 4-1. Prototipo de manos robóticas aplicadas al área del entretenimiento..... | 63 |
| Ilustración 4-2. Evidencia de toma de muestras del sensor magnético de la mano derecha.... | 66 |
| Ilustración 4-3. Evidencia de toma de muestras del sensor magnético de la mano izquierda ... | 67 |
| Ilustración 4-4. Evidencia de toma de datos para la estabilidad del sensor magnético de la mano derecha | 69 |
| Ilustración 4-5. Evidencia de la toma de datos para la estabilidad del sensor magnético de la mano izquierda..... | 70 |
| Ilustración 4-6. Evidencia de la toma de datos para la estabilidad de la primera canción..... | 72 |
| Ilustración 4-7. Evidencia de la toma de datos para la estabilidad de la última canción..... | 73 |
| Ilustración 4-8. Medidas tomadas entre la tecla inicial y final | 75 |

| | |
|--|----|
| Ilustración 4-9. Evidencia de la toma de muestras de la mano derecha en la tecla inicial | 75 |
| Ilustración 4-10. Evidencia de la toma de muestras de la mano derecha en la tecla final | 77 |
| Ilustración 4-11. Evidencia de la toma de muestras de la mano izquierda en la tecla inicial.... | 79 |
| Ilustración 4-12. Evidencia para la toma de muestras de la mano izquierda en la tecla final ... | 80 |
| Ilustración 4-13. Evidencia de la toma de datos para la prueba de comunicación | 82 |
| Ilustración 4-14. Corriente medida durante la reproducción de la canción 1 | 84 |
| Ilustración 4-15. Observación del prototipo por el pianista Kenny Bonilla | 85 |

ÍNDICE DE ANEXOS

ANEXO A: TORNILLO LINEAL FUYU FSL40

ANEXO B: FUENTE DE PODER T-250W 24V 10A

ANEXO C: FUENTE DE ALIMENTACIÓN RS25-5

ANEXO D: SENSOR MAGNÉTICO

ANEXO E: LFD-01 9G ROBOT SERVO 180°

ANEXO F: CONTROLADOR PARA MOTOR PASO A PASO DM542T

ANEXO G: MICROCONTROLADOR ESP32

ANEXO H: ESQUEMAS PCB

ANEXO I: PROGRAMACIÓN MÓDULO MAESTRO ESP32

ANEXO J: PROGRAMACIÓN MÓDULO ESCLAVO ESP32

ANEXO K: APLICACIÓN MÓVIL

ANEXO L: PRUEBAS DE COMUNICACIÓN VISUALIZADAS EN EL MONITOR SERIAL
IDE

ANEXO M: HOJA DE DATOS DE LA PINZA AMPERIMÉTRICA

ANEXO N: PRUEBAS DE CONSUMO ENERGÉTICO

RESUMEN

El presente trabajo de titulación tuvo como objetivo el diseño y construcción de manos robóticas programables aplicadas al área del entretenimiento en el campo musical. Para su desarrollo se tuvo como enfoque los robots pianistas y mediante revisión bibliográfica de proyectos relacionados al diseño de estos, se estableció los requerimientos de *software* y *hardware* necesarios para que el prototipo funcione como robot pianista. Consta de dos manos y cada una posee un sensor magnético que permite su posicionamiento inicial usando como referencia un dedo guía el cual debe alcanzar una tecla inicial, para su desplazamiento en el eje x, emplea un motor paso a paso que incorpora un tornillo lineal sin fin, además posee cuatro dedos los cuales son accionados mediante servomotores permitiendo el desplazamiento en el eje z para tocar las teclas blancas del piano, como controlador se empleó tarjetas de desarrollo ESP32 para cada mano las cuales están distribuidas como módulos maestro y esclavo, el módulo maestro es el único que posee conexión Wi-Fi para una comunicación con la aplicación móvil y además este se encarga de enviar las instrucciones hacia el módulo esclavo permitiendo así la interpretación de las canciones. Para la validación del prototipo se realizaron pruebas de exactitud y estabilidad para los sensores magnéticos de la mano derecha e izquierda y se obtuvo errores relativos de 1.43% y 1.56% y coeficientes de variación de 3.06% y 3.28% respectivamente. De las pruebas de precisión de desplazamiento de la mano derecha en su tecla inicial y final se obtuvo errores relativos de 1.59% y 8.25% respectivamente y de la mano izquierda en su tecla inicial y final se obtuvo errores relativos de 1.75% y 8.89% respectivamente. De esto se concluye que el prototipo implementado cumple con los requerimientos establecidos para interpretar canciones en un piano.

Palabras clave: <ROBOT PIANISTA>, <MANOS ROBÓTICAS>, <ESP32>, <SERVOMOTOR>, <MOTOR PASO A PASO> <WI-FI>, <ERROR RELATIVO>, <COEFICIENTE DE VARIACIÓN>.

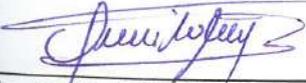


1707-DBRA-UPT-2023

SUMMARY

The objective of this degree work was the design and construction of programmable robotic hands applied to the area of entertainment in the musical field. For its development, pianist robots were targeted, and through a bibliographic review of projects related to their design, the software and hardware requirements necessary for the prototype to function as a pianist robot were established. It consists of two hands and each one has a magnetic sensor that allows its initial positioning using as a reference a guide finger which must reach an initial key, for its displacement in the x axis, it uses a stepper motor that incorporates an endless lineal screw, it also has four fingers which are actuated by servomotors allowing displacement in the z axis to play the white keys of the piano, as a controller ESP32 development cards were used for each hand which are distributed as master and slave modules, the master module is the only one that has a Wi-Fi connection for communication with the mobile application and it is also in charge of sending the instructions to the slave module, thus allowing the interpretation of the songs. For the validation of the prototype, accuracy and stability tests were carried out for the magnetic sensors of the right and left hand and it was obtained relative errors of 1.43% and 1.56%, and coefficients of variation of 3.06% and 3.28% respectively. From the displacement precision tests of the right hand in its initial and final key, relative errors of 1.59% and 8.25% respectively were obtained and of the left hand in its initial and final key, relative errors of 1.75% and 8.89% respectively were obtained. It is concluded that the implemented prototype meets the established requirements to play songs on a piano.

Keywords: <PIANIST ROBOT>, <ROBOTIC HANDS>, <ESP32 DEVELOPMENT BOARD>, <SERVOMOTOR>, <STEPPER MOTOR>, <WI-FI TECHNOLOGY>, <RELATIVE ERROR>, <COEFFICIENT OF VARIATION>.



Lic. Nelly Padilla P. Mgs
0603818717
DOCENTE FIE

INTRODUCCIÓN

La interacción humana con la tecnología es una tendencia casi natural, ya que está presente en la mayoría de las actividades del ser humano. En cuanto al sistema educativo, un fenómeno creciente es el uso de la robótica educativa para el desarrollo de habilidades en niños y jóvenes que apoyen el aprendizaje. Se han detectado varios enfoques, incluido el uso de robots como herramientas para que los estudiantes aprendan a construirlos, programarlos y así generar nuevos prototipos. Todo esto en pro del desarrollo de habilidades del pensamiento lógico-matemático, el algorítmico y habilidades como la creatividad, el trabajo colaborativo, la comunicación, entre otras (González, Flores, Muñoz 2021).

Uno de los principales factores por los que América Latina ha tenido un avance más lento en el desarrollo de la robótica en comparación al resto del mundo es que no existe fondos necesarios para incentivar la innovación, pero sin embargo se busca la manera de reducir esta brecha. Como es el caso en Ecuador el cual propuso iniciativas con el plan llamado “Código INGENIOS” el cual busca la manera para que la gente presente innovaciones en diferentes campos (Secretaría de Educación Superior, Ciencia, Tecnología e Innovación sin fecha). Uno de los ámbitos de la robótica menos estudiados ha sido el musical, pero hay varios países que han desarrollado investigaciones en este ámbito específicamente para que el robot logre interpretar canciones en el piano.

A nivel mundial se han desarrollado ciertas investigaciones en este ámbito, como en Taiwán en el cual se desarrolló dos manos las cuales son capaces de desplazarse de forma lineal a lo largo del piano, y de esta forma poder entonar una canción establecida (Jen-Chang et al. 2014). En Suecia fue desarrollado el robot Mozart2000 el cual es capaz de tocar solo ocho teclas del piano en específico, y procesar imágenes mediante una cámara y de esta manera poder tocar las notas que puede visualizar de una partitura (PHAN, MALM 2019). En Estados Unidos fue desarrollado un prototipo que consta de tres dedos los cuales pueden realizar un pequeño desplazamiento lineal y tocar notas musicales previamente programadas (Ornelas 2022).

A nivel regional en Venezuela, se desarrolló una mano robótica conformada de tres dedos los cuales son movidos a base de motores y engranajes, además esta mano posee un desplazamiento lineal para tocar las teclas de un piano (Pérez, Mendoza 2008). En Colombia se realizó un robot el cual simula el movimiento de un solo dedo, para interpretar melodías sencillas en un piano (Alberto Buitrago Suescún, Consuelo Prieto Ramírez, Augusto Peña Cortés 2012).

El proyecto propuesto se basa en el diseño y construcción de manos robóticas programables aplicadas al área del entretenimiento en el campo musical, utilizando como fuente de inspiración

el principio de funcionamiento de un robot pianista y tomando en cuenta las ventajas y desventajas más relevantes de proyectos previamente realizados.

CAPÍTULO I

1. DIAGNÓSTICO DEL PROBLEMA

En el presente capítulo se analiza el planteamiento del problema del prototipo a desarrollar, la justificación teórica donde se mencionan los diferentes proyectos relacionados al tema, la justificación aplicativa donde se presenta una breve idea de cómo va a funcionar el prototipo, los objetivos y la metodología utilizada en el presente proyecto.

1.1 Planteamiento del problema

¿Cómo diseñar y construir las manos robóticas programables aplicadas al área del entretenimiento en el campo musical?

1.2 Justificación Teórica

La robótica ha tenido un crecimiento notable en todos sus ámbitos en los últimos años, ya que se intentaba reducir la intervención de las personas fomentando el desarrollo en los ámbitos del aprendizaje, tareas en el agro e intervenciones médicas entre otras, dando un crecimiento notable a los robots de servicio, se estima que a nivel mundial estos robots crecerían hasta en un 25% anual y en los robots industriales un 16% (Alfessi et al. 2020).

La creciente que ha ido presentando la robótica a nivel mundial ha sido muy variado dependiendo de los ámbitos en los cuales han sido aplicados, como por ejemplo la robótica a nivel industrial entre los años 2021 - 2026 se estima un crecimiento anual del 14.11% (MordorIntelligence 2023a), la robótica a nivel de asistencia entre los años 2020 – 2025 se estima un crecimiento anual del 21% (MordorIntelligence 2023b), la robótica a nivel educativo entre los años 2020 – 2025 se estima un crecimiento anual del 16% (MordorIntelligence 2023c), la robótica a nivel de agro entre los años 2020 – 2027 se estima un crecimiento anual del 28.7% (Redagrícola 2022), la robótica a nivel de la medicina en el ámbito de la cirugía entre los años 2021 – 2026 se estima un crecimiento anual del 11.4% (MordorIntelligence 2023d).

En lo que respecta a América Latina la robótica entre los años 2023 – 2028 se estima un crecimiento anual del 11%, estando como pilares fundamentales México y Brasil (MordorIntelligence 2023e). En Ecuador la robótica ha tenido un crecimiento demasiado bajo en comparación al resto de América Latina siendo uno de los países menos desarrollados en todos los ámbitos en los que consta la robótica.

Según el diario “EL UNIVERSO” publicado el 19 de agosto de 2018, publicó “Era del robot se instala a paso lento en Ecuador”, se ha visto involucrado en los ámbitos industriales, educativos, médicos, de agro, entre otros. Un ejemplo de un robot para el ámbito de la medicina se encuentra al robot Da Vinci el cual es utilizado para cirugías, para el ámbito del agro existen los robots fumigadores, actualmente la robótica también se ha visto involucrada en medios de entretenimiento un ejemplo de este ámbito son las competencias de robótica donde se puede apreciar una variedad de categorías, la robótica se ha ido presentando en todos los ámbitos, pero un ámbito el cual no ha recibido mucha atención es el musical en especial para interpretar melodías en el piano a continuación se presentan algunos ejemplos:

A nivel mundial dentro del ámbito de la robótica aplicada a la música, existen prototipos como es el caso de Taiwán en el año 2014 en la Minghsin University of Science and Technology, en la cual se desarrolló dos manos conformadas por cinco dedos cada una, las cuales son capaces de desplazarse de forma lineal a lo largo del piano dependiendo de su programación, y de esta forma poder entonar una melodía establecida (Jen-Chang et al. 2014).

En Suecia en el KTH Royal Institute Of Technology en el año 2019, fue desarrollado el robot Mozart2000, capaz de tocar ocho teclas del piano dado que no posee la capacidad de moverse de forma lineal, pero es capaz de procesar imágenes mediante una cámara y de esta manera poder tocar las notas que puede visualizar de una partitura (PHAN, MALM 2019). En Estados Unidos en el año 2022 en Massachusetts Institute of Technology fue desarrollado un prototipo el cual consta de tres dedos, los cuales pueden realizar un pequeño desplazamiento lineal y cada dedo posee dos grados de libertad para poder tocar las notas musicales previamente programadas (Ornelas 2022).

A nivel regional se han desarrollado prototipos como es el caso de la Universidad Rafael Belioso Chasin de Venezuela en el año 2008, el cual dispone de una mano robótica conformada de tres dedos los cuales son movidos a base de motores y engranajes, además esta mano posee un desplazamiento lineal para tocar las diferentes teclas del piano (Pérez, Mendoza 2008).

En el año 2012 en la Universidad Pedagógica y Tecnológica de Colombia se realizó un Robot educativo de bajo costo que interpreta melodías en piano, simulando el movimiento de un solo dedo, en el cual el propósito de este robot es servir como instrumento de prácticas en materias relacionadas con fundamentos de robótica, el cual es un brazo de tipo antropomórfico el mismo que posee cuatro grados de libertad los cuales proporcionan movimientos básicos e independientes (Alberto Buitrago Suescún, Consuelo Prieto Ramírez, Augusto Peña Cortés 2012).

A nivel nacional no se encontró información referente a este tipo de robots que sean capaces de tocar el piano; así que al ver que a nivel mundial la robótica se está implicando en todos los

ámbitos, y ya no solo en el industrial, al ser la música uno de campos que ha desarrollado menos investigaciones se presentará este proyecto el cual es el diseño y construcción de manos robóticas programables aplicadas al área del entretenimiento en el campo musical., entrando así en los nuevos planes que tiene el gobierno para promover las invenciones como lo es el “Código INGENIOS” (Secretaría de Educación Superior, Ciencia, Tecnología e Innovación sin fecha), y según el código orgánico de la economía social de los conocimientos art.4 el cual uno de sus principios dice: “El Estado propiciará el entorno favorable para la expansión y fortalecimiento de las actividades artísticas y culturales, incentivando, principalmente, la libre creación; la investigación en el arte y la cultura, con enfoque de igualdad de género y no discriminación; así como, la interacción de éstas con las otras actividades de la economía social basada en los conocimientos, la creatividad y la innovación” (Asamblea Nacional 2016).

1.3 Justificación Aplicativa

Este proyecto propone el diseño y construcción de un prototipo de manos robóticas programables con la capacidad para tocar el piano, el cual consiste en la interpretación de canciones; este está formado por dos manos y para su funcionamiento consta de la etapa de adquisición de datos la cual obtiene información sobre la posición inicial de cada mano, la etapa de actuación consiste en realizar los movimientos de los dedos y el desplazamiento lineal de las manos.

Para la etapa de control consta de dos tarjetas de desarrollo una para cada mano, las cuales envían señales a la etapa de actuación para realizar los diferentes movimientos y además reciben la información de la etapa de adquisición de datos para su posicionamiento, en la etapa de comunicación inalámbrica consta de un módulo el cual permite establecer la comunicación entre la tarjeta de desarrollo y la base de datos que contiene la información de las canciones a ser tocadas por el prototipo. Para la etapa de alimentación el prototipo consta de una fuente de alimentación para poder obtener una constante alimentación hacia el prototipo.

En la etapa de intervención con el usuario consta de una aplicación móvil amigable con el usuario para que pueda escoger las canciones para que el prototipo pueda interpretarlas. Para la implementación del prototipo se considera que debido al tiempo que se dispone para su realización este deberá tener la capacidad de interpretar el fragmento de diferentes canciones empleando las teclas blancas y además de tener la capacidad de tocar con precisión las diferentes notas en un teclado.

1.4 Objetivos

Objetivo General

Diseño y construcción de manos robóticas programables aplicadas al área del entretenimiento en el campo musical.

Objetivo Específicos

- Estudiar los parámetros que intervienen para la interpretación de la música y que procesos se debe seguir para reproducirla mediante procesos automatizados.
- Definir los requerimientos que debe cumplir el prototipo para que realice la función de un robot pianista.
- Definir el diseño que permite cumplir con los requerimientos establecidos para la construcción del prototipo.
- Seleccionar el *hardware* y *software* que permite implementar el diseño propuesto.
- Evaluar si el prototipo implementado cumple con los requerimientos planteados.

1.5 Metodología

Se presenta los métodos y técnicas que se utilizarán para el desarrollo del presente trabajo de titulación:

- *Revisión documental*: para conocer la información respectiva acerca de la construcción del prototipo, se realiza la respectiva indagación en artículos, revistas, libros, etc.
- *Histórico-lógico*: empleado para estudiar los elementos electrónicos para la movilidad del prototipo.
- *Sistematización*: para aplicar la información bibliográfica y experimental al diseño del sistema de control.
- *Análisis y síntesis*: para evaluar los resultados, llegar a conclusiones y recomendaciones.
- *Experimentación*: para realizar la comprobación de los diversos bloques que conforman el prototipo.
- *Medición*: para la evaluación de las variables durante la implementación práctica de la investigación.
- *Observación*: para la validación del prototipo diseñado.

CAPÍTULO II

2. MARCO TEÓRICO

En el presente capítulo se realiza una revisión bibliográfica para comprender como está constituido el piano, cual es la escala en que se interpreta la música, el ancho que presenta una tecla, además se analiza los diferentes diseños que presentan proyectos relacionados con robots pianistas, se establece una arquitectura para el prototipo en el cual se consideran las tarjetas de desarrollo, actuadores, sensores, alimentación, tecnologías de comunicación y base de datos.

2.1 El piano

Es un instrumento que se toca mediante un teclado que está constituido normalmente por 88 teclas las cuales están distribuidas entre blancas y negras, además se puede obtener más de 7 octavas, las teclas poseen un tono en específico las cuales tienen frecuencias más altas hacia el extremo derecho del teclado y frecuencias más bajas hacia la izquierda (PHAN, MALM 2019). Por lo general posee una dimensión de 127 cm de largo, el ancho estándar de las teclas blancas es de 2.2 cm, sus teclas negras son de 1.37 cm, una tecla de un teclado estándar requiere de 1.8 Kg.cm de fuerza actuando sobre su cara (Cunningham et al. 2018).

En la mayoría de las partituras modernas, los acordes se escriben utilizando el sistema de notación anglosajón, es un sistema muy sencillo, puesto que las siete notas se nombran con las siete primeras letras del alfabeto siendo: La, Si, Do, Re, Mi, Fa y Sol reemplazadas por A, B, C, D, E, F y G respectivamente (Pawlak, Pawlak 2018). En la Ilustración 2-1 se puede observar un ejemplo de un teclado con esta nomenclatura.



Ilustración 2-1. Nombres de las teclas de un piano

Fuente: (Pawlak, Pawlak 2018)

Escala de DO. Es una de las escalas más básicas y fundamentales de la música. Es una escala mayor donde constan de siete notas: Do, Re, Mi, Fa, Sol, y Si. Cuando se toca el piano en la escala DO, solo se usan las teclas blancas, lo que significa que solo se tocan las notas naturales de la escala DO, sin tocar las teclas negras que se les denomina sostenidos o bemoles. Esta también es una forma útil de comprender la teoría básica de la música, como la relación entre las notas de

escala y el tono en que se reproduce (Henaio et al. 2003). Para la implementación del prototipo se considera un teclado de 49 teclas dentro de las cuales contiene 4 octavas, según la información obtenida el ancho de una tecla blanca es de 2.2 cm, una octava está constituida por las siete notas musicales por lo cual una octava tendrá una distancia de 15.4 cm, esto se calcula mediante la Ecuación 2-1.

Ecuación 2-1. Distancia entre teclas.

$$D = 2.2 \text{ cm} * n$$

donde:

D, es la distancia por recorrer en centímetros.

n, es el número de teclas a recorrer.

2.2 Proyectos de robots pianistas

Se presenta algunos de los proyectos más representativos referentes a robots pianistas a nivel mundial y regional.

2.2.1 *Diseño de controles y hardware de dedos robóticos para tocar el piano*

El proyecto realizado en el Instituto de Tecnología de Massachusetts en Estados Unidos en el año 2022 presenta una propuesta de mano robótica la cual consta de tres dedos, cada uno de ellos posee un grado de libertad para su movimiento y son capaces de presionar un conjunto de tres teclas, y además pueden alcanzar altas aceleraciones (Ornelas 2022).



Ilustración 2-2. Diseño de controles y hardware de dedos robóticos para tocar el piano

Fuente: (Ornelas 2022)

Arquitectura. – Utilizan motores CC con escobillas de la serie Pololu 25D para su movimiento, para su sistema de control se emplea el microcontrolador Teensy 4.0, dado que posee la suficiente

capacidad de procesamiento para este proyecto, usan el software Matlab, la serie de notas de cada dedo son transcritas en un array con la dinámica musical con la que se pulsa la tecla y la duración del tiempo, además emplea tres sistemas de control los cuales son: control de velocidad, control de sujeción para no generar fuerza innecesaria y el control de posición mediante un PID. Para su validación se considera la velocidad con la cual son capaces de presionar las teclas; otra prueba realizada es la demostración del sistema al momento de interpretar una canción, para esto se toma los primeros treinta segundos de la canción y además se considera la ayuda de un profesional para validar su precisión (Ornelas 2022).

2.2.2 *Pianista robótico haciendo hablar a los pianos*

El proyecto realizado en el Instituto de Música Electrónica y Acústica Graz en Austria en el año de 2011 presenta una estructura formada de 88 dedos electromecánicos los cuales se posicionan a lo largo del piano, además emplea un mapeo de voz para llegar a la notación de la canción a tocar en tiempo real (Ritsch 2011).



Ilustración 2-3. Pianista robótico haciendo hablar a los pianos

Fuente: (Ritsch 2011)

Arquitectura. – Utiliza las placas de microcontroladores Escher, el microcontrolador empleado es el dsPIC33F708MC, el cual posee un controlador de 16 bits con 160 MIPS y posee un controlador Ethernet, posee tres placas Escher de las cuales dos son empleadas como esclavas y una como maestro, su comunicación se realiza mediante Ethernet. Para el movimiento de los dedos se emplean solenoides para su control cada Escher controla dos placas EscherFET los cuales constan con amplificadores FET de 16 canales, cada FET soporta una frecuencia de conmutación de 100 MHz y una corriente de 4 – 20 A. Principalmente realiza un análisis de audio con FFT para extraer las notas a ser tocadas en el piano, para su reproducción se emplean archivos MIDI. El lenguaje de programación empleado es Pure Data el cual puede adaptarse e integrarse fácilmente a otros proyectos, el firmware fue escrito en lenguaje ensamblador para tener un desempeño eficiente con el microcontrolador, con Escher se utiliza lenguaje C con inclusión de ensamblador para

tareas de tiempo crítico. Para su validación se considera la interpretación de dos canciones para verificar su precisión y que estas sean entendibles (Ritsch 2011).

2.2.3 *El tercer pulgar robótico supernumerario para tareas musicales*

El proyecto realizado en la Universidad Imperial de Londres en el año de 2018 presenta una prótesis de dedo robótico con dos grados de libertad permitiendo un movimiento horizontal y vertical el cual pretende ayudar a mejorar el alcance de una persona al momento de interpretar una canción en un piano en tiempo real (Cunningham et al. 2018).



Ilustración 2-4. El tercer pulgar robótico supernumerario para tareas musicales

Fuente: (Cunningham et al. 2018)

Arquitectura. – Posee sensores los cuales envían las señales de activación a los motores de 9V DC, los motores poseen una relación de 16:1 además están equipados con codificadores ENX10 EASY 256IMP para la lectura de posición y velocidad, los controladores de los motores son los EPOS2 24/2 DC, lo que permite el control de la corriente, la posición y la velocidad. Para el movimiento horizontal emplea un sensor de flexión el cual recibe información del pulgar y su salida está conectada a la entrada analógica del controlador del motor EPOS2 24/2, para el movimiento vertical emplea un acelerómetro analógico ADXL335 para detectar la flexión dorsal del pie y su salida está conectada a la entrada analógica del controlador del motor EPOS2 24/2, los controladores de los motores están conectados a una PC a través de USB. Para que el dispositivo pueda adaptarse a los movimientos del usuario en tiempo real emplea un filtro Kalman para evaluar la covarianza del error. El control del sistema robótico ejecuta bajo un algoritmo en lenguaje C# en el software Visual Studio, el cual procesa la información de los sensores para el movimiento de los motores, la velocidad del movimiento horizontal posee un control proporcional. Para su validación se considera su rango de trabajo y su tiempo respuesta (Cunningham et al. 2018).

2.2.4 Diseño de una mano robótica para tocar el piano

El proyecto realizado en la Universidad de Ciencia y Tecnología de Minghsin en Taiwán en el año de 2014, esta investigación se centra en el control mecánico y eléctrico de un robot que toca el piano. El diseño se basa en una mano antropomórfica formada por cinco dedos, los cuales poseen acciones de rotación, presión y levantamiento (Jen-Chang et al. 2014).



Ilustración 2-5. Diseño de una mano robótica para tocar el piano

Fuente: (Jen-Chang et al. 2014)

Arquitectura. – Está compuesto por controladores basados en computadora, dos motores lineales sin núcleo de la serie LMX1E-C, están controlados por un controlador, cada dedo está controlado por un motor paso a paso, servomotores, un cilindro neumático, guía lineal y un dispositivo de transmisión flexible. El motor paso a paso, el dispositivo neumático y el motor lineal son controlados por el sistema a través de controladores FPGA, cada nota se genera cuando el robot toca la tecla en cuatro etapas: movimiento de la palma, presión con los dedos, sujeción con los dedos, y liberación de los dedos. Al recibir la información de la nota musical y el ritmo, el sistema determinará la siguiente posición de destino para la reproducción de música en función de la información de la nota musical. Para su validación las pruebas se centran en la tensión que se genera en el mecanismo al momento de tocar las teclas del piano, otra prueba fue observar si el levantamiento de los dedos se realiza correctamente para de esta manera evitar contactos laterales con las teclas al momento de desplazarse (Jen-Chang et al. 2014).

2.2.5 Brazo robótico para tocar el piano

El proyecto realizado en el Instituto Politécnico de Worcester en Estados Unidos en el año de 2019 presenta la implementación de cuatro dedos en un brazo industrial ABB para tocar el piano usando procesamiento de audio y técnicas de aprendizaje automático mediante redes neuronales recurrentes, dadas las dimensiones del brazo utiliza un teclado de cuarenta y nueve teclas para la interpretación de las melodías (Topper, Maloney 2019).

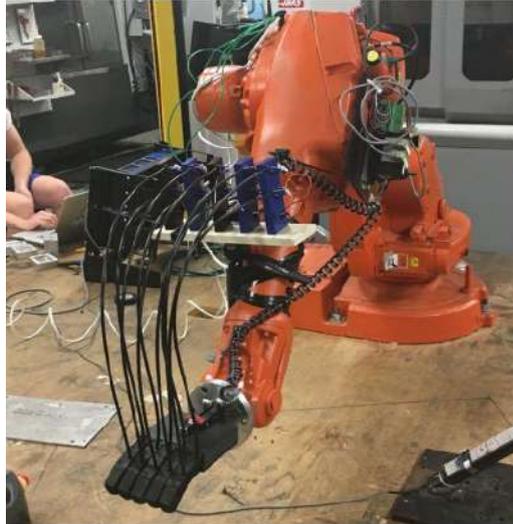


Ilustración 2-6. Brazo robótico para tocar el piano

Fuente: (Topper, Maloney 2019)

Arquitectura. – Utiliza el brazo industrial ABB IRB 1600-6/1.45 dado que es un brazo muy exacto y preciso. Para su control utiliza tres controladores: un controlador maestro, un controlador para la mano empleando Arduino y un controlador para el brazo el cual envía los comandos al robot ABB. El controlador maestro emite todas las instrucciones de alto nivel a los controladores de la mano y el brazo, el controlador de la mano controla los servos mediante disparadores de alto y bajo según los tiempos requeridos y el controlador del brazo utiliza un protocolo de comunicación basado en ASCII para que exista una comunicación con el robot ABB. Todos los servos se conectan a una placa controladora de servo Adafruit PCA9685 la cual posee un PWM de 12 bits y 16 canales. Emplean un procesamiento de entrada de audio para luego aplicar un método de aprendizaje automático mediante una red neuronal recurrente. Para su validación las pruebas realizadas se enfocan en la velocidad y tiempos al momento de entonar las notas, para esto se emplea fragmentos de treinta segundos de canciones interpretadas por un profesional (Topper, Maloney 2019).

2.2.6 Robot que lee música y toca el piano

El proyecto realizado en el KTH Real Instituto de Tecnología en Suecia en el año de 2019, presenta una estructura formada por ocho dedos electromecánicos movidos por solenoides los cuales están ubicados en una parte específica del piano, para interpretar canciones solo con ocho teclas del piano. Emplean procesamiento de imágenes para identificar las notas musicales, pero no considera su valor rítmico (PHAN, MALM 2019).



Ilustración 2-7. Robot que lee música y toca el piano

Fuente: (PHAN, MALM 2019)

Arquitectura. – Para el movimiento de los dedos emplean solenoides los cuales poseen transistores de potencia para cambiar las señales electrónicas entre cero y un valor específico, posee diodos para eliminar picos repentinos de voltaje por los cambios de conmutación, para su control emplea una tarjeta de desarrollo Raspberry Pi, como dispositivo de almacenamiento es empleada una tarjeta microSD de 16 GB, para captar las imágenes emplea un módulo de cámara de Raspberry Pi. El algoritmo se basa en encontrar coordenadas para líneas de pentagrama y notas usando procesamiento de imágenes. La programación es realizada en Python con algunas funciones extraídas de la librería OpenCV. Para su validación las pruebas realizadas se enfocan en ver la precisión que posee la cámara y la eficiencia que esta presenta al momento de captar las imágenes en diferentes entornos, para después entonar las notas que esta capte (PHAN, MALM 2019).

2.2.7 Desarrollo de un novedoso robot de piano a dos manos

El proyecto realizado en la Universidad Nacional de Ciencia y Tecnología de Taiwán en Taiwán en el año de 2017 presenta dos manos robóticas que poseen dieciséis dedos cada una, un sistema de visión para reconocer partituras y tocar el piano en tiempo real, este recibe las señales de una computadora host para luego accionar los solenoides que poseen los dedos (Fahn, Tsai, Lin 2017).



Ilustración 2-8. Desarrollo de un novedoso robot de piano a dos manos

Fuente: (Fahn, Tsai, Lin 2017)

Arquitectura. – El sistema de brazo está impulsado por dos motores lineales, dos controladores de motor de CA, además este posee treinta y dos solenoides tubulares y dos controladores para el movimiento de los dedos y para su sistema de visión emplea una cámara PTZ para captar las imágenes de las partituras. Posee un sistema de control para evitar colisiones entre las manos, la adquisición de la información para las notas musicales se realiza mediante técnicas de procesamiento de imágenes y reconocimiento de patrones, para su programación se usa lenguaje C++ en el software Borland C++. Para su validación se realiza pruebas para ver su capacidad de lectura de partituras y su interpretación en tiempo real (Fahn, Tsai, Lin 2017).

2.2.8 Robot educativo que interpreta melodías en piano

El proyecto realizado en la Universidad de Pamplona en Colombia en el año 2012 presenta una propuesta de brazo antropomórfico que consta de cuatro grados de libertad con fines educativos para interpretar melodías en piano para ser utilizado en prácticas de cinemática directa y cinemática inversa (Alberto Buitrago Suescún, Consuelo Prieto Ramírez, Augusto Peña Cortés 2012).



Ilustración 2-9. Robot educativo de bajo costo, interpreta melodías en piano

Fuente: (Alberto Buitrago Suescún, Consuelo Prieto Ramírez, Augusto Peña Cortés 2012)

Arquitectura. – Para su movimiento consta de cuatro servomotores los cuales son uno por cada grado de libertad, además posee una tarjeta SERCOM para controlar los servomotores, la comunicación con la PC se realiza mediante USB. Para el control de los movimientos de los servomotores se utiliza el software Matlab en su programación consta de una matriz donde están almacenadas las notas y para los movimientos puede ser por cinemática directa o inversa. Para su validación se realizaron pruebas de exactitud al momento de interpretar una canción infantil (Alberto Buitrago Suescún, Consuelo Prieto Ramírez, Augusto Peña Cortés 2012).

2.2.9 *Desarrollo de una mano robótica con capacidad de manipulación*

El proyecto realizado en la Universidad Rafael Beloso Chacín en Venezuela en el año de 2008 presenta una mano robótica con tres dedos con la finalidad de tocar piezas musicales en un piano eléctrico su desarrollo de hardware y software esta realizado en base a lenguaje VHDL (Pérez, Mendoza 2008).



Ilustración 2-10. Desarrollo de una mano robótica con capacidad de manipulación

Fuente: (Pérez, Mendoza 2008)

Arquitectura. – Está formado por un sistema de engranajes y resortes, servomotores, motores de corriente continua, transistores y para su desplazamiento consta de riel de impresora Epson 1170, como controlador se emplea el PIC16F84A el cual está encargada de enviar la información a los actuadores para ejecutar los movimientos, posee un sensor en el riel la cual sirve para localización, como pantalla selectora de canciones se emplea el PIC 508B el cual se comunica con el PIC16F84A, además posee una memoria donde esta almacenada la información de las canciones. Para su validación se realiza pruebas interpretando las canciones almacenadas en la memoria (Pérez, Mendoza 2008).

2.2.10 *Análisis de los elementos utilizados en los proyectos de robot pianista*

Para los actuadores se observa que los empleados en los trabajos previos son: cilindros neumáticos, solenoides, motores y servomotores. Se considera que los cilindros neumáticos generan demasiado ruido lo que dificultaría escuchar el sonido del piano, al utilizar solenoides se puede generar problemas de potencia lo que puede aumentar la temperatura de sus elementos además su estructura para la implementación del prototipo necesita de más espacio por lo que se perdería el objetivo de que la estructura debe ser similar a la forma de manos y la utilización de servomotores y motores es la consideración más viable dado que no generan demasiado ruido y se puede mantener una estructura similar a las manos al momento de implementar el prototipo.

Para el control y el procesamiento se observa que los empleados en los trabajos previos son: PIC, PC y tarjetas de desarrollo. Se considera que la utilización de una PC resulta incomoda dado que debe mantenerse conectada constantemente al prototipo dificultando la movilidad de este, la utilización de PIC resulta poco productiva, dado que para realizar una comunicación inalámbrica se debe considerar módulos adicionales y la utilización de tarjetas de desarrollo es la consideración más adecuada dada que ya poseen módulos de comunicación inalámbrica incorporados y facilita su movilidad.

Para el posicionamiento en los trabajos previos estos emplean sensores. De igual forma para la realización del prototipo se utilizará sensores para determinar su posición inicial. Para el almacenamiento de las líneas de programación de las canciones en su mayoría de los proyectos mencionados emplean tarjetas MicroSD o en su defecto se mantiene en un PC el cual posee la programación dado que se mantiene conectada constantemente al robot para la selección de las canciones.

En base al análisis realizado en párrafos anteriores se llega a la siguiente arquitectura general de robot pianista la cual está conformada por los bloques de alimentación, tarjeta de desarrollo, actuadores, sensores y almacenamiento como se observa en la Ilustración 2-11.

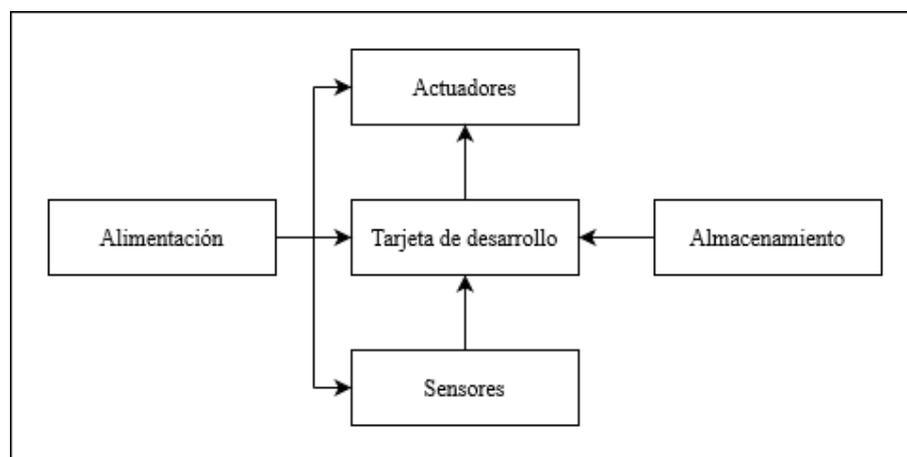


Ilustración 2-11. Arquitectura general de un robot pianista

Realizado por: (Aguirre Marco y Meneses Jean 2023)

2.3 Arquitectura de un robot pianista empleada para el prototipo

Según el análisis de la arquitectura de los robots estudiados se puede determinar que las tarjetas de desarrollo con más reincidencia son las que poseen un procesamiento de 32 bits, en lo que respecta a los actuadores se considera que para el desplazamiento lineal debe poseer la suficiente precisión para tocar las teclas de manera acertada además se debe considerar que el prototipo debe mantener la estructura de manos, para la selección de las canciones se consideró una aplicación

móvil para evitar la utilización de una PC, para esto se utiliza una comunicación inalámbrica y además se utiliza una base de datos que contiene la información de las canciones aprovechando que el prototipo va a poseer una comunicación inalámbrica. Teniendo estas nuevas observaciones se llegó a una nueva arquitectura para el prototipo de robot pianista a realizar, como se puede apreciar en la Ilustración 2-12, llegando a los siguientes bloques los cuales intervienen en el desarrollo del prototipo.

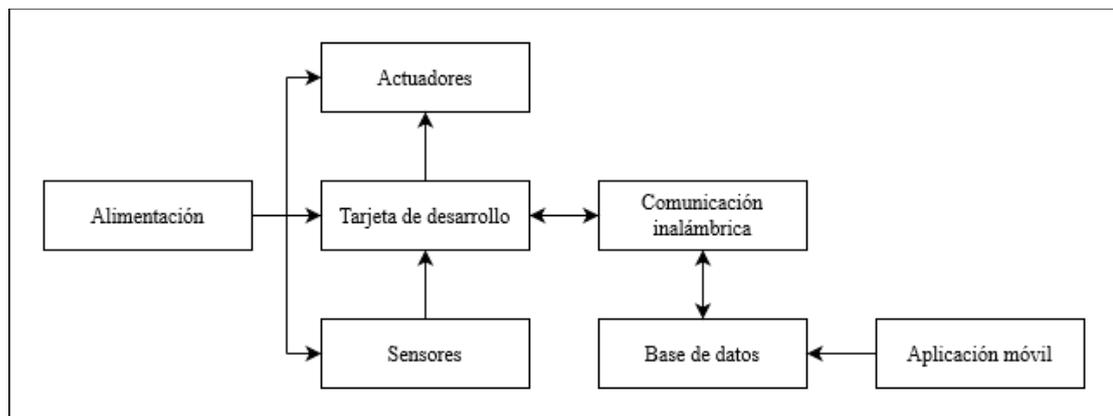


Ilustración 2-12. Arquitectura del prototipo de robot pianista

Realizado por: (Aguirre Marco y Meneses Jean 2023)

A continuación, se detalla los bloques que intervienen en la arquitectura propuesta para el desarrollo del robot pianista.

2.4 Tarjetas de desarrollo

Estas son utilizadas principalmente para el desarrollo de sistemas embebidos, cada una de estas placas cuentan con su propio lenguaje de programación o un sistema operativo (Rodríguez 2019). Se basan en microcontroladores además incorporan gran parte de módulos adicionales, de forma que cuando se necesite realizar un prototipado rápido solo se tendrá que conectar y programar (Alva, Alcorta 2020).

A continuación, se enlista algunas de las tarjetas de desarrollo de hardware más utilizadas: (Alva, Alcorta 2020).

- Arduino. Son placas de desarrollo basadas en los microcontroladores AVR. Son fabricadas por la compañía Arduino. Existen muchas placas clónicas.
- Teensy. Son placas de desarrollo basadas en diversos microcontroladores, fabricadas por la empresa PJRC. COM LLC. Muchas de sus placas usan arquitectura ARM.
- STM32xxxxDISCOVERY. Son placas de desarrollo que utilizan como base los microcontroladores STM32.

- Blue Pill. Son placas de Desarrollo basadas en el uC STM.
- Wemos. Son placas de desarrollo basadas en el chip ESP8266.
- NodeMCU. Son placas de desarrollo basadas en el chip ESP8266. Son similares a las placas Wemos, pero poseen más pines.
- NodeMCU-32. Son placas de desarrollo basadas en el microcontrolador ESP32.

2.4.1 Selección de tarjeta de desarrollo de hardware

Teniendo en cuenta la revisión bibliográfica se obtiene la Tabla 2-1 la cual presenta una comparativa de las tarjetas de desarrollo más populares y basándose en la clasificación mencionada anteriormente se consideran las familias de Arduino, Teensy y NodeMCU.

Tabla 2-1. Comparativa de tarjetas de desarrollo

| Características | Arduino | Teensy | ESP8266 | ESP32 |
|-----------------------------|--|---|------------------------|--|
| Microcontrolador | Intel Curie, ARM Cortex-M0, ATSAM3X8E, SAMD21 Cortex-M0+ | ARM Cortex-M0+, ARM Cortex-M4, ARM Cortex-M7 | Tensilica Xtensa LX106 | Tensilica Xtensa LX6 Dual-Core |
| Procesador | 32 bits | 32 bits | 32 bits | 32 bits |
| Memoria SRAM | 24KB – 96KB | 8KB – 1024KB | 32KB – 96KB | 520KB |
| Frecuencia de reloj | 32MHz – 84MHz | 48MHz – 600MHz | 80MHz – 160MHz | Doble núcleo de 160MHz – 240MHz cada uno |
| Voltaje de operación | 3.3V – 5V | 3.3V – 5V | 3.3V | 3.3V |
| Conectividad | Wi-Fi, Bluetooth, USB, SPI, UART, I2C, TWI | UART, I2C, SPI | Wi-Fi, UART, I2C, SPI | Wi-Fi, Bluetooth, UART, SPI, I2C |
| Memoria Flash | 196KB – 512KB | 62KB – 1984KB | 512KB – 4MB | 4MB – 16MB |
| Pines | I/O digitales 8 – 54 Entadas analógicas 6 – 12 | I/O digitales 27 – 64 Entadas analógicas 13 – 27 | 17 GPIO | 34 GPIO |
| Consumo de energía | 15mA – 130mA | 15mA – 250mA | 80mA – 225mA | 80mA – 225mA |
| Costo | \$44.60 – \$73.55 | \$12.85 – \$35.65 | \$7.99 – \$35.08 | \$9.99 – 44.99 |

Fuente: (Novillo et al. 2018) (Arias 2016) (PJRC 2023) (Gonzalez 2017) (Ikiss 2020) (Guerra 2020)

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En base a la Tabla 2-1 se toma en cuenta para la implementación del prototipo la tarjeta de desarrollo ESP32 de la familia NodeMCU dada que está ya posee una comunicación inalámbrica Wi-Fi la cual nos ayuda a cumplir con ese objetivo sin la necesidad de módulos adicionales, además posee un procesamiento adecuado y una frecuencia de reloj de doble núcleo la cual nos ayuda a realizar tareas en paralelo.

2.5 Actuadores

Existen magnitudes físicas que pueden ser transformadas en otras magnitudes mediante propiedades que pueden interactuar con el entorno, las variables involucradas con estas magnitudes pueden ser de fuerza, posición, velocidad y aceleración, para lo cual existen dispositivos que pueden realizar las transformaciones de dichas magnitudes y se les conoce como actuadores, una clasificación detallada de los actuadores se la puede observar en (Corona Ramírez, Abarca Jiménez, Mares Carreño 2014). Se seleccionó actuadores eléctricos dado que son instrumentos conformados por motores eléctricos y reductores los cuales permiten el accionamiento de cualquier dispositivo para poder realizar movimientos o acciones necesarias para el funcionamiento del prototipo.

2.5.1 Motor paso a paso

También conocido como convertidor electromagnético o simplemente a pasos, son motores de corriente directa, pueden ser de imán permanente o reluctancia variable, los cuales destacan como característica el rotar en sus dos direcciones, giran con incrementos angulares precisos, soportan un torque de retención a velocidad cero que puede ser controlada por medio de circuitos digitales, el posicionamiento de estos motores lo realizan con gran precisión y son capaces de no usar sensores de retroalimentación (Acosta et al. 2021).

Los movimientos que puedan realizar los motores paso a paso dependen de las características constructivas, comprendiendo ángulos de entre menos 1° hasta 15° o más, dependiendo del modelo. Dado que los motores a paso tienen una elevada capacidad de posicionamiento que los hace ideales para obtener un control exacto en dirección, velocidad y posicionamiento (Conti 2005).

Tabla 2-2. Comparativa entre motores paso a paso lineales

| Características | Motor paso a paso lineal de riel | Motor paso a paso de eje lineal |
|-----------------------|----------------------------------|---------------------------------|
| Serie | FSK30 | FSL120 |
| Rango de carrera | 50 - 300 mm | 50 – 1500 mm |
| Precisión de posición | 0.05 mm | 0.02 mm |
| Máxima velocidad | 300 mm/s | 500 mm/s |

| Características | Motor paso a paso lineal de riel | Motor paso a paso de eje lineal |
|------------------------|---|--|
| Carga máxima | 3 kg | 100 kg |
| Precio | \$83 - \$108 | \$736 - \$1308 |

Fuente: (Fuyumotion 2010a), (Fuyumotion 2010b)

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En base a la Tabla 2-2, se puede distinguir distintos rasgos para la elección de un motor paso a paso lineal para el movimiento de las manos robóticas en el eje x, dentro de las características presentadas se puede llegar a distinguir que los motores paso a paso de eje lineal son los más adecuados, pero debido a su elevado precio se selecciona los motores paso a paso lineales de riel dado que sus características también son óptimas para realizar el desplazamiento en el eje x, tanto como su precisión y su velocidad máxima para realizar el desplazamiento adecuado, dado que el ancho de una tecla a otra es de 2.2 cm.

2.5.2 Servomotores

Es un motor eléctrico que puede controlar la velocidad, posición y dirección. El servomotor trabaja junto a un encoder, conocido como decodificador que es el encargado de convertir los movimientos mecánicos o giros de los ejes en pulsos digitales que son descifrados por un controlador de movimiento, también utilizan un Driver para establecer su posición, torque y velocidad. Los servomotores tienen un ángulo de giro del eje de 180° los cuales pueden ser modificados para alcanzar 360° (Arboleda et al. 2014).

Tabla 2-3. Comparativa entre la familia de servomotores

| Características | TowerPro | TopModel | Lewansoul |
|-------------------------------|-----------------|-----------------|------------------|
| Serie | SG90 | SMS1012 | LFD-01 |
| Velocidad de operación | 0.1 s/60° | 0.12 s/60° | 0.12 s/60° |
| Torque detenido | 1.8 kg.cm | 1 kg.cm | 1.8 kg.cm |
| Angulo de rotación | 180° | 180° | 180° |
| Dimensiones (mm) | 29.8x22.6x11.8 | 25.5x21.5x11.8 | 30x12x32.6 |

Fuente: (Xukyo 2018)

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En base a la Tabla 2-3, se considera la elección del servomotor Lewansoul, que es ideal para el prototipo donde se toma en cuenta sus dimensiones considerando el diseño de la mano robótica, también se analiza su torque para poder ejercer la fuerza necesaria y la velocidad de operación para tocar las diferentes teclas del piano.

2.6 Sensores

También llamados transductores, son los encargados de obtener la información de los diferentes procesos a los que se estén usando, también son capaces de convertir cualquier tipo de magnitud física, química en una magnitud eléctrica (Corona Ramírez, Abarca Jiménez, Mares Carreño 2014).

2.6.1 *Sensor magnético*

Son dispositivos los cuales tienen como función detectar y medir campos magnéticos. Estos sensores utilizan principios físicos y magnéticos para convertir la información magnética en señales eléctricas que pueden procesarse y usarse en varias aplicaciones (Pérez Chilibingua 2019).

Tabla 2-4. Comparativa entre sensores de distancia

| Sensores magnéticos | Rango de detección | Alimentación |
|-------------------------------------|---------------------------|---------------------|
| Sensor de efecto hall | 3 cm | 4.5 V – 24 V |
| Sensor ultrasónico | 35 cm – 110 cm | 18 – 30 V |
| Sensor interruptor magnético | 15 mm – 25 mm | 5 V – 100 V |

Fuente: (González Gavilanes, Carrillo Trujillo 2019)

Realizado por: (Aguirre Marco y Meneses Jean 2023)

2.6.2 *Selección del sensor*

En base a la comparativa realizada en la Tabla 2-4, se llega a la conclusión que los sensores de interruptor magnético son los más adecuados para la detección de la posición inicial del prototipo dado el rango de detección que poseen.

2.7 Alimentación

Para determinar el tipo de alimentación para nuestro proyecto técnico se considera realizar una comparativa entre fuentes de alimentación y baterías de lo cual se detalla a continuación:

2.7.1 *Fuente de Alimentación*

Es un dispositivo electrónico que transforma la tensión de red eléctrica de corriente alterna a una tensión de corriente continua filtrada, con el fin de alimentar todo tipo de equipos en el hogar e industria. Se pueden clasificar de dos tipos (Martín Castillo 2017):

- Fuentes de alimentación lineales
- Fuentes de alimentación conmutadas

Fuentes de alimentación lineales

Es el equipo encargado de proporcionar un valor adecuado de voltaje para el funcionamiento de cualquier dispositivo, se encarga de transformar o convertir la entrada de voltaje alterna de la red a un voltaje continuo el cual está constituido por varias etapas las cuales son transformación, rectificación, filtrado y regulación. Estos dispositivos tienen un rendimiento menor y su eficiencia energética es limitada (Martín Castillo 2017).

Fuentes de alimentación conmutadas

Son también conocidas como SMPS, Switch mode power supply, cuyo objetivo tienen de convertir tensión AC a tensión DC, pueden trabajar a frecuencias altas entre los 100 kHz, su principal ventaja ante las fuentes de alimentación lineales es que el tamaño del transformador disminuye al igual que las pérdidas de energía y el volumen de la fuente (Martín Castillo 2017).

2.7.2 Baterías

Son dispositivos que convierten la energía química en energía eléctrica a través de una reacción electroquímica, se utilizan en una variedad de aplicaciones, desde dispositivos electrónicos portátiles hasta vehículos eléctricos y sistemas de almacenamiento de energía renovable (Torrente Artero 2013), existen diferentes tipos de baterías para los cuales se mencionan las más conocidas:

Batería de plomo ácido

Dispositivos recargables, utilizados principalmente en vehículos y motocicletas, es la tecnología de almacenamiento más antigua. Un promedio de 500 a 1000 ciclos, la densidad de energía de 30 a 50 Wh/Kg, está marcada por su corta duración, su eficiencia está entre el 75 y el 80% (Alba Padilla, Muñoz Guillen 2017).

Batería de polímero de litio (LiPo)

Dispositivos recargables de última generación, donde su característica se basa en su peso, volumen y voltaje de desempeño con relación a otros dispositivos recargables. Estas baterías usan un polímero sólido como electrolítico, presentan una densidad de energía de 300 Wh/Kg con una mejor tasa de descarga, son ideales para aplicaciones que requieren más de 1 A (Cabezas Terán 2016).

2.7.3 Selección de la alimentación

En función a la bibliografía revisada, se genera la Tabla 2-5 donde se muestra una comparativa entre los diferentes tipos de alimentación.

Tabla 2-5. Comparativa entre fuente de alimentación y baterías

| Características | Fuente conmutada | Plomo acido | LiPo |
|---------------------|---|---|-------------------------------|
| Densidad energética | Relativamente alta | 30 a 50 Wh/Kg | 300 Wh/Kg |
| Tiempo de carga | -- | 8 – 16 h | 1 – 1.5 h |
| Duración | Constante alimentación | 100 h | 10 h |
| Eficiencia | >85% | 70 % | 90 % |
| Vida útil | 3 – 10 años | 3 – 5 años | 2 – 3 años |
| Aplicaciones | Sistemas de iluminación LED, sistemas de control y automatización, sistemas de audio, | Uso automotriz, sistemas de tracción industrial y domestico | Sistemas electrónicos móviles |

Fuente: (Martín Castillo 2017) (Alba Padilla, Muñoz Guillen 2017) (Cabezas Terán 2016)

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En base a la comparativa de la Tabla 2-5, se llega a la conclusión que el sistema de alimentación adecuado a utilizar es una fuente conmutada, dada que su densidad energética es relativamente alta, y considerando que el consumo energético del prototipo va a ser elevado debido a la cantidad de actuadores y sensores utilizados es necesario mantener una alimentación constante para que no existan caídas de potencia.

2.8 Aplicación móvil

Son programas informáticos diseñados y desarrollados para operar en dispositivos móviles, como teléfonos inteligentes o tabletas. También conocida como app, las aplicaciones móviles ofrecen funciones y servicios específicos, como juegos, redes sociales, herramientas de productividad, servicios bancarios, entre otros. Esta aplicación está diseñada para adaptarse a las características y capacidad de los dispositivos móviles, utilizando sus funciones, como GPS, cámara, acelerómetro y conexión a Internet. Las aplicaciones móviles se pueden descargar e instalar desde tiendas en línea, como Google Play Store para dispositivos Android y App Store para dispositivos iOS (Garrido Cobo 2013).

Tabla 2-6. Sistemas operativos para móviles a nivel mundial

| Sistemas Operativos | Participación en el mercado 2010 | Participación en el mercado 2022 |
|---------------------|----------------------------------|----------------------------------|
| Android | 23.3% | 71.74% |
| iOS | 15.6% | 27.63% |

Fuente: (Garrido Cobo 2013), (Muñoz Segura 2016), (Ramírez 2022)

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Tabla 2-7. Sistemas operativos para móviles a nivel de Ecuador

| Sistemas Operativos | Participación en el mercado entre 2021 y 2022 |
|---------------------|---|
| Android | 86.75 % |
| iOS | 12.93 % |

Fuente: (Medina 2022)

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 2-6, se observa la participación de los sistemas operativos móviles a nivel mundial en el mercado donde Android es uno de los sistemas más utilizados, de igual forma en la Tabla 2-7, se puede observar como el sistema operativo Android a nivel de Ecuador es superior por su versatilidad de sus aplicaciones y además por tener software libre, esto significa que existe una amplia base de usuarios potenciales para la aplicación, lo cual brindara la oportunidad de llegar a un público más amplio y aumentar la visibilidad de la aplicación, para el desarrollo la aplicación móvil se considera el entorno de desarrollo integrado Android Studio.

Android Studio

Es una herramienta libre desarrollada por Google, ofrece un conjunto completo de características y herramientas para facilitar el proceso de creación de aplicaciones de Android, además incluye un editor de código, emulador de dispositivos y muchas otras funciones que ayudan a los desarrolladores a escribir, limpiar y optimizar sus aplicaciones. Es ampliamente utilizado por programadores de Android por su interfaz amigable, la integración con otras herramientas y su capacidad para adaptarse a las últimas actualizaciones y características de Android. Por lo tanto, Android Studio es un poderoso entorno de desarrollo diseñado para agilizar y simplificar el proceso de crear aplicaciones móviles para dispositivos Android (Santillán Rivadeneira, Maldonado Jaramillo 2017).

2.9 Bases de datos en la nube

Esta es una modalidad de almacenamiento que nos permite almacenar grandes cantidades de información de manera organizada para que luego pueda ser fácilmente encontrada y utilizada. A los servicios de almacenamiento en nube, se puede acceder por diferentes medios, como un servicio, interfaz de programación de aplicaciones, interfaz de usuario o alguna otra seleccionada por el cliente (Contreras 2022).

2.9.1 Selección de base de datos en la nube

En la Tabla 2-8 se presenta una comparativa de las principales bases de datos en la nube las cuales son: Firebase, CloudKit y AWS Amplify.

Tabla 2-8. Comparativa de base de datos en la nube

| | Firestore | CloudKit | AWS Amplify |
|--------------------------------|---|--|---|
| Funciones | Bases de datos en tiempo real | Gestionar datos remotos | Configurar backends sin servidor con autenticación |
| | Autenticación de usuarios | | Conectar aplicaciones móviles y web a los recursos de Amazon Web Services |
| | Almacenamiento en la nube | Almacenamiento en la nube | Implementar y alojar aplicaciones |
| | Seguimiento de errores | | Motor de almacenamiento |
| | Testear la aplicación en dispositivos virtuales | Permite datos estructurados y datos bulk | Ejecutar solicitudes HTTP |
| | Configuración remota | Proporciona una base de datos NoSQL en la nube | Agregar capacidades analíticas y de marketing |
| | Envío de notificaciones y mensajes | | Crear bots conversacionales |
| | Servidor para alojar las aplicaciones | Es un servicio para mover datos a y desde la nube iCloud | Crear flujos de incorporación |
| | Mediciones y análisis de los eventos | Alojar aplicaciones | |
| Plataformas compatibles | iOS, Android, Web | macOS, iOS | iOS, Android, React Native, Web |
| Ventajas | Se puede comenzar de forma gratuita. | Admite datos estructurados y masivos | Forma rápida y sencilla de crear aplicaciones |
| | Velocidad de desarrollo. | | |
| | Plataforma de desarrollo de aplicaciones integral | Desarrollo de aplicaciones | Solo cobra los servicios que se utiliza |
| | Desarrollado por Google | | |
| | Los desarrolladores pueden centrarse en el desarrollo de frontend | Se puede comenzar de forma gratuita. | La administración de la infraestructura y las aplicaciones es sencilla |
| | Es sin servidor | Privacidad | |
| | Firestore | CloudKit | AWS Amplify |
| | Ofrece capacidades de aprendizaje automático | Estabilidad | La interfaz es sencilla |
| | Genera tráfico a sus aplicaciones | | |
| | Monitoreo de errores | Seguridad | Proporciona una forma rentable de crear aplicaciones nativas |
| Seguridad | | | |
| Desventajas | El número de usuarios simultáneos y el espacio | Funciones disponibles para equipos Apple | La plataforma no es apta para gente sin experiencia |

| | | | |
|--|---|---|--|
| | de almacenamiento es limitado en el modo gratuito | | |
| | Posee dos maneras de pago: por servicios y plan mensual | Pocas opciones de personalización en el modo gratuito | No especializada para entornos regulados |

Fuente: (López 2020) (Microsoft 2022) (Correa 2021)

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En base a la comparativa de la Tabla 2-8 se escoge la base de datos en la nube Firebase, dadas las numerosas ventajas que presenta y su compatibilidad con diversas plataformas.

CAPÍTULO III

3. MARCO METODOLÓGICO

En este capítulo se presenta los requerimientos que debe cumplir el prototipo, la concepción general de la arquitectura describiendo cada una de las etapas que lo componen, diagrama de bloques, su diseño mecánico, *hardware* y *software* necesarios para el desarrollo, además se detalla las conexiones y el diagrama de flujo necesarios para la construcción del prototipo.

3.1 Requerimientos del prototipo

En base al estudio realizado en el capítulo anterior se definen los requerimientos necesarios para el diseño y construcción de manos robóticas programables aplicadas al área del entretenimiento en el campo musical, los cuales son:

- El prototipo debe ser capaz de interpretar canciones en escala Do en un teclado de 49 teclas.
- La estructura principal del prototipo está constituida por dos manos las cuales poseen cuatro dedos cada una.
- Para el movimiento se emplea motores paso a paso para cada mano y cada dedo consta de un servomotor para su accionamiento.
- Las manos del prototipo tienen la capacidad de desplazarse en el eje x para ubicarse en una posición específica para que una vez alcanzada sus dedos puedan moverse en el eje z para presionar las teclas blancas del teclado.
- Para su alimentación está conectado a la red eléctrica de 110 V, además posee un interruptor de encendido para que de esta manera pueda permanecer conectado constantemente y solo se active cuando sea necesario.
- El prototipo consta de siete etapas de proceso las cuales son: etapa de alimentación, etapa de adquisición de datos, etapa de control y procesamiento, etapa de actuación, etapa de comunicación, etapa de almacenamiento y etapa de intervención con el usuario.
- La etapa de adquisición de datos dispone de sensores magnéticos los cuales se emplean para detectar la posición inicial de cada mano.
- La etapa de actuación está formada por motores paso a paso y servomotores los cuales serán los encargados de realizar los movimientos.
- Los motores paso a paso proporcionan desplazamientos de 2.1 cm abarcando espacio entre tecla y tecla.
- Los servomotores poseen una velocidad de reacción de hasta 0.0625s para presionar una tecla, además debe poseer una fuerza de 1.8 Kg.cm.

- La etapa de comunicación se realiza de manera inalámbrica mediante Wi-Fi entre la tarjeta de desarrollo, la etapa de almacenamiento y la aplicación móvil.
- La etapa de almacenamiento consiste en una base de datos la cual contiene la información de las canciones a seleccionar.
- La etapa de intervención con el usuario dispone de una aplicación móvil la cual envía la información mediante Wi-Fi hacia la tarjeta de desarrollo de la canción a interpretar.
- Mediante la aplicación móvil el usuario puede escoger una de las canciones que están almacenadas en la base de datos.
- La etapa de control y procesamiento consiste en dos tarjetas de desarrollo ESP32, una para cada mano para que no existan conflictos en los tiempos de actuación.
- En la etapa de control y procesamiento esta la programación principal en la cual consta la cantidad del número de pasos que debe desplazarse cada mano, dependiendo de las notas a interpretar según el teclado empleado, las velocidades, la comunicación y el control de inicio de posición.
- Una de las tarjetas de desarrollo ESP32 cumple la función de maestro la cual está conectada a la mano derecha y la otra ESP32 se emplea como esclavo la cual se conecta a la mano izquierda.
- Para levantar la comunicación Wi-Fi se la realiza por medio del módulo maestro ESP32.
- El módulo maestro ESP32 es el que recibe la información de la aplicación móvil y se encarga de transmitirla al módulo esclavo ESP32 mediante comunicación serial.

3.2 Concepción de arquitectura del sistema

En la Ilustración 3-1 se visualiza la arquitectura general del prototipo de robot pianista que se compone de las siguientes etapas:

Etapa de alimentación. – esta se encuentra conectada a la red eléctrica de 110 V y por medio de la fuente de poder T-250W la transforma a 24 V con 10 A para alimentar los motores paso a paso y además se emplea una fuente de poder RS-25-5 para obtener 5 V y alimentar las tarjetas de desarrollo, servomotores y los sensores magnéticos. Se emplean estas fuentes debido a que poseen una densidad de energía relativamente alta y permite evitar que exista caídas de potencia.

Etapa de adquisición de datos. – está formado por dos sensores magnéticos los cuales son empleados para detectar la posición inicial de cada mano. Cada sensor magnético estará conectado y enviará la información a su respectiva tarjeta de desarrollo ESP32 para proceder con el procesamiento de esta información.

Etapa de control y procesamiento. – Está formada por dos tarjetas de desarrollo ESP32 que reciben, procesan e interpretan la información obtenida de la etapa de adquisición de datos y la aplicación móvil, para activar los actuadores si es necesario. Dentro de la programación de las tarjetas de desarrollo ESP32 consta la información de la cantidad de pasos que se debe mover las manos para ubicarse en cada nota y además del movimiento necesario que debe realizar el servomotor para que cada dedo presione una tecla.

Etapa de actuación. – está formada por servomotores para cada dedo, también está formada por motores paso a paso para cada mano. Estos reciben la información para su movimiento de la etapa de control y procesamiento. La selección de la canción a tocar es enviada desde la etapa de intervención con el usuario.

Etapa de comunicación. – esta se realiza de forma inalámbrica mediante Wi-Fi, el módulo de comunicación Wi-Fi se encuentra integrado en la tarjeta de desarrollo ESP32 el cual se conecta a un router para permitir el acceso a internet. Solo el módulo maestro ESP32 tiene habilitada su conexión Wi-Fi.

Etapa de almacenamiento. – en esta se encuentra almacenada la información de las canciones a interpretar, para esto se emplea la base de datos Firebase.

Etapa de intervención con el usuario. – esta posee una aplicación móvil la cual permite que el usuario pueda escoger una canción de las que se encuentran en la etapa de almacenamiento, y de esta manera enviar la selección hacia la etapa de control y procesamiento mediante comunicación Wi-Fi, para que el prototipo interprete la canción.

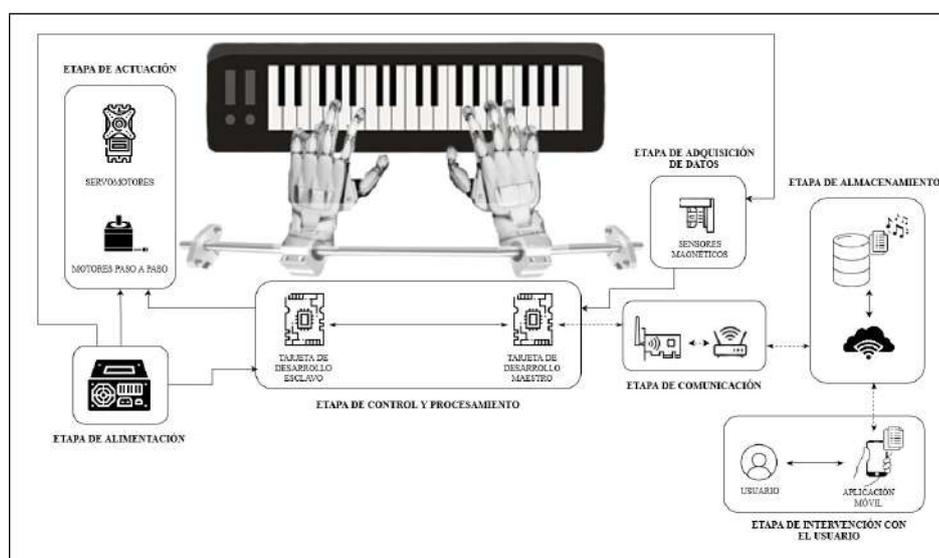


Ilustración 3-1. Concepción de la arquitectura general del prototipo de robot pianista

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.3 Diagrama de bloques del sistema

Después de realizar la concepción de la arquitectura general del prototipo, se detalla el diseño que conforma el prototipo mediante diagramas de bloques como se observa en la Ilustración 3-2 el cual consta de ocho bloques los cuales se detallan a continuación:

- Bloque de alimentación: está compuesto por dos fuentes conectadas a la red de 110 V las cuales transforman este voltaje a un voltaje de 24 y 5 V. Estos voltajes son empleados para energizar los elementos que conforman los bloques de actuadores, adquisición de datos y procesamiento.
- Bloque de actuadores: formado por servomotores y motores paso a paso estos se activan y realizan sus movimientos según las ordenes que sean enviadas desde el bloque de procesamiento.
- Bloque de adquisición de datos: compuesto por sensores magnéticos los cuales son empleados para detectar la posición inicial de cada mano y esta información es enviada al bloque de procesamiento.
- Bloque de procesamiento: está formado por un microcontrolador ESP32 para cada mano, de los cuales uno es usado como maestro y el otro como esclavo, la comunicación entre estos es de manera serial, estos recibirán la información enviada por el bloque de adquisición de datos. Además, el módulo maestro ESP32 es el encargado de realizar la comunicación Wi-Fi para lograr una conexión con los bloques de base de datos y aplicación móvil, y de esta manera recibir la información de la canción seleccionada que se desea interpretar, para luego procesar toda la información.
- Bloque de router: es el encargado de comunicar la información de los bloques de aplicación móvil y base de datos con el bloque de procesamiento.
- Bloque de base de datos: es una plataforma localizada en la nube, en la cual se encontrará la información de cada canción y está vinculada con la aplicación móvil.
- Bloque de aplicación móvil: esta me permite visualizar un listado de las canciones que se encuentran disponibles y enviar la información de mi selección hacia el bloque de procesamiento para poder interpretar la canción elegida mediante el bloque de actuadores.
- Bloque de usuario: este posee a su disposición solo la aplicación móvil para escoger cualquier canción de las que se encuentran almacenadas para su interpretación.

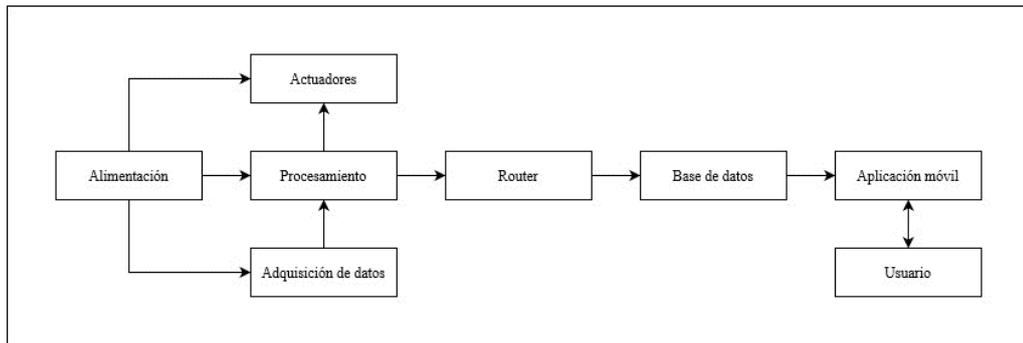


Ilustración 3-2. Diagrama de bloques del prototipo.

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.4 Diseño mecánico del prototipo

De acuerdo con los requerimientos de estructura que debe cumplir el prototipo para su funcionamiento se realiza el diseño en 3D de las manos y del tornillo lineal a emplear desarrollados en SolidWorks 2022 el cual es un software de diseño CAD 3D para modelar piezas y ensamblajes en 3D y planos en 2D, además ofrece la posibilidad de crear, diseñar, simular, fabricar y gestionar los datos del proceso de diseño (SolidWorks Corporation 2002). A continuación, se puede observar los diseños.

3.4.1 Diseño de las manos

Dada que la estructura del prototipo está formada por dos manos su diseño se encuentra dividido entre mano derecha e izquierda las cuales se detallan a continuación:

Diseño de la mano derecha

Para su diseño se considera que esta mano debe estar formada por cuatro dedos y además cada uno es activado por servomotores en las Ilustraciones 3-3 hasta la 3-13 se puede observar sus diferentes piezas junto con sus dimensiones que se encuentran en centímetros, las dimensiones de los dedos se obtienen en base a la distancia que deben alcanzar y a su ángulo de descenso.

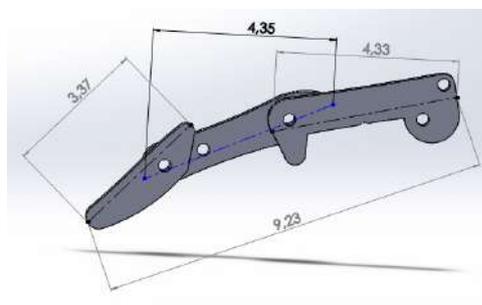


Ilustración 3-3. Estructura y dimensiones de los dedos de la mano derecha

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-3, se puede observar la estructura que posee los dedos de esta mano, en la Ilustración 3-4 se observa el modelo de los cuatro dedos creados.

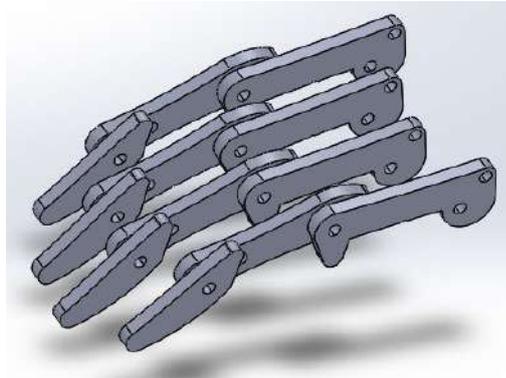


Ilustración 3-4. Creación de los 4 dedos para el ensamblaje de la mano derecha

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-5 se observa el modelo de la base la cual se emplea para la sujeción de los dedos y los servomotores.

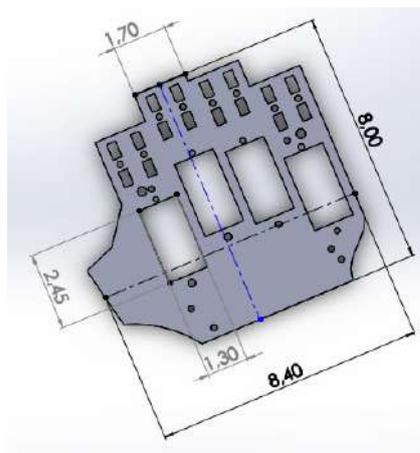


Ilustración 3-5. Base para los dedos y servomotores

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Se crea sujetadores para los dedos, como se observa en la Ilustración 3-6, se necesitan dos de estos sujetadores para cada uno de los dedos que van junto con la base y en la Ilustración 3-7 se muestra el sujetador que va a estar unido a los servomotores conjunto con los dedos, de este se emplea uno para cada dedo.

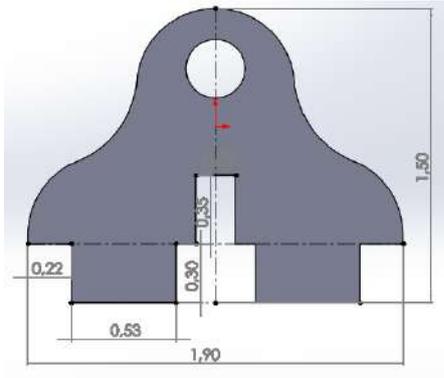


Ilustración 3-6. Sujetadores para cada uno de los dedos

Realizado por: (Aguirre Marco y Meneses Jean 2023)

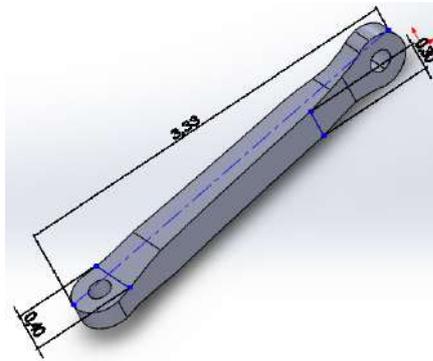


Ilustración 3-7. Sujetador entre el servo y el dedo

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Además, se crea sujetadores para los servomotores tanto frontal como posterior como se observa en la Ilustración 3-8.

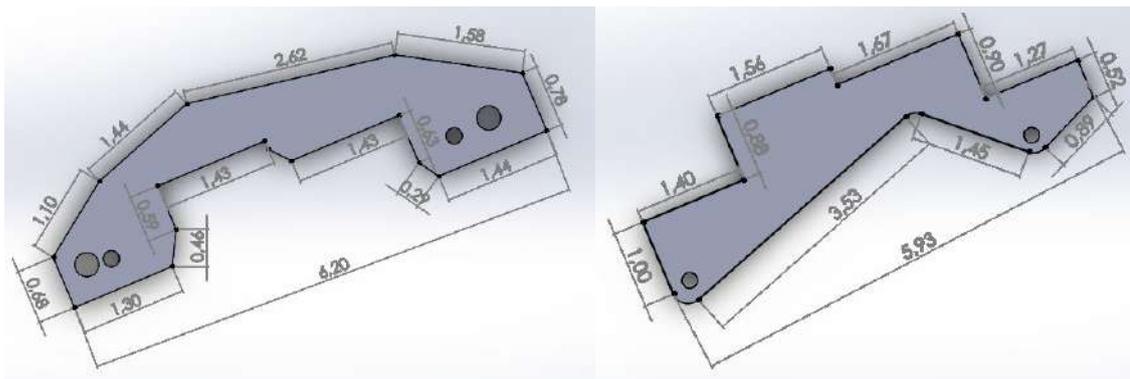


Ilustración 3-8. Sujetador para los servomotores parte frontal y posterior

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-9 se observa la base empleada para cubrir los servomotores.

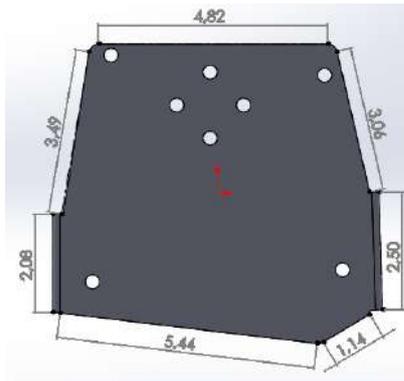


Ilustración 3-9. Base para cubrir los servomotores

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Finalmente se crea las piezas que se muestra en la Ilustración 3-10, la cuales sirven para la sujeción de la mano con el tornillo lineal.

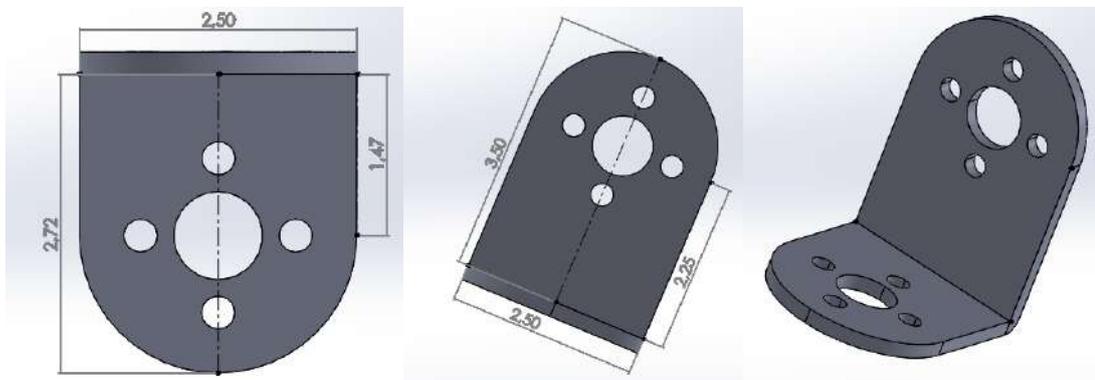


Ilustración 3-10. Piezas de sujeción

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-11 se muestra el ensamble de los dedos con base para lo que intervienen los sujetadores de las Ilustraciones 3-6 y 3-7.

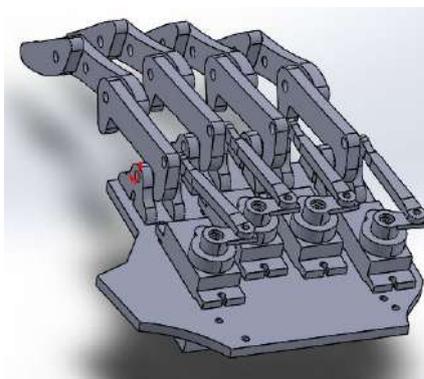


Ilustración 3-11. Ensamble de los dedos y la base

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-12 se muestra el ensamble que corresponde al sujetador frontal y posterior de los servomotores.

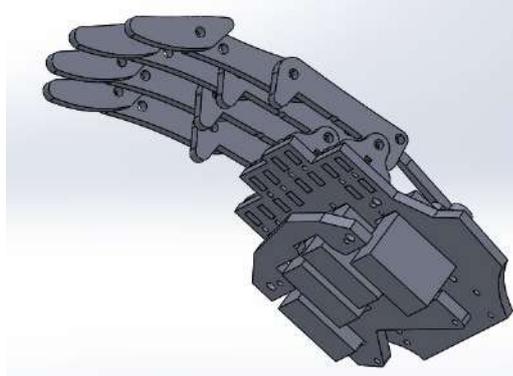


Ilustración 3-12. Ensamble de los sujetadores de los servomotores

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-13 se puede observar el ensamble completo de la mano derecha

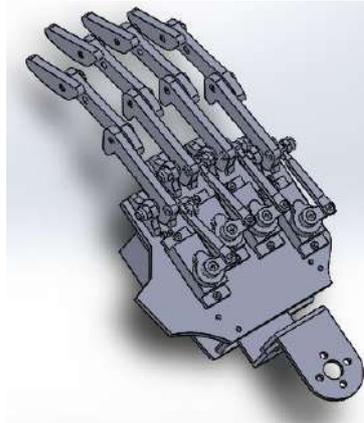


Ilustración 3-13. Ensamble completo de la mano derecha

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Diseño de la mano izquierda

Para su diseño se considera que esta mano debe estar formada por cuatro dedos y además cada uno es activado por servomotores en las Ilustraciones 3-14 hasta la 3-24 se puede observar sus diferentes piezas junto con sus dimensiones que se encuentran en centímetros, las dimensiones de los dedos se obtienen en base a la distancia que deben alcanzar y a su ángulo de descenso.

En la Ilustración 3-14, se puede observar la estructura que posee los dedos de esta mano, en la Ilustración 3-15 se observa el modelo de los cuatro dedos creados.

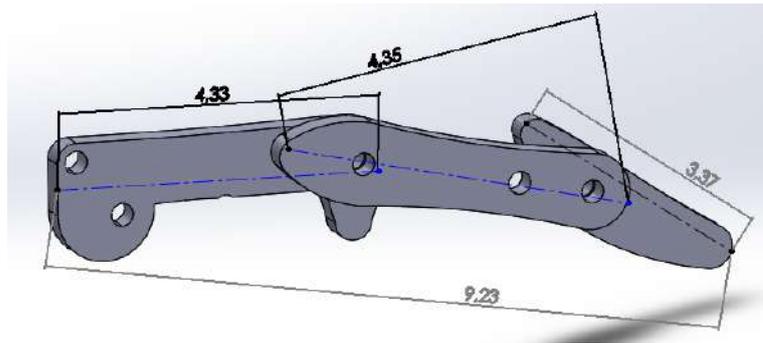


Ilustración 3-14. Estructura de los dedos para la mano izquierda

Realizado por: (Aguirre Marco y Meneses Jean 2023)

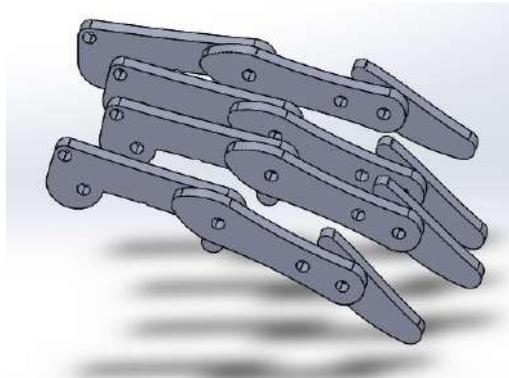


Ilustración 3-15. Creación de los 4 dedos para el ensamblaje de la mano izquierda

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-16 se observa el modelo de base usado para la sujeción de dedos y servomotores.

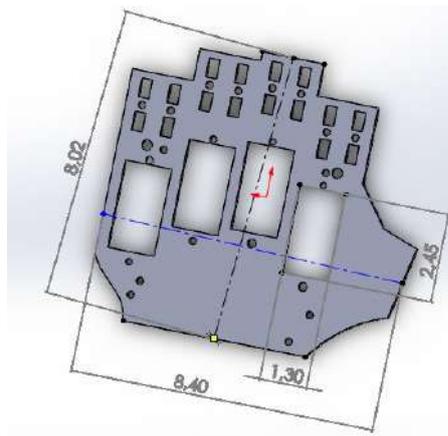


Ilustración 3-16. Base para los dedos y servomotores de la mano izquierda

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Se crea sujetadores para los dedos, como se observa en la Ilustración 3-17, se necesitan dos de estos sujetadores para cada uno de los dedos que van junto con la base.

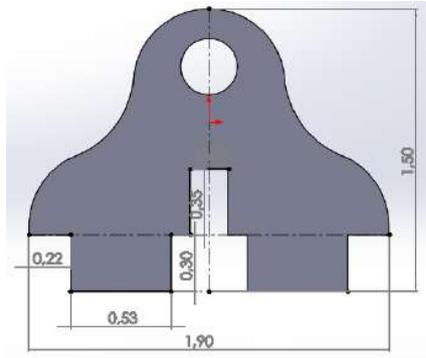


Ilustración 3-17. Sujetadores para cada uno de los dedos

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-18 se muestra el sujetador que va a estar unido a los servomotores conjunto con los dedos, de este se emplea uno para cada dedo.

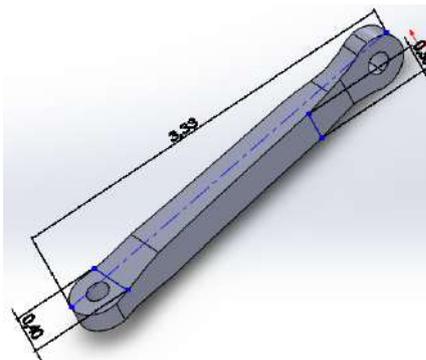


Ilustración 3-18. Sujetador entre el servo y el dedo

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Además, se crea sujetadores para los servomotores tanto frontal como posterior como se observa en la Ilustración 3-19.

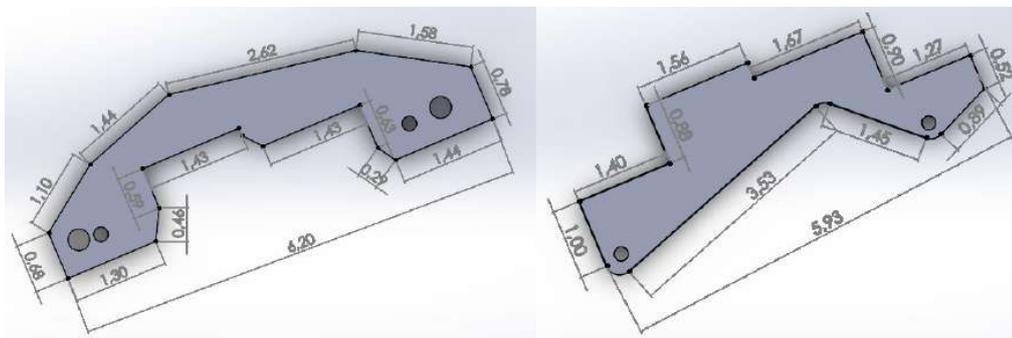


Ilustración 3-19. Sujetador para los servomotores parte frontal y posterior

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-20 se observa la base empleada para cubrir los servomotores.

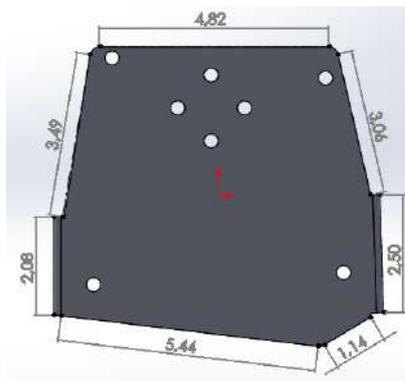


Ilustración 3-20. Base para cubrir los servomotores

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Finalmente se crea las piezas que se muestra en la Ilustración 3-21, la cuales sirven para la sujeción de la mano con el tornillo lineal.

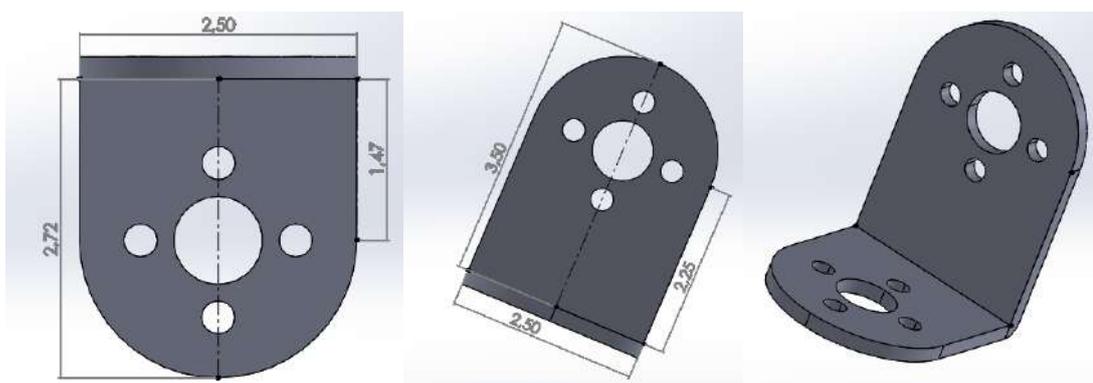


Ilustración 3-21. Piezas de sujeción

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-22 se muestra el ensamble de los dedos con base para lo que intervienen los sujetadores de las Ilustraciones 3-17 y 3-18.

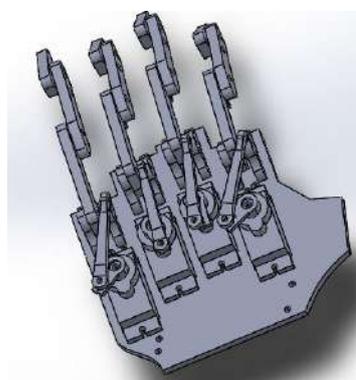


Ilustración 3-22. Ensamble de los dedos y la base

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-23 se muestra el ensamble que corresponde al sujetador frontal y posterior de los servomotores.

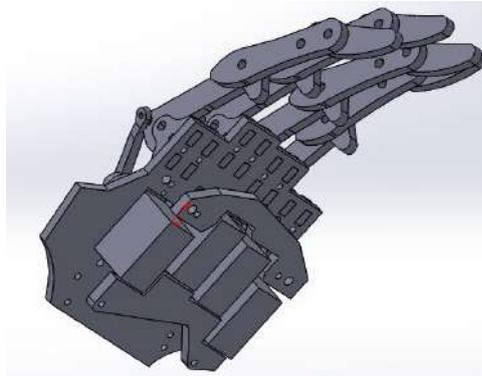


Ilustración 3-23. Ensamble de los sujetadores de los servomotores

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-24 se ve el ensamble completo de la mano izquierda.

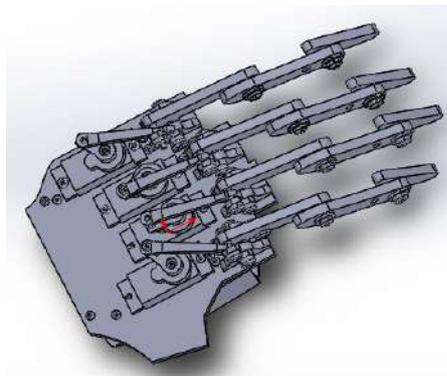


Ilustración 3-24. Ensamble completo de la mano izquierda

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.4.2 *Diseño del tornillo lineal*

En la Ilustración 3-25 se observa el diseño del modelo 3D del tornillo lineal usado en el diseño del prototipo y sus respectivas dimensiones.



Ilustración 3-25. Modelo 3D del tornillo lineal

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.5 Descripción de los elementos *hardware* del sistema

Una vez desarrollada la arquitectura general del prototipo y su diagrama de bloques se realiza el desglose de los elementos *hardware* que conforman el prototipo, a continuación, se presenta información técnica resumida de sus elementos a emplear.

3.5.1 Tornillo lineal Fuyu FSL40

Para el movimiento de las manos en el eje x, se utiliza un tornillo de bola de mesa de guía lineal Fuyu FSL40 el cual viene incorporado un motor Nema 23, el mismo que se puede visualizar en la Ilustración 3-26.



Ilustración 3-26. Tornillo de bola de mesa Fuyu FSL40

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 3-1 se puede observar las principales características técnicas que dispone el tornillo de bola de mesa de guía lineal Fuyu FSL40 junto con su motor Nema 23.

Tabla 3-1. Características técnicas del tornillo lineal Fuyu FSL40

| Especificaciones del tornillo lineal | Características |
|--------------------------------------|-----------------------|
| Precisión de posición | ± 0.01 mm |
| Carga Horizontal máxima | 25 kg |
| Carga vertical máxima | 15 kg |
| Máxima velocidad horizontal | 150 mm/s |
| Velocidad máxima montada | 500 mm/s ² |
| Aceleración nominal | 376 N |
| Empuje máximo | -15°C - +60°C |
| Ángulo de paso | 1.8° |
| Longitud del motor | 56 mm |
| Par de retención | 0.95 Nm |
| Corriente | 2 A |

Fuente: ANEXO A

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.5.2 Fuente de poder T-250W 24V 10A

Para la alimentación del prototipo se ha definido usar la fuente de poder T-250W 24V como se observa en la Ilustración 3-27, que permite la alimentación adecuada para los actuadores y sensores.



Ilustración 3-27. Fuente de alimentación T-250W 24 V 10A

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 3-2, se encuentran las principales especificaciones de la fuente de alimentación para la implementación del prototipo.

Tabla 3-2. Especificaciones de la fuente de alimentación T-250W 24 V 10 A

| Especificaciones | Características |
|----------------------------|-------------------------|
| Voltaje de entrada | 200 V – 240 V AC |
| Voltaje de salida | 24 V DC |
| Corriente de salida máxima | 10 A |
| Potencia máxima | 250 W |
| Eficiencia de salida | 83 % |
| Temperatura de operación | -10° hasta + 60° C |
| Tamaño | 215 mm x 113 mm x 50 mm |

Fuente: ANEXO B

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.5.3 Fuente de alimentación RS25-5

Para la alimentación adecuada de los ESP32 maestro y esclavo se utiliza una fuente de alimentación RS25-5, como se observa en la Ilustración 3-28.



Ilustración 3-28. Fuente de alimentación RS25-5

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 3-3, se encuentra las principales características técnicas que dispone la fuente de alimentación RS25-5.

Tabla 3-3. Especificaciones de la fuente de alimentación RS25-5

| Especificaciones | Características |
|----------------------------|-----------------------|
| Voltaje de salida | 5 V DC |
| Corriente de salida máxima | 5 A |
| Potencia máxima | 25 W |
| Eficiencia de salida | 78.5 % |
| Tamaño | 78 mm x 51 mm x 28 mm |

Fuente: ANEXO C

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.5.4 *Sensor magnético*

Permite al prototipo encontrar su posición inicial y que las manos se enceren y lleguen a su posición adecuada. En la Ilustración 3-29, se puede evidenciar el sensor a utilizar.

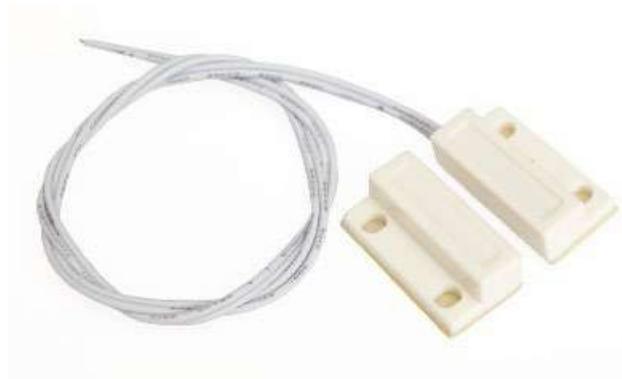


Ilustración 3-29. Sensor magnético

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 3-4, se puede observar con más detalle las principales características técnicas que ayudan a validar el proyecto.

Tabla 3-4. Especificaciones del sensor magnético

| Especificaciones | Características |
|-------------------------|------------------------|
| Voltaje máximo | 100 V |
| Corriente máxima | 0.5 A |
| Distancia de activación | 15 – 25 mm |
| Tamaño | 34 mm x 41 mm x 6.5 mm |

Fuente: ANEXO D

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.5.5 LFD-01 9G robot servo 180°

Este dispositivo permite la manipulación de cada uno de los dedos de las manos siendo así uno de los dispositivos que por su tamaño y dimensión es el más adecuado para poder mantener la estructura similar a la de una mano, en la Ilustración 3-30 se puede visualizar el dispositivo a utilizar y en la Tabla 3-5 se puede evidenciar las principales características técnicas del servo.



Ilustración 3-30. Servomotor LFD-01.

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Tabla 3-5. Especificaciones del servomotor LFD-01

| Especificaciones | Características |
|-----------------------|---------------------------------------|
| Voltaje de trabajo | 4.8 V - 6 V |
| Corriente de pérdida | 700 mA |
| Velocidad de rotación | 0.12 seg/ 60° 4.8V, 0.10 seg/ 60° 6 V |
| Torque | 1.8 kg.cm |
| Rango de rotación | 0 – 180° |
| Tamaño | 30 mm x 12 mm x 32.6 mm |

Fuente: ANEXO E

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.5.6 Controlador para motor paso a paso DM542T

Este dispositivo permite controlar los motores paso a paso para mejorar la suavidad del sistema, proporcionando un par eficiente que ayuda a la inestabilidad de rango medio. En la Ilustración 3-31, se puede visualizar el dispositivo a emplear en el prototipo.



Ilustración 3-31. Controlador DM542T

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Tabla 3-6. Especificaciones del controlador DM542T

| Especificaciones | Características |
|----------------------------|-----------------------|
| Voltaje de entrada | 20 V - 50 V |
| Corriente máxima de salida | 1 A – 4.2 A |
| Frecuencia de entrada | 0- 200 kHz |
| Velocidad | 3 RPM |
| Potencia | 8 W |
| Tamaño | 120 mm x 75 mm x 3 mm |

Fuente: ANEXO F

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.5.7 Microcontrolador ESP32

Es un microcontrolador enriquecido por su doble conectividad Wi-Fi y Bluetooth que sirve para aplicaciones de gama alta gracias a sus interfaces SPI/SDIO o I2C / UART. Este MCU tiene la facilidad de funcionar en temperaturas que varían de -40°C a +125°C con un bajo consumo de energía.



Ilustración 3-32. Microcontrolador ESP32

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 3-7 se puede observar con más detalle las principales características técnicas que posee este microcontrolador.

Tabla 3-7. Características técnicas del microcontrolador ESP32

| Especificaciones | Características |
|----------------------|-----------------------|
| Voltaje de operación | 2.2 V a 3.3 V |
| Consumo de corriente | 80 mA a 225 mA máximo |
| E/S analógicas | 11 (3.3 V) |
| E/S digitales | 1 (0.1 V) |
| Conectividad | Wi-Fi y Bluetooth |
| Memoria RAM | 520 KB |
| Memoria ROM | 448 KB |
| Memoria SRAM | 520 KB |
| Oscilador | 8 MHz -240MHz |

Fuente: ANEXO G

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.6 Esquema de conexión del prototipo

En base a los elementos de *hardware* mostrados anteriormente, se realiza los diagramas de conexión de los dispositivos que conforman el prototipo, además en la Ilustración 3-33 podemos observar una distribución por bloques de los elementos considerados para tener una idea clara de la distribución de estos.

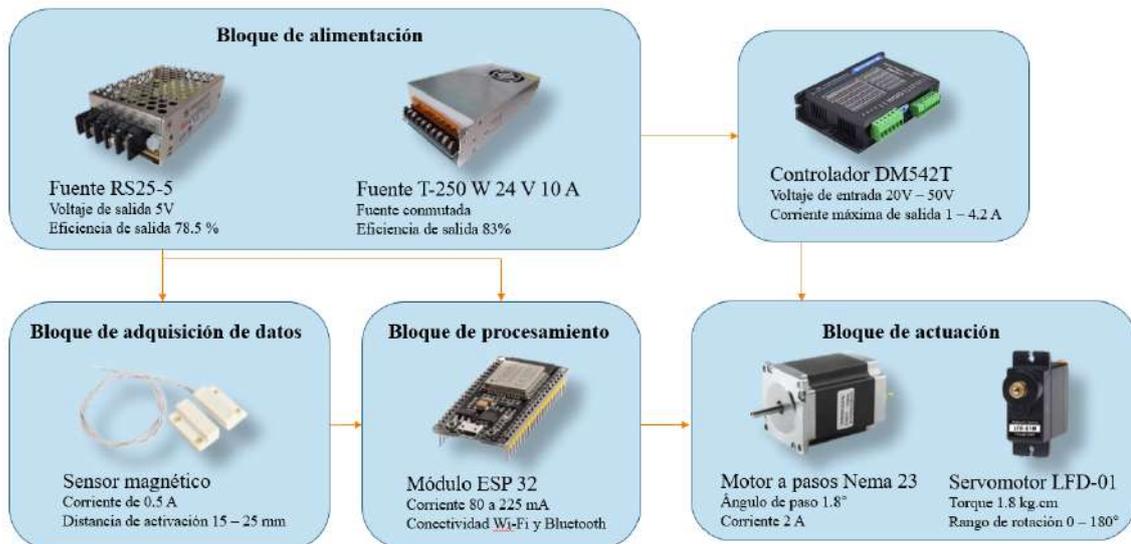


Ilustración 3-33. Distribución por bloques de los elementos hardware.

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.6.1 Esquema de conexión de la placa principal

En la Ilustración 3-34 se muestra el diagrama de conexión de la placa principal del prototipo desarrollada en EAGLE 7.6.0 el cual es un software de automatización de diseño electrónico que permite diseño de placas de circuito impreso, diagramas esquemáticos enrutamiento de PCB y contenido de biblioteca integral (Autodesk 2015). La placa principal está conformada por dos tarjetas de desarrollo ESP32 las cuales están encargadas de recibir, procesar y enviar información, estas reciben la información de los sensores para detectar la posición inicial de cada mano, una de las tarjetas de desarrollo cumple con la función de maestro y solo esta posee una conexión Wi-Fi para el envío y recepción de información, luego de procesar la información cada tarjeta de desarrollo enviara la información a cada actuador para generar su movimiento. A continuación, se detalla las conexiones realizadas para el funcionamiento del prototipo:

- La placa principal del prototipo esta alimentada con 5V, al ingreso de este voltaje se emplea un condensador electrolítico de 100 μ F conectado en paralelo a la alimentación, el cual cumple con la función filtro.
- Para la entrada de 5V de cada ESP32 de igual forma se emplea un condensador cerámico 0,1 μ F que de igual forma estarán conectados en paralelo a la alimentación para evitar picos de voltaje.
- Para la alimentación del módulo esclavo ESP32 tiene un ingreso de 5V que proviene de la fuente RS25-5 al terminal EXT_5V y los terminales GND1 y GND3 están conectados al GND de la fuente.

- El terminal IO34 del módulo esclavo ESP32 tiene como entrada la señal del sensor magnético de la mano izquierda el cual me ayuda a detectar la posición inicial de esta.
- El terminal IO32 del módulo esclavo ESP32 envía la señal de activación hacia el primer servomotor de la mano izquierda.
- El terminal IO33 del módulo esclavo ESP32 envía la señal de activación hacia el segundo servomotor de la mano izquierda.
- El terminal IO25 del módulo esclavo ESP32 envía la señal de activación hacia el tercer servomotor de la mano izquierda.
- El terminal IO26 del módulo esclavo ESP32 envía la señal de activación hacia el cuarto servomotor de la mano izquierda.
- El terminal IO27 del módulo esclavo ESP32 envía una señal al terminal DIR- del controlador DM542T de la mano izquierda para controlar la dirección de desplazamiento de esta.
- El terminal IO14 del módulo esclavo ESP32 envía una señal al terminal PUL- del controlador DM542T de la mano izquierda para controlar el conteo de pasos para el desplazamiento de esta.
- El terminal IO12 del módulo esclavo ESP32 envía una señal al terminal ENA- del controlador DM542T de la mano izquierda para activar el motor paso a paso cuando sea necesario.
- El terminal IO13 del módulo esclavo ESP32 envía una señal de activación hacia la tira led de la mano izquierda dependiendo de los servomotores que se activen en esta.
- El terminal IO17 del módulo esclavo ESP32 se conecta al terminal IO16 del módulo maestro ESP32 para tener comunicación serial.
- El terminal IO16 del módulo esclavo ESP32 se conecta al terminal IO17 del módulo maestro ESP32 para tener comunicación serial.
- Para la alimentación del módulo maestro ESP32 tiene un ingreso de 5V que proviene de la fuente RS25-5 al terminal EXT_5V y los terminales GND1 y GND3 están conectados al GND de la fuente.
- El terminal IO34 del módulo maestro ESP32 tiene como entrada la señal del sensor magnético de la mano derecha el cual me ayuda a detectar la posición inicial de esta.
- El terminal IO32 del módulo maestro ESP32 envía la señal de activación hacia el primer servomotor de la mano derecha.
- El terminal IO33 del módulo maestro ESP32 envía la señal de activación hacia el segundo servomotor de la mano derecha.
- El terminal IO25 del módulo maestro ESP32 envía la señal de activación hacia el tercer servomotor de la mano derecha.
- El terminal IO26 del módulo maestro ESP32 envía la señal de activación hacia el cuarto servomotor de la mano derecha.

- El terminal IO27 del módulo maestro ESP32 envía una señal al terminal DIR- del controlador DM542T de la mano derecha para controlar la dirección de desplazamiento de esta.
- El terminal IO14 del módulo maestro ESP32 envía una señal al terminal PUL- del controlador DM542T de la mano derecha para controlar el conteo de pasos para el desplazamiento de esta.
- El terminal IO12 del módulo maestro ESP32 envía una señal al terminal ENA- del controlador DM542T de la mano derecha para activar el motor paso a paso cuando sea necesario.
- El terminal IO13 del módulo maestro ESP32 envía una señal de activación hacia la tira led de la mano derecha dependiendo de los servomotores que se activen en esta.
- El terminal IO17 del módulo maestro ESP32 se conecta al terminal IO16 del módulo esclavo ESP32 para tener comunicación serial.
- El terminal IO16 del módulo maestro ESP32 se conecta al terminal IO17 del módulo esclavo ESP32 para tener comunicación serial.
- El terminal IO2 del módulo maestro ESP32 envía una señal hacia la base de un transistor 2N3904 y el colector de este está conectado hacia un buzzer el cual emite un pidió indicando la activación de las manos.

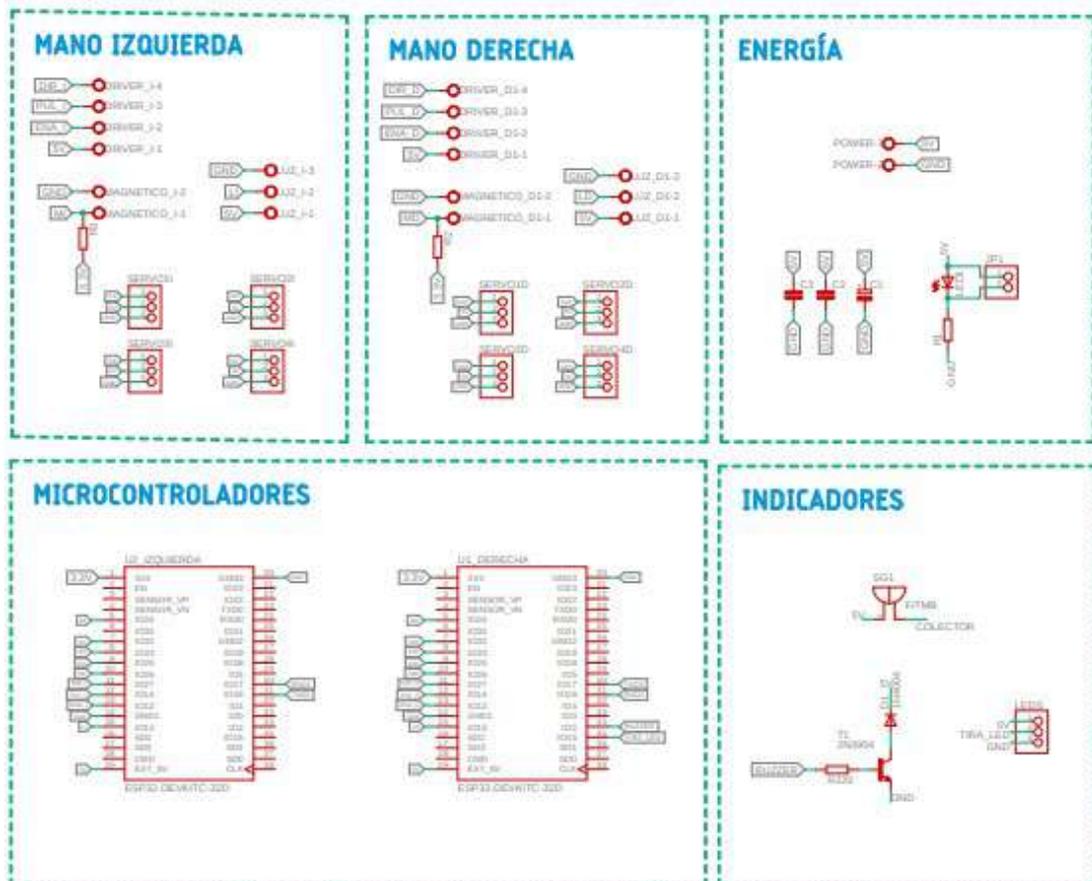


Ilustración 3-34. Esquema de conexión de la placa principal

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Una vez realizadas las respectivas conexiones de cada uno de los componentes se procede a elaborar la placa en Fusion 360, el cual es un software de desarrollo de productos que integra diseño, simulación, electrónica, fabricación, etc. (Autodesk 2015). En la Ilustración 3-35 se puede observar el diseño de la placa.

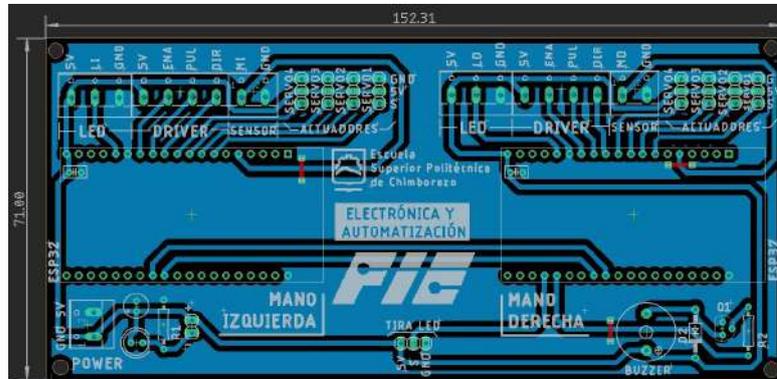


Ilustración 3-35. Placa PCB

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-36 se evidencia el modelo en 3D de la placa PCB, donde se observan los componentes a usar y la ubicación exacta de cada uno.

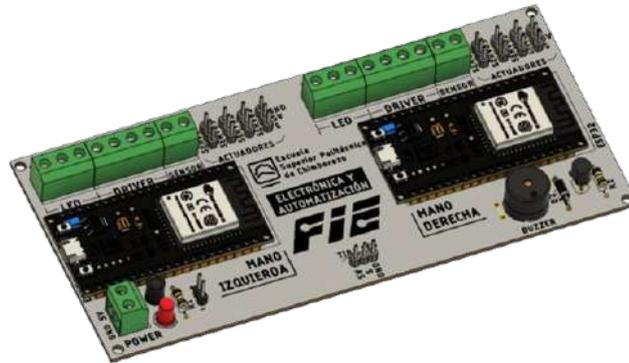


Ilustración 3-36. Modelo en 3D de la placa

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-37 se puede observar la placa PCB implementada físicamente con sus respectivos elementos.



Ilustración 3-37. Placa PCB implementada físicamente

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.6.2 *Conexión del controlador DM542T*

Una vez realizadas las conexiones de la placa principal se procede a conectar los terminales restantes del controlador DM542T empleado para cada mano. A continuación, se detalla las conexiones realizadas:

- Los terminales PUL+, DIR+ y ENA+ del controlador de la mano izquierda están conectados a los 5V que nos entrega la fuente RS25-5.
- Los terminales GND y +Vdc del controlador de la mano izquierda están conectados a la fuente T-250W 24 V 10A, dado que esta fuente nos entrega un voltaje de 24V.
- Los terminales A+, A-, B+ y B- del controlador de la mano izquierda están conectados a los bobinados del motor paso a paso de la misma mano.
- Los terminales PUL+, DIR+ y ENA+ del controlador de la mano derecha están conectados a los 5V que nos entrega la fuente RS25-5.
- Los terminales GND y +Vdc del controlador de la mano derecha están conectados a la fuente T-250W 24 V 10A, dado que esta fuente nos entrega un voltaje de 24V.
- Los terminales A+, A-, B+ y B- del controlador de la mano derecha están conectados a los bobinados del motor paso a paso de la misma mano.

3.6.3 *Diseño de la estructura del prototipo*

Una vez realizadas todas las conexiones pertinentes se procede a realizar el diseño de la estructura que contiene al prototipo. En la Tabla 3-8 se encuentra detalladas las partes de la estructura junto con sus medidas y además en la Ilustración 3-38 y 3-39 se puede observar el diseño de cada parte de forma más detalla junto con las dimensiones en centímetros.

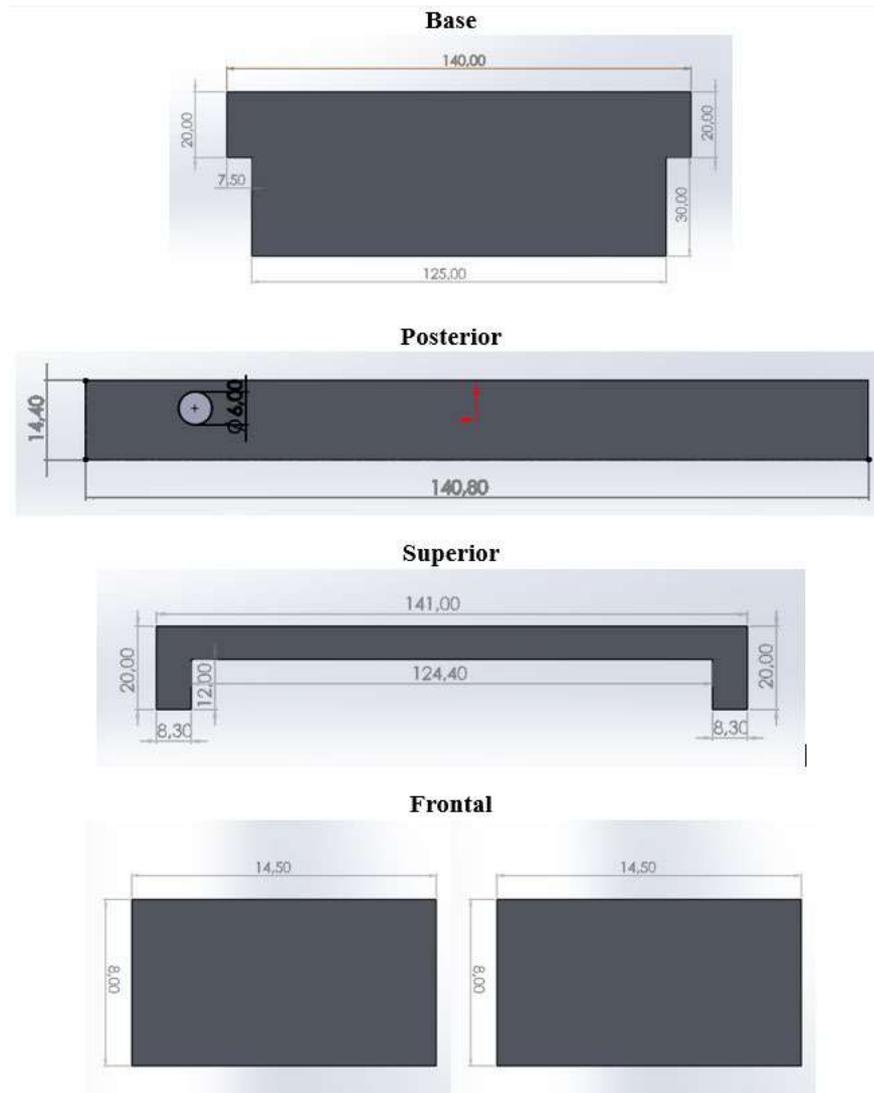


Ilustración 3-38. Dimensiones de la estructura del prototipo primera parte
Realizado por: (Aguirre Marco y Meneses Jean 2023)

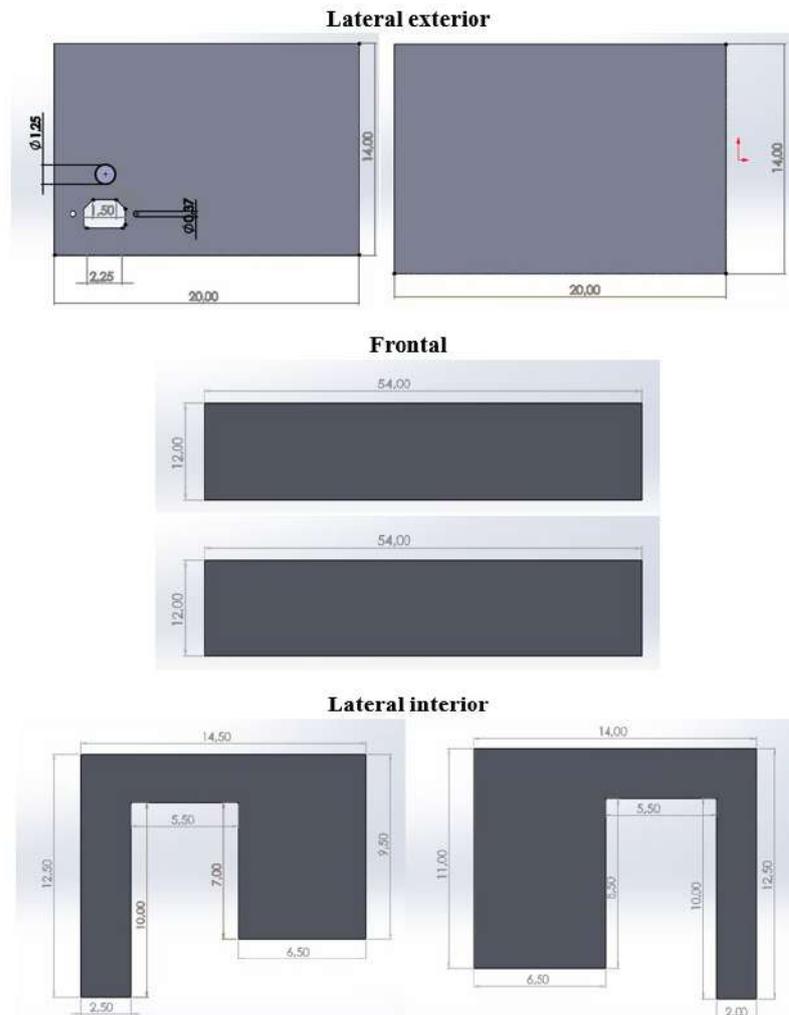


Ilustración 3-39. Dimensiones de la estructura del prototipo segunda parte

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-40 se observa el diseño 3D de la estructura, donde se encuentran contenidos todos los elementos usados para implementar el prototipo.

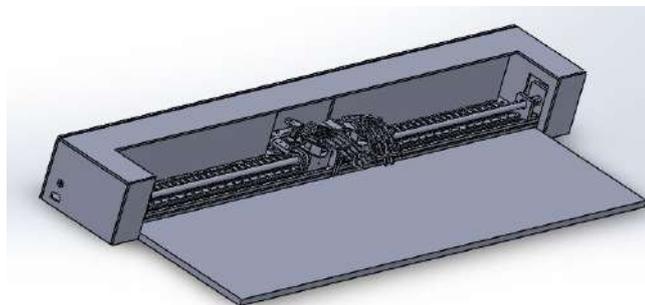


Ilustración 3-40. Diseño en 3D de la estructura

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Tabla 3-8. Medidas que constituye la estructura del prototipo

| Partes de la estructura | Medidas | |
|-----------------------------------|-----------------|----------|
| Base | Largo posterior | 140 cm |
| | Largo frontal | 125 cm |
| | Ancho posterior | 20 cm |
| | Ancho frontal | 30 cm |
| Posterior | Largo | 140 cm |
| | Ancho | 14 cm |
| Superior | Largo posterior | 141 cm |
| | Largo frontal | 124.4 cm |
| | Largo extremos | 8.3 cm |
| | Ancho | 20 cm |
| Frontal | Largo | 14.5 cm |
| | Ancho | 8 cm |
| Lateral exterior | Largo | 20 cm |
| | Ancho | 14 cm |
| Frontal | Largo | 54 cm |
| | Ancho | 12 cm |
| Lateral interior izquierdo | Largo | 14.5 cm |
| | Ancho | 12.5 cm |
| Lateral interior derecho | Largo | 14 cm |
| | Ancho | 12.5 cm |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.7 Programación del prototipo

A continuación, se detalla el diagrama de flujo correspondiente a la programación de las tarjetas de desarrollo ESP32 y se describe su funcionamiento, en el Anexo I y en el Anexo J se puede observar la programación de cada tarjeta la cual fue realizada en Arduino IDE 2.1.1 el cual es un software de código abierto que admiten los lenguajes de programación C y C++ además nos permite escribir, depurar, editar y grabar programas (Arduino 2023). A continuación, se detallan las librerías empleadas para la programación de las tarjetas de desarrollo.

Librerías empleadas en el módulo maestro ESP32

En la Tabla 3-9, se describe las librerías utilizadas para la programación del módulo maestro ESP32 el cual permite la comunicación Wi-Fi y serial entre el módulo esclavo ESP32 para la adecuada interpretación de las canciones.

Tabla 3-9. Librerías usadas en la programación del módulo maestro

| Librerías | Descripción |
|-----------------------------|--|
| WiFiClient.h | Crea un cliente que puede conectarse a una dirección IP y a un puerto definido. |
| HTTPClient.h | Sirve para realizar solicitudes a un servidor web. |
| WiFiManager.h | Permite administrar la configuración Wi-Fi con un puerto de configuración web para las placas de la familia ESPx. |
| WiFi.h | Permite establecer la configuración y la comunicación a internet. |
| FirestoreESP32.h | Biblioteca de base de datos en tiempo real, segura, rápida y confiable para leer, almacenar, actualizar, eliminar, escuchar, hacer una copia de seguridad y restaurar datos. |
| ESP8266WiFi.h | Biblioteca para internet de las cosas se encarga de enviar y controlar datos |
| Addons/TokenHelper.h | Proporciona la información de proceso de generación del token |
| Separador.h | Separa los arreglos de las canciones a interpretar. |
| Servo.h | Permite el control de los servomotores. |
| Map | Mapea un número de un rango a otro, también maneja los números negativos. |
| Adafruit_NeoPixel.h | Controla los pixeles y tiras LED de un solo cable. |
| avr/power.h | permiten reducir el consumo de energía al deshabilitar o habilitar varios periféricos a bordo según sea necesario. |

Fuente: (Arduino 2023)

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Para la programación del módulo maestro ESP32 se considera los siguientes pasos.

Inicialización

- Incluye las librerías descritas anteriormente en la Tabla 3-9.
- Se declara los terminales de salida para la comunicación, donde TX y RX son los terminales de transmisión y recepción del módulo maestro, correspondiente a los pines 17 y 16 del ESP32.
- Se declara los terminales para la conexión de la mano derecha con los servomotores conectados a los pines 32, 33, 25, 26 del ESP32.
- Se declara los terminales para la conexión del controlador para el motor paso a paso para la mano derecha conectados a los pines 18, 14, 27 del ESP32.
- Declaración de las variables.
- Creamos las notas musicales
- Se inicia la comunicación con la base de datos
- Se habilita la comunicación entre los módulos maestro y esclavo
- Se establece la comunicación entre la aplicación

Bucle de posición

- Las manos se encuentran en la posición inicial

- Establece comunicación Wi-Fi.
- Se establece comunicación con la aplicación y la base de datos.

Bucle de Selección de modo

- La aplicación se comunica con el módulo maestro para escoger la opción manual o piano.
- Al seleccionar el modo manual se puede escoger una nota musical en específico a interpretar.
- Comunicación entre los módulos maestro y esclavo.
- Activación de los motores paso a paso para el desplazamiento en el eje x, además se activan los servomotores para mover cada dedo acorde a la tecla a interpretar.
- Al finalizar la nota musical se tiene un bucle para escoger otra o regresar a la selección de modo.
- Al seleccionar el modo piano se puede escoger las canciones a interpretar
- Comunicación entre los módulos maestro y esclavo.
- Activación de los motores paso a paso para el desplazamiento en el eje x, además se activan los servomotores para mover cada dedo acorde a la canción a interpretar.
- Al finalizar la canción se tiene un bucle para verificar la posición inicial de las manos y de igual manera se tiene otro bucle para escoger otra canción a interpretar o regresar a la selección de modo.

Librerías empleadas en el módulo esclavo ESP32

En la Tabla 3-10, se describe las librerías usadas para la programación del módulo esclavo ESP32 la cual debe poseer comunicación serial con el módulo maestro ESP32 para la interpretación de las canciones.

Tabla 3-10. Librerías usadas en la programación del módulo esclavo ESP32

| Librerías | Descripción |
|----------------------------|--|
| Servo.h | Permite el control de los servomotores. |
| map | Mapea un número de un rango a otro, también maneja los números negativos. |
| Adafruit_NeoPixel.h | Controla los pixeles y tiras LED de un solo cable. |
| avr/power.h | Permiten reducir el consumo de energía al deshabilitar o habilitar varios periféricos a bordo según sea necesario. |

Fuente: (Arduino 2023)

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Para la programación del módulo esclavo ESP32 se considera los siguientes pasos.

Inicialización

- Incluye las librerías descritas en la Tabla 3-10.

- Se declara los terminales de salida para la comunicación, donde TX y RX son los terminales de transmisión y recepción con el módulo esclavo, conectados a los pines 17 y 16 del ESP32.
- Se declara los terminales para la conexión de la mano izquierda con los servomotores conectados a los pines 32, 33, 25, 26 del ESP32.
- Se declara los terminales para la conexión del controlador para el motor a pasos para la mano izquierda conectados a los pines 18, 14, 27 del ESP32.
- Declaración de las variables
- Creamos las notas musicales
- Se comunica con el módulo maestro para recibir la información

Bucle de posición

- Las manos se encuentran en la posición inicial
- Establece comunicación Wi-Fi.
- Se establece comunicación con la aplicación y la base de datos.

Bucle de Selección de modo

- La aplicación se comunica con el módulo maestro para escoger la opción manual o piano.
- Al seleccionar el modo manual se puede escoger una nota musical en específico a interpretar.
- Comunicación entre los módulos maestro y esclavo.
- Activación de los motores paso a paso para el desplazamiento en el eje x, además se activan los servomotores para mover cada dedo acorde a la tecla a interpretar.
- Al finalizar la nota musical se tiene un bucle para escoger otra nota musical o regresar a la selección de modo.
- Al seleccionar el modo piano se puede escoger las canciones a interpretar
- Comunicación entre los módulos maestro y esclavo.
- Activación de los motores paso a paso para el desplazamiento en el eje x, además se activan los servomotores para mover cada dedo acorde a la canción a interpretar.
- Al finalizar la canción se tiene un bucle para verificar la posición inicial de las manos y de igual manera se tiene otro bucle para escoger otra canción a interpretar o regresar a la selección de modo.

En la Ilustración 3-41, se puede evidenciar el proceso para la programación del módulo maestro ESP32 y para el módulo esclavo ESP32.

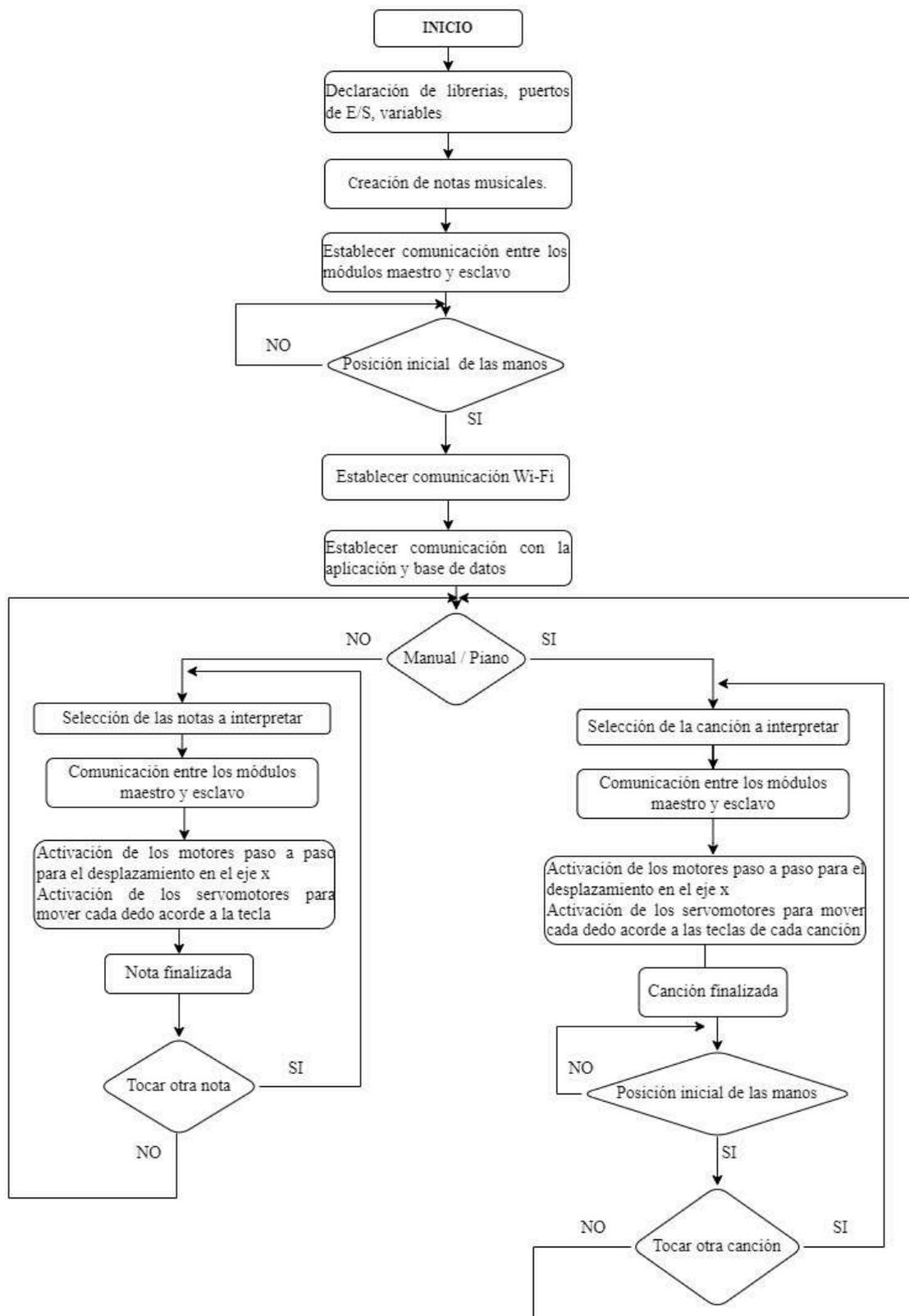


Ilustración 3-41. Diagrama de flujo del prototipo

Realizado por: (Aguirre Marco y Meneses Jean 2023)

A continuación, en la Tabla 3-11 se puede observar el número de pasos considerados para la creación de cada nota.

Tabla 3-11. Distribución de número de pasos para la creación de las notas musicales

| Notación Anglosajón | Número de pasos |
|---------------------|-----------------|
| A4 | 0 |
| B4 | 465 |
| C5 | 930 |
| D5 | 1395 |
| E5 | 1860 |
| F5 | 2325 |
| G5 | 2790 |
| A5 | 3255 |
| B5 | 3720 |
| C6 | 4185 |
| D6 | 4650 |
| E6 | 5115 |
| F6 | 5580 |
| G6 | 6045 |
| A6 | 6510 |
| B6 | 6975 |
| C7 | 7440 |
| D7 | 7905 |
| E7 | 8370 |
| F7 | 8835 |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.8 Base de datos en Firebase

Para el presente trabajo se crea una base de datos en donde estarán almacenadas las canciones a interpretar, para esto se usa Firebase la cual es una base de datos gratuita no relacional alojada en la nube, donde la información es almacenada en formato JSON y permite una sincronización en tiempo real con el usuario (Firebase 2023).

Para crear la base de datos en Firebase se realiza los siguientes pasos:

- Entrar a la página oficial Firebase “<https://firebase.google.com/?hl=es-419>”
- Se selecciona ir a consola
- Agregar un nuevo proyecto
- Asignar un nombre para el proyecto “Piano”
- Habilitar Google Analytics
- Configurar Google Analytics
- Finalmente se crea una base de datos en tiempo real.

Una vez creada la base de datos en tiempo real se deben asignar los nombres de las variables a utilizar para la asignación de las canciones a interpretar. Para vincular la base de datos en tiempo real junto con la programación del ESP32 es necesario asignar el API KEY y DATABASE URL, para obtener se realiza los siguientes pasos:

- Dirigirse descripción general, configuración del proyecto, cuentas de servicios, secretos de la base de datos, finalmente en mostrar para obtener el Api Key.
- Copiar el URL y eliminar ‘https://’ y ‘/’.

Realtime Database

Es una base de datos en tiempo real que permite almacenar y sincronizar datos en la nube, los mismos se actualizan en cuestión de milisegundos. En la Ilustración 3-42 se puede visualizar la creación de la base de datos. La misma tendrá comunicación directa con el módulo maestro ESP32, este recibirá la información de la canción a interpretar y una vez culminada enviará que la canción finalizó hacia la base de datos.



Ilustración 3-42. Información que contiene la base de datos

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Las canciones almacenadas en la base de datos envían información hacia el módulo maestro ESP32 para ser interpretadas, los datos enviados son de tipo String los mismos que se pueden observar en la Ilustración 3-43.

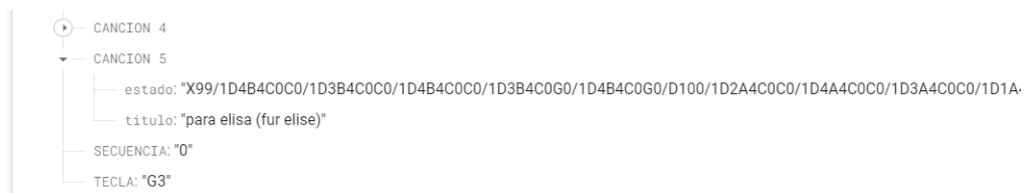


Ilustración 3-43. Envío de la información en tipo String

Realizado por: (Aguirre Marco y Meneses Jean 2023)

3.9 Interfaz de intervención con el usuario

El prototipo cuenta con una aplicación para dispositivos móviles Android desarrollada en Android Studio 2022.2.1 el cual es un software con un entorno de desarrollo integrado (IDE) que se usa en el desarrollo de apps para Android, basado en un potente editor de código y herramientas para desarrolladores (Android Studio 2020). La aplicación tiene como objetivo permitir al usuario escoger entre modo manual y piano; el modo manual permite enviar las manos a una nota en específico, mientras que el modo piano permite seleccionar una de las canciones almacenadas para ser interpretadas por el prototipo. El desarrollo de la aplicación móvil se muestra en el Anexo K.

La aplicación móvil cuenta con una interfaz constituida de un conjunto de ventanas, en la Ilustración 3-44 se observa la pantalla principal la cual permite escoger entre el modo manual y el modo piano.



Ilustración 3-44. Pantalla principal de la aplicación móvil

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-45 se observa la interfaz manual que permite escoger una nota específica para desplazar las manos hacia la tecla designada.



Ilustración 3-45. Pantalla del modo manual de la aplicación móvil

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-46 se observa la interfaz del modo piano que permite escoger una de las canciones almacenadas para interpretarse.

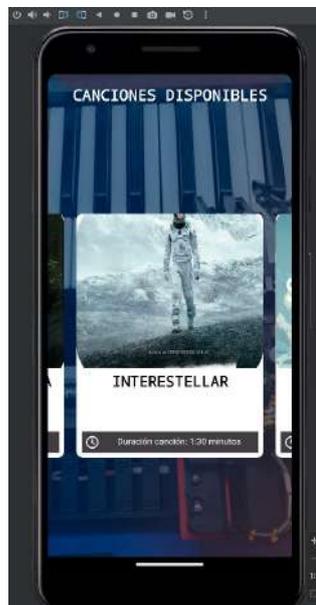


Ilustración 3-46. Pantalla del modo piano de la aplicación móvil

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Ilustración 3-47 se observa la interfaz que se genera una vez que escogemos una de las canciones almacenadas, esta interfaz depende de la canción seleccionada.



Ilustración 3-47. Pantalla del modo piano de la aplicación móvil

Realizado por: (Aguirre Marco y Meneses Jean 2023)

CAPITULO IV

4. PRUEBAS Y RESULTADOS

En el presente capítulo se evalúa el desempeño del prototipo, esta evaluación se realiza tanto del *software* como de *hardware* mediante las respectivas pruebas de validación de sensores, estabilidad, precisión de desplazamiento al momento de tocar una tecla, comunicación, consumo total de corriente y funcionamiento general del prototipo.

En la Ilustración 4-1 se observa el prototipo a evaluar, el cual está constituido por una placa PCB de control, sus manos que cumplen la función de tocar el piano y su aplicación móvil.



Ilustración 4-1. Prototipo de manos robóticas aplicadas al área del entretenimiento

Realizado por: (Aguirre Marco y Meneses Jean 2023)

4.1 Consideraciones generales

Para evaluar el prototipo se realizó pruebas que permitan comprobar la validez y la confiabilidad, según (Villasis-Keever et al. 2018) indica que al realizar esta serie de pruebas se puede considerar que los resultados obtenidos son válidos con un bajo grado de error.

Para la validación de exactitud de sensores se indica que tan cerca está el valor medido con el valor real o patrón, para encontrar este valor se ocupó la Ecuación 4-1, en donde se encuentra el error absoluto.

Ecuación 4-1. Error absoluto

$$\Delta X = |X_r - X_m|$$

Donde:

X_r : Valor real.

X_m : Valor medido.

Para la exactitud hallada mediante error absoluto según (Bell 2003), es necesario una cantidad de 30 muestras, para evaluar un correcto funcionamiento.

El error relativo es fundamental para la validación del prototipo, ya que es un indicador estadístico que permite determinar la calidad del resultado de la muestra, para encontrar dicho valor se emplea la Ecuación 4-2.

Ecuación 4-2. Error relativo.

$$e_r = \frac{\Delta X}{X_r} * 100$$

Donde:

ΔX : Error absoluto.

X_r : Valor real.

Para la validación de exactitud y confiabilidad empleando el error relativo según (Bell 2003), cuando se obtiene un error menor al 1% el desempeño es excelente, si el error es menor al 2% es bueno, si el error es menor al 10% es aceptable y si es mayor al 10% es poco confiable.

Para determinar la precisión del prototipo se realiza mediante el cálculo del coeficiente de variación definido en la Ecuación 4-3 y la desviación estándar definida en la Ecuación 4-4

Ecuación 4-3. Coeficiente de variación.

$$CV = \left(\frac{\sigma}{\bar{X}} \right) * 100\%$$

Donde:

σ : Desviación estándar.

\bar{X} : Media de las muestras.

Ecuación 4-4. Desviación estándar.

$$\sigma = \sqrt{\frac{\sum_i^N (X_i - \bar{X})^2}{N - 1}}$$

Donde:

X_i : Número i de la variable X.

\bar{X} : Media de la variable X.

N: Número de observaciones.

Para la prueba de estabilidad por coeficiente de variación es necesario la toma de una cantidad de 10 muestras según (Portuondo Paisa, Portuondo Moret 2010). Los resultados obtenidos menores al 5% son óptimos, si son menores al 10% son aceptables y si son mayores al 10% son poco confiables (Romero Ramos 2016).

En la Tabla 4-1 y en la Tabla 4-2 se puede observar los parámetros considerados para las pruebas y resultados considerando su número de muestras y criterios de valoración de resultados según los autores.

Tabla 4-1. Parámetros para la toma de muestras

| Autores | Número de muestras |
|-----------------------------------|---------------------------|
| David Bell | 30 |
| Portuondo Paisa y Portuondo Moret | 10 |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Tabla 4-2. Criterios de valoración para los resultados

| Autores | Criterio de valoración |
|----------------|--|
| David Bell | Error Relativo < 1%, desempeño excelente Error Relativo < 2%, es bueno Error Relativo < 10%, es aceptable Error Relativo > 10%, es poco confiable |
| Romero Ramos | CV < 5%, son óptimos CV < 10%, son aceptables CV > 10%, son poco confiables |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

4.2 Validación de sensores

Se tiene como objetivo verificar la exactitud de los sensores magnéticos ubicados en la estructura del prototipo por medio del cálculo del error relativo, según (Bell 2003) se debe considerar una toma de 30 muestras para una correcta evaluación, para la toma de datos se empleó una regla de la marca “Estilo” la cual presenta un error de 1%.

4.2.1 Validación del sensor magnético de la mano derecha

Se verifica que el sensor magnético de la mano derecha esté ubicado en la posición inicial, por lo cual se consideró que el sensor magnético se activará a una distancia de 1.4 cm de su contactor para evitar que la estructura sufra colisiones como se observa en la Ilustración 4-2.



Ilustración 4-2. Evidencia de toma de muestras del sensor magnético de la mano derecha

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 4-3 se muestra los cálculos del valor absoluto y relativo del sensor magnético de la mano derecha.

Tabla 4-3. Error absoluto y relativo del sensor magnético de la mano derecha

| Número de muestras | Medida patrón [cm] | Medición de la distancia de activación del sensor magnético [cm] | Error absoluto [cm] | Error relativo (%) |
|--------------------|--------------------|--|---------------------|--------------------|
| 1 | 1.4 | 1.5 | 0.1 | 7.14 |
| 2 | 1.4 | 1.4 | 0 | 0 |
| 3 | 1.4 | 1.3 | 0.1 | 7.14 |
| 4 | 1.4 | 1.4 | 0 | 0 |
| 5 | 1.4 | 1.4 | 0 | 0 |
| 6 | 1.4 | 1.4 | 0 | 0 |
| 7 | 1.4 | 1.4 | 0 | 0 |
| 8 | 1.4 | 1.4 | 0 | 0 |
| 9 | 1.4 | 1.4 | 0 | 0 |
| 10 | 1.4 | 1.4 | 0 | 0 |
| 11 | 1.4 | 1.4 | 0 | 0 |
| 12 | 1.4 | 1.4 | 0 | 0 |
| 13 | 1.4 | 1.4 | 0 | 0 |
| 14 | 1.4 | 1.4 | 0 | 0 |
| 15 | 1.4 | 1.4 | 0 | 0 |
| 16 | 1.4 | 1.4 | 0 | 0 |
| 17 | 1.4 | 1.4 | 0 | 0 |
| 18 | 1.4 | 1.3 | 0.1 | 7.14 |
| 19 | 1.4 | 1.4 | 0 | 0 |
| 20 | 1.4 | 1.3 | 0.1 | 7.14 |

| Número de muestras | Medida patrón [cm] | Medición de la distancia de activación del sensor magnético [cm] | Error absoluto [cm] | Error relativo (%) |
|--------------------|--------------------|--|---------------------|--------------------|
| 21 | 1.4 | 1.4 | 0 | 0 |
| 22 | 1.4 | 1.4 | 0 | 0 |
| 23 | 1.4 | 1.4 | 0 | 0 |
| 24 | 1.4 | 1.4 | 0 | 0 |
| 25 | 1.4 | 1.4 | 0 | 0 |
| 26 | 1.4 | 1.4 | 0 | 0 |
| 27 | 1.4 | 1.3 | 0.1 | 7.14 |
| 28 | 1.4 | 1.4 | 0 | 0 |
| 29 | 1.4 | 1.3 | 0.1 | 7.14 |
| 30 | 1.4 | 1.4 | 0 | 0 |
| Promedio | | | ±0.02 | 1.43 |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En base a la Tabla 4-3, se determina que el error relativo del sensor magnético de la mano derecha es de 1.43%, por lo cual según el criterio de (Bell 2003) este porcentaje es menor al 2%, determinando que el sensor posee una exactitud buena.

4.2.2 Validación del sensor magnético de la mano izquierda

Se verifica que el sensor magnético de la mano izquierda este ubicada en la posición inicial, por lo cual se consideró que el sensor magnético se activará a una distancia de 1.5 cm de su contactor para evitar que la estructura sufra colisiones como se muestra en la Ilustración 4-3.

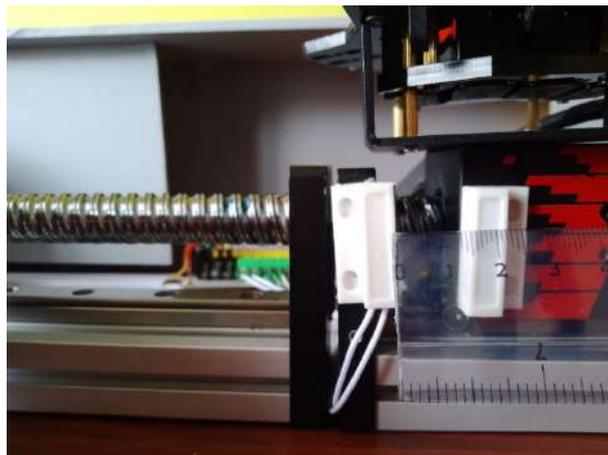


Ilustración 4-3. Evidencia de toma de muestras del sensor magnético de la mano izquierda

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 4-4 se muestra los cálculos del valor absoluto y relativo del sensor magnético de la mano izquierda.

Tabla 4-4. Error absoluto y relativo del sensor magnético de la mano izquierda

| Número de muestras | Medida patrón [cm] | Medición de la distancia de activación del sensor magnético [cm] | Error absoluto [cm] | Error relativo (%) |
|--------------------|--------------------|--|---------------------|--------------------|
| 1 | 1.5 | 1.5 | 0 | 0 |
| 2 | 1.5 | 1.5 | 0 | 0 |
| 3 | 1.5 | 1.5 | 0 | 0 |
| 4 | 1.5 | 1.4 | 0.1 | 6.67 |
| 5 | 1.5 | 1.4 | 0.1 | 6.67 |
| 6 | 1.5 | 1.5 | 0 | 0 |
| 7 | 1.5 | 1.4 | 0.1 | 6.67 |
| 8 | 1.5 | 1.4 | 0.1 | 6.67 |
| 9 | 1.5 | 1.4 | 0.1 | 6.67 |
| 10 | 1.5 | 1.5 | 0 | 0 |
| 11 | 1.5 | 1.5 | 0 | 0 |
| 12 | 1.5 | 1.5 | 0 | 0 |
| 13 | 1.5 | 1.5 | 0 | 0 |
| 14 | 1.5 | 1.5 | 0 | 0 |
| 15 | 1.5 | 1.4 | 0.1 | 6.67 |
| 16 | 1.5 | 1.5 | 0 | 0 |
| 17 | 1.5 | 1.5 | 0 | 0 |
| 18 | 1.5 | 1.5 | 0 | 0 |
| 19 | 1.5 | 1.5 | 0 | 0 |
| 20 | 1.5 | 1.5 | 0 | 0 |
| 21 | 1.5 | 1.4 | 0.1 | 6.67 |
| 22 | 1.5 | 1.5 | 0 | 0 |
| 23 | 1.5 | 1.5 | 0 | 0 |
| 24 | 1.5 | 1.5 | 0 | 0 |
| 25 | 1.5 | 1.5 | 0 | 0 |
| 26 | 1.5 | 1.5 | 0 | 0 |
| 27 | 1.5 | 1.5 | 0 | 0 |
| 28 | 1.5 | 1.5 | 0 | 0 |
| 29 | 1.5 | 1.5 | 0 | 0 |
| 30 | 1.5 | 1.5 | 0 | 0 |
| Promedio | | | ±0.02 | 1.56 |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En base a la Tabla 4-4, se determina que el error relativo del sensor magnético de la mano izquierda es de 1.56%, por lo cual según el criterio de (Bell 2003) este porcentaje es menor al 2%, determinando que el sensor posee una exactitud buena.

4.3 Estabilidad de sensores

Esta prueba tiene objetivo verificar la precisión de los sensores magnéticos implementados en el prototipo mediante el análisis del coeficiente de variación, según (Portuondo Paisa, Portuondo Moret 2010) se debe considerar una toma de 10 muestras, para la toma de datos se empleó una regla de la marca “Estilo” la cual presenta un error de 1%.

4.3.1 Estabilidad del sensor magnético de la mano derecha

Se determina el comportamiento del sensor magnético de la mano derecha durante un periodo de tiempo con un intervalo de diez segundos entre muestras iniciando a las 15:00:00, manteniendo las mismas condiciones de funcionamiento como se observa en la Ilustración 4-4.



Ilustración 4-4. Evidencia de toma de datos para la estabilidad del sensor magnético de la mano derecha

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 4-5 se registra la distancia del sensor magnético de la mano derecha al contactor.

Tabla 4-5. Prueba de estabilidad del sensor magnético de la mano derecha

| Número de muestras | Hora | Medición de la distancia de activación del sensor magnético (cm) |
|--------------------|----------|--|
| 1 | 15:00:00 | 1.4 |
| 2 | 15:00:10 | 1.4 |
| 3 | 15:00:20 | 1.3 |
| 4 | 15:00:30 | 1.3 |

| Número de muestras | Hora | Medición de la distancia de activación del sensor magnético (cm) |
|----------------------------|----------|--|
| 5 | 15:00:40 | 1.4 |
| 6 | 15:00:50 | 1.4 |
| 7 | 15:01:00 | 1.4 |
| 8 | 15:01:10 | 1.4 |
| 9 | 15:01:20 | 1.4 |
| 10 | 15:01:30 | 1.4 |
| Media | | 1.38 |
| Desviación estándar | | 0.04 |
| CV | | 3.06 % |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En base a los cálculos de la Tabla 4-5, se tiene que el coeficiente de variación del sensor magnético de la mano derecha es de 3.06%, determinando que la estabilidad del sensor mediante el criterio de (Romero Ramos 2016) es menor al 5%, concluyendo que la precisión del sensor es óptima.

4.3.2 Estabilidad del sensor magnético de la mano izquierda

Se determina el comportamiento del sensor magnético de la mano izquierda durante un periodo de tiempo con un intervalo de diez segundos entre muestras iniciando a las 15:10:00, manteniendo las mismas condiciones de funcionamiento como se observa en la Ilustración 4-5.



Ilustración 4-5. Evidencia de la toma de datos para la estabilidad del sensor magnético de la mano izquierda

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 4-6 se registra la distancia del sensor magnético de la mano izquierda al contactor.

Tabla 4-6. Prueba de estabilidad del sensor magnético de la mano izquierda

| Número de muestras | Hora | Medición de la distancia de activación del sensor magnético (cm) |
|----------------------------|----------|--|
| 1 | 15:10:00 | 1.4 |
| 2 | 15:10:10 | 1.5 |
| 3 | 15:10:20 | 1.5 |
| 4 | 15:10:30 | 1.5 |
| 5 | 15:10:40 | 1.5 |
| 6 | 15:10:50 | 1.4 |
| 7 | 15:11:00 | 1.4 |
| 8 | 15:11:10 | 1.5 |
| 9 | 15:11:20 | 1.5 |
| 10 | 15:11:30 | 1.5 |
| Media | | 1.47 |
| Desviación estándar | | 0.05 |
| CV | | 3.28 % |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En base a los cálculos de la Tabla 4-6, se tiene que el coeficiente de variación del sensor magnético de la mano izquierda es de 3.28%, determinando que la estabilidad del sensor mediante el criterio de (Romero Ramos 2016) es menor al 5%, concluyendo que la precisión del sensor es óptima.

4.4 Estabilidad de los tiempos de reproducción de las canciones

Para la prueba de estabilidad de los tiempos de reproducción de las canciones que interpreta el prototipo se consideró la primera y última canción que se puede visualizar en la aplicación móvil, además se consideró una cantidad de 30 muestras con la finalidad de obtener una mayor precisión en los resultados.

4.4.1 Estabilidad de reproducción de la primera canción

Se determina el tiempo de reproducción de la primera canción durante un periodo de tiempo, manteniendo las mismas condiciones de funcionamiento. La toma de muestras se inició a las 10:00:00, con un intervalo de treinta y cinco segundos entre muestras por medio de la función de cronometro de un dispositivo móvil como se observa en la Ilustración 4-6, los valores obtenidos se muestran en la Tabla 4-7.



Ilustración 4-6. Evidencia de la toma de datos para la estabilidad de la primera canción

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Tabla 4-7. Prueba de estabilidad del tiempo de reproducción de la primera canción

| Número de muestras | Hora | Tiempo medido [s] |
|--------------------|----------|-------------------|
| 1 | 10:00:00 | 29.52 |
| 2 | 10:00:35 | 29.31 |
| 3 | 10:01:10 | 29.51 |
| 4 | 10:01:45 | 29.24 |
| 5 | 10:02:20 | 29.01 |
| 6 | 10:02:55 | 29.57 |
| 7 | 10:03:30 | 29.37 |
| 8 | 10:04:05 | 29.38 |
| 9 | 10:04:40 | 29.37 |
| 10 | 10:05:15 | 29.36 |
| 11 | 10:05:50 | 29.41 |
| 12 | 10:06:25 | 29.3 |
| 13 | 10:07:00 | 29.41 |
| 14 | 10:07:35 | 29.44 |
| 15 | 10:08:10 | 29.23 |
| 16 | 10:08:45 | 29.37 |
| 17 | 10:09:20 | 29.24 |
| 18 | 10:09:55 | 29.34 |
| 19 | 10:10:30 | 29.51 |
| 20 | 10:11:05 | 29.51 |
| 21 | 10:11:40 | 29.44 |
| 22 | 10:12:15 | 29.44 |
| 23 | 10:12:50 | 29.39 |
| 24 | 10:13:25 | 29.36 |
| 25 | 10:14:00 | 29.35 |
| 26 | 10:14:35 | 29.37 |

| Número de muestras | Hora | Tiempo medido [s] |
|----------------------------|----------|-------------------|
| 27 | 10:15:10 | 29.39 |
| 28 | 10:15:45 | 29.44 |
| 29 | 10:16:20 | 29.36 |
| 30 | 10:16:55 | 29.23 |
| Media | | 29.37 |
| Desviación estándar | | 0.11 |
| CV | | 0.38 % |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En base a los cálculos de la Tabla 4-7, se tiene que el coeficiente de variación del tiempo de reproducción de la canción uno es de 0.38%, determinando que la estabilidad mediante el criterio de (Romero Ramos 2016) es menor al 5%, concluyendo que la estabilidad de reproducción en la primera canción es óptima.

4.4.2 Estabilidad de reproducción de la última canción

Se determina el tiempo de reproducción de la última canción durante un periodo de tiempo, manteniendo las mismas condiciones de funcionamiento. La toma de muestras se inició a las 10:30:00, con un intervalo de ochenta segundos entre muestras por medio de la función de cronometro de un dispositivo móvil como se observa en la Ilustración 4-7, los valores obtenidos se muestran en la Tabla 4-8.



Ilustración 4-7. Evidencia de la toma de datos para la estabilidad de la última canción

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Tabla 4-8. Prueba de estabilidad del tiempo de reproducción de la última canción

| Número de muestras | Hora | Tiempo medido [s] |
|----------------------------|----------|-------------------|
| 1 | 10:30:00 | 71.98 |
| 2 | 10:31:20 | 72.09 |
| 3 | 10:32:40 | 71.87 |
| 4 | 10:34:00 | 71.97 |
| 5 | 10:35:20 | 72.1 |
| 6 | 10:36:40 | 71.88 |
| 7 | 10:38:00 | 71.93 |
| 8 | 10:39:20 | 71.86 |
| 9 | 10:40:40 | 71.84 |
| 10 | 10:42:00 | 71.98 |
| 11 | 10:43:20 | 71.9 |
| 12 | 10:44:40 | 71.84 |
| 13 | 10:46:00 | 71.92 |
| 14 | 10:47:20 | 71.91 |
| 15 | 10:48:40 | 71.81 |
| 16 | 10:50:00 | 71.84 |
| 17 | 10:51:20 | 72.11 |
| 18 | 10:52:40 | 71.83 |
| 19 | 10:54:00 | 71.86 |
| 20 | 10:55:20 | 71.83 |
| 21 | 10:56:40 | 71.91 |
| 22 | 10:58:00 | 71.88 |
| 23 | 10:59:20 | 71.89 |
| 24 | 11:00:40 | 71.85 |
| 25 | 11:02:00 | 71.83 |
| 26 | 11:03:20 | 72.06 |
| 27 | 11:04:40 | 72.94 |
| 28 | 11:06:00 | 71.85 |
| 29 | 11:07:20 | 71.81 |
| 30 | 11:08:40 | 71.81 |
| Media | | 71.94 |
| Desviación estándar | | 0.21 |
| CV | | 0.29 % |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En base a los cálculos de la Tabla 4-8, se tiene que el coeficiente de variación del tiempo de reproducción de la última canción es de 0.29%, determinando que la estabilidad mediante el criterio de (Romero Ramos 2016) es menor al 5%, concluyendo que la estabilidad de reproducción en la última canción es óptima.

4.5 Precisión de desplazamiento del prototipo

Se tiene como objetivo verificar la precisión de desplazamiento de cada mano del prototipo, para esto se tomó como guía el dedo índice de cada mano, para obtener los datos para esto se lo realizó por medio del cálculo del error relativo. Para esta prueba se consideró una tecla inicial y una final, con una dimensión de 2.1 cm de ancho, para lo cual se toma como dato patrón el punto medio de cada tecla la cual es de 1.05 cm medidos de derecha a izquierda con una regla de la marca “Estilo” la cual posee un error del 1%, como se observa en la Ilustración 4-8.



Ilustración 4-8. Medidas tomadas entre la tecla inicial y final

Realizado por: (Aguirre Marco y Meneses Jean 2023)

4.5.1 Precisión de desplazamiento de la mano derecha

La prueba de estabilidad en la tecla inicial del dedo guía de la mano derecha tiene como objetivo calcular el error absoluto y el error relativo respecto a la tecla “A4” como se muestra en la Ilustración 4-9.

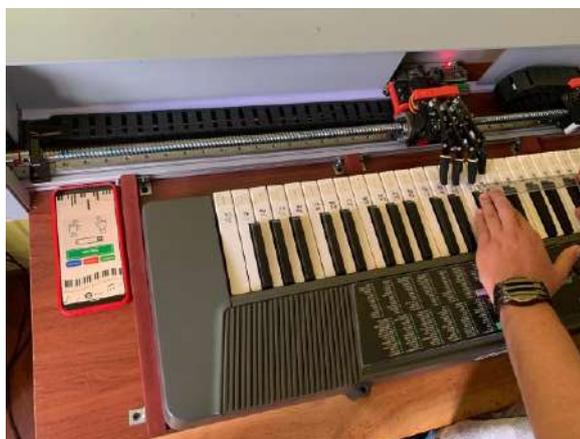


Ilustración 4-9. Evidencia de la toma de muestras de la mano derecha en la tecla inicial

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 4-9 se realiza el registro del cálculo del error absoluto y relativo respecto a la tecla inicial del dedo guía de la mano derecha.

Tabla 4-9. Error absoluto y relativo del dedo guía de la mano derecha en la tecla inicial

| Número de muestras | Medición con equipo patrón [cm] | Dedo guía mano derecha tecla inicial [cm] | Error absoluto [cm] | Error relativo (%) |
|--------------------|---------------------------------|---|---------------------|--------------------|
| 1 | 1.05 | 1.05 | 0 | 0 |
| 2 | 1.05 | 1.05 | 0 | 0 |
| 3 | 1.05 | 1.05 | 0 | 0 |
| 4 | 1.05 | 1 | 0.05 | 4.76 |
| 5 | 1.05 | 1 | 0.05 | 4.76 |
| 6 | 1.05 | 1.05 | 0 | 0 |
| 7 | 1.05 | 1 | 0.05 | 4.76 |
| 8 | 1.05 | 1.05 | 0 | 0 |
| 9 | 1.05 | 1.05 | 0 | 0 |
| 10 | 1.05 | 1 | 0.05 | 4.76 |
| 11 | 1.05 | 1 | 0.05 | 4.76 |
| 12 | 1.05 | 1 | 0.05 | 4.76 |
| 13 | 1.05 | 1 | 0.05 | 4.76 |
| 14 | 1.05 | 1.05 | 0 | 0 |
| 15 | 1.05 | 1.05 | 0 | 0 |
| 16 | 1.05 | 1.05 | 0 | 0 |
| 17 | 1.05 | 1.05 | 0 | 0 |
| 18 | 1.05 | 1.05 | 0 | 0 |
| 19 | 1.05 | 1.05 | 0 | 0 |
| 20 | 1.05 | 1.05 | 0 | 0 |
| 21 | 1.05 | 1 | 0.05 | 4.76 |
| 22 | 1.05 | 1 | 0.05 | 4.76 |
| 23 | 1.05 | 1.05 | 0 | 0 |
| 24 | 1.05 | 1.05 | 0 | 0 |
| 25 | 1.05 | 1.05 | 0 | 0 |
| 26 | 1.05 | 1 | 0.05 | 4.76 |
| 27 | 1.05 | 1.05 | 0 | 0 |
| 28 | 1.05 | 1.05 | 0 | 0 |
| 29 | 1.05 | 1.05 | 0 | 0 |
| 30 | 1.05 | 1.05 | 0 | 0 |
| Promedio | | | ± 0.02 | 1.59 |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Basándose en los cálculos de la Tabla 4-9 se llega a determinar que la precisión del dedo guía al momento de desplazarse para tocar la tecla inicial según (Bell 2003) es buena, dado que posee un error relativo del 1.59%.

La prueba de estabilidad en la tecla final del dedo guía de la mano derecha tiene como objetivo calcular el error absoluto y el error relativo respecto a la tecla “G6” como se muestra en la Ilustración 4-10.



Ilustración 4-10. Evidencia de la toma de muestras de la mano derecha en la tecla final

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 4-10 se evidencia los datos medidos por medio del equipo patrón de derecha a izquierda.

Tabla 4-10. Error absoluto y relativo del dedo guía de la mano derecha en la tecla final

| Número de muestras | Medición con equipo patrón [cm] | Dedo guía mano derecha tecla final [cm] | Error absoluto [cm] | Error relativo (%) |
|--------------------|---------------------------------|---|---------------------|--------------------|
| 1 | 1.05 | 1.1 | 0.05 | 4.76 |
| 2 | 1.05 | 1.2 | 0.15 | 14.29 |
| 3 | 1.05 | 1.1 | 0.05 | 4.76 |
| 4 | 1.05 | 1.1 | 0.05 | 4.76 |
| 5 | 1.05 | 1.1 | 0.05 | 4.76 |
| 6 | 1.05 | 1.1 | 0.05 | 4.76 |
| 7 | 1.05 | 1.2 | 0.15 | 14.29 |
| 8 | 1.05 | 1.2 | 0.15 | 14.29 |
| 9 | 1.05 | 1.2 | 0.15 | 14.29 |
| 10 | 1.05 | 1.2 | 0.15 | 14.29 |
| 11 | 1.05 | 1.1 | 0.05 | 4.76 |
| 12 | 1.05 | 1.1 | 0.05 | 4.76 |
| 13 | 1.05 | 1.1 | 0.05 | 4.76 |
| 14 | 1.05 | 1.1 | 0.05 | 4.76 |
| 15 | 1.05 | 1.1 | 0.05 | 4.76 |
| 16 | 1.05 | 1.2 | 0.15 | 14.29 |

| Número de muestras | Medición con equipo patrón [cm] | Dedo guía mano derecha tecla final [cm] | Error absoluto [cm] | Error relativo (%) |
|--------------------|---------------------------------|---|---------------------|--------------------|
| 17 | 1.05 | 1.2 | 0.15 | 14.29 |
| 18 | 1.05 | 1.1 | 0.05 | 4.76 |
| 19 | 1.05 | 1.1 | 0.05 | 4.76 |
| 20 | 1.05 | 1.1 | 0.05 | 4.76 |
| 21 | 1.05 | 1.1 | 0.05 | 4.76 |
| 22 | 1.05 | 1.2 | 0.15 | 14.29 |
| 23 | 1.05 | 1.1 | 0.05 | 4.76 |
| 24 | 1.05 | 1.1 | 0.05 | 4.76 |
| 25 | 1.05 | 1.1 | 0.05 | 4.76 |
| 26 | 1.05 | 1.2 | 0.15 | 14.29 |
| 27 | 1.05 | 1.2 | 0.15 | 14.29 |
| 28 | 1.05 | 1.2 | 0.15 | 14.29 |
| 29 | 1.05 | 1.1 | 0.05 | 4.76 |
| 30 | 1.05 | 1.1 | 0.05 | 4.76 |
| Promedio | | | ± 0.09 | 8.25 |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Basándose en los cálculos de la Tabla 4-10 se llega a determinar que la precisión del dedo guía al momento de desplazarse para tocar la tecla final según (Bell 2003) es aceptable dado que posee un error relativo del 8.25%.

Al comparar el error relativo entre la tecla inicial y final se evidencio que existe un aumento del 6.66% llegando a la conclusión que a pesar del aumento del error relativo no afecta a la interpretación de la nota.

4.5.2 Precisión de desplazamiento de la mano izquierda

La prueba de estabilidad en la tecla inicial del dedo guía de la mano izquierda tiene como objetivo calcular el error absoluto y el error relativo respecto a la tecla “F4” como se muestra en la Ilustración 4-11.



Ilustración 4-11. Evidencia de la toma de muestras de la mano izquierda en la tecla inicial

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 4-11 se realiza el registro del cálculo del error absoluto y relativo respecto a la tecla inicial del dedo guía de la mano izquierda.

Tabla 4-11. Error absoluto y relativo del dedo guía de la mano izquierda en la tecla inicial

| Número de muestras | Medición con equipo patrón [cm] | Dedo guía mano izquierda tecla inicial [cm] | Error absoluto [cm] | Error relativo (%) |
|--------------------|---------------------------------|---|---------------------|--------------------|
| 1 | 1.05 | 1 | 0.05 | 4.76 |
| 2 | 1.05 | 1 | 0.05 | 4.76 |
| 3 | 1.05 | 1.05 | 0 | 0 |
| 4 | 1.05 | 1.05 | 0 | 0 |
| 5 | 1.05 | 1.05 | 0 | 0 |
| 6 | 1.05 | 1.05 | 0 | 0 |
| 7 | 1.05 | 1.05 | 0 | 0 |
| 8 | 1.05 | 1 | 0.05 | 4.76 |
| 9 | 1.05 | 1 | 0.05 | 4.76 |
| 10 | 1.05 | 1 | 0.05 | 4.76 |
| 11 | 1.05 | 1.05 | 0 | 0 |
| 12 | 1.05 | 1.05 | 0 | 0 |
| 13 | 1.05 | 1.05 | 0 | 0 |
| 14 | 1.05 | 1 | 0.05 | 4.76 |
| 15 | 1.05 | 1.05 | 0 | 0 |
| 16 | 1.05 | 1.05 | 0 | 0 |
| 17 | 1.05 | 1.05 | 0 | 0 |
| 18 | 1.05 | 1 | 0.05 | 4.76 |
| 19 | 1.05 | 1 | 0.05 | 4.76 |
| 20 | 1.05 | 1.05 | 0 | 0 |
| 21 | 1.05 | 1.05 | 0 | 0 |

| Número de muestras | Medición con equipo patrón [cm] | Dedo guía mano izquierda tecla inicial [cm] | Error absoluto [cm] | Error relativo (%) |
|--------------------|---------------------------------|---|---------------------|--------------------|
| 22 | 1.05 | 1.05 | 0 | 0 |
| 23 | 1.05 | 1.05 | 0 | 0 |
| 24 | 1.05 | 1.05 | 0 | 0 |
| 25 | 1.05 | 1 | 0.05 | 4.76 |
| 26 | 1.05 | 1 | 0.05 | 4.76 |
| 27 | 1.05 | 1 | 0.05 | 4.76 |
| 28 | 1.05 | 1.05 | 0 | 0 |
| 29 | 1.05 | 1.05 | 0 | 0 |
| 30 | 1.05 | 1.05 | 0 | 0 |
| Promedio | | | ± 0.02 | 1.75 |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Basándose en los cálculos de la Tabla 4-11 se llega a determinar que la precisión del dedo guía al momento de desplazarse para tocar la tecla inicial según (Bell 2003) es buena, dado que posee un error relativo del 1.75%.

La prueba de estabilidad de la tecla final del dedo guía de la mano izquierda tiene como objetivo calcular el error absoluto y el error relativo respecto a la tecla “F3” como se muestra en la Ilustración 4-12.

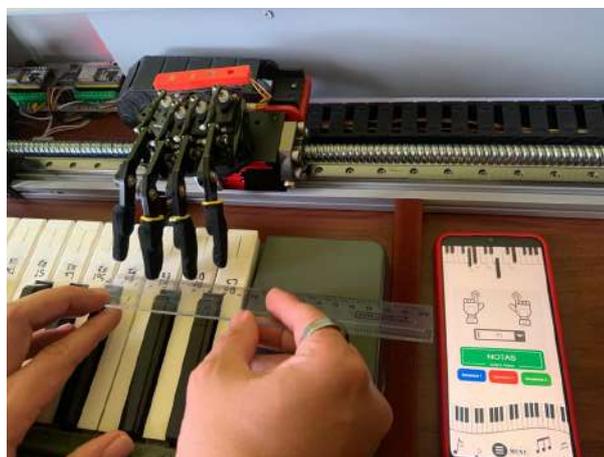


Ilustración 4-12. Evidencia para la toma de muestras de la mano izquierda en la tecla final

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 4-12 se evidencia los datos medidos por medio del equipo patrón de derecha a izquierda.

Tabla 4-12. Error absoluto y relativo del dedo guía de la mano izquierda en la tecla final

| Número de muestras | Medición con equipo patrón [cm] | Dedo guía mano izquierda tecla final [cm] | Error absoluto [cm] | Error relativo (%) |
|--------------------|---------------------------------|---|---------------------|--------------------|
| 1 | 1.05 | 1.1 | 0.05 | 4.76 |
| 2 | 1.05 | 1.2 | 0.15 | 14.29 |
| 3 | 1.05 | 1.1 | 0.05 | 4.76 |
| 4 | 1.05 | 1.1 | 0.05 | 4.76 |
| 5 | 1.05 | 1.1 | 0.05 | 4.76 |
| 6 | 1.05 | 1.2 | 0.15 | 14.29 |
| 7 | 1.05 | 1.2 | 0.15 | 14.29 |
| 8 | 1.05 | 1.1 | 0.05 | 4.76 |
| 9 | 1.05 | 1.1 | 0.05 | 4.76 |
| 10 | 1.05 | 1.1 | 0.05 | 4.76 |
| 11 | 1.05 | 1.1 | 0.05 | 4.76 |
| 12 | 1.05 | 1.2 | 0.15 | 14.29 |
| 13 | 1.05 | 1.1 | 0.05 | 4.76 |
| 14 | 1.05 | 1.1 | 0.05 | 4.76 |
| 15 | 1.05 | 1.1 | 0.05 | 4.76 |
| 16 | 1.05 | 1.2 | 0.15 | 14.29 |
| 17 | 1.05 | 1.2 | 0.15 | 14.29 |
| 18 | 1.05 | 1.2 | 0.15 | 14.29 |
| 19 | 1.05 | 1.2 | 0.15 | 14.29 |
| 20 | 1.05 | 1.1 | 0.05 | 4.76 |
| 21 | 1.05 | 1.1 | 0.05 | 4.76 |
| 22 | 1.05 | 1.1 | 0.05 | 4.76 |
| 23 | 1.05 | 1.1 | 0.05 | 4.76 |
| 24 | 1.05 | 1.1 | 0.05 | 4.76 |
| 25 | 1.05 | 1.2 | 0.15 | 14.29 |
| 26 | 1.05 | 1.1 | 0.05 | 4.76 |
| 27 | 1.05 | 1.2 | 0.15 | 14.29 |
| 28 | 1.05 | 1.2 | 0.15 | 14.29 |
| 29 | 1.05 | 1.2 | 0.15 | 14.29 |
| 30 | 1.05 | 1.2 | 0.15 | 14.29 |
| Promedio | | | ± 0.09 | 8.89 |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Basándose en los cálculos de la Tabla 4-12 se llega a determinar que la precisión del dedo guía al momento de desplazarse para tocar la tecla final según (Bell 2003) es aceptable dado que posee un error relativo del 8.89%.

Al comparar el error relativo entre la tecla inicial y final se evidencio que existe un aumento del 7.14% llegando a la conclusión que a pesar del aumento del error relativo no afecta a la interpretación de la nota.

4.6 Comunicación

Esta prueba tiene como finalidad comprobar la existencia de comunicación en la aplicación móvil, base de datos, módulo maestro ESP32 y el módulo esclavo ESP32 para garantizar que el prototipo pueda interpretar las canciones seleccionadas en la aplicación móvil. Para lo cual se emplea el monitor serial del IDE de Arduino 2.1.1, como se muestra en el Anexo L, además se observa que coincida la información enviada desde la aplicación móvil con la receptada en el módulo maestro ESP32 y la información recibida en el módulo esclavo ESP32 como se muestra en la Ilustración 4-13. Para el registro de los datos se realizó mediante la conexión a internet con una velocidad de subida de 62.24 Mbps y una velocidad de bajada de 71.85 Mbps.



Ilustración 4-13. Evidencia de la toma de datos para la prueba de comunicación

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En la Tabla 4-13 se verifica el envío y recepción de la información entre la aplicación móvil, módulo maestro y esclavo del ESP32.

Tabla 4-13. Prueba de comunicación

| Número de muestras | Canciones aplicación móvil | Recepción en el módulo maestro ESP32 | Recepción en el módulo esclavo ESP32 | Tiempo de latencia de envío de respuesta del módulo maestro ESP32 [s] |
|--------------------|----------------------------|--------------------------------------|--------------------------------------|---|
| 1 | Canción 1 | Canción 1 | Canción 1 | 2.1 |
| 2 | Canción 1 | Canción 1 | Canción 1 | 2.49 |
| 3 | Canción 1 | Canción 1 | Canción 1 | 0.10 |
| 4 | Canción 1 | Canción 1 | Canción 1 | 0.10 |

| Número de muestras | Canciones aplicación móvil | Recepción en el módulo maestro ESP32 | Recepción en el módulo esclavo ESP32 | Tiempo de latencia de envío de respuesta del módulo maestro ESP32 [s] |
|--------------------|----------------------------|--------------------------------------|--------------------------------------|---|
| 5 | Canción 1 | Canción 1 | Canción 1 | 2.33 |
| 6 | Canción 1 | Canción 1 | Canción 1 | 2.18 |
| 7 | Canción 2 | Canción 2 | Canción 2 | 0.82 |
| 8 | Canción 2 | Canción 2 | Canción 2 | 1.0 |
| 9 | Canción 2 | Canción 2 | Canción 2 | 1.13 |
| 10 | Canción 2 | Canción 2 | Canción 2 | 1.14 |
| 11 | Canción 2 | Canción 2 | Canción 2 | 1.18 |
| 12 | Canción 2 | Canción 2 | Canción 2 | 0.86 |
| 13 | Canción 3 | Canción 3 | Canción 3 | 1.73 |
| 14 | Canción 3 | Canción 3 | Canción 3 | 1.54 |
| 15 | Canción 3 | Canción 3 | Canción 3 | 1.93 |
| 16 | Canción 3 | Canción 3 | Canción 3 | 1.49 |
| 17 | Canción 3 | Canción 3 | Canción 3 | 1.61 |
| 18 | Canción 3 | Canción 3 | Canción 3 | 1.74 |
| 19 | Canción 4 | Canción 4 | Canción 4 | 0.44 |
| 20 | Canción 4 | Canción 4 | Canción 4 | 1.09 |
| 21 | Canción 4 | Canción 4 | Canción 4 | 0.10 |
| 22 | Canción 4 | Canción 4 | Canción 4 | 0.10 |
| 23 | Canción 4 | Canción 4 | Canción 4 | 0.10 |
| 24 | Canción 4 | Canción 4 | Canción 4 | 0.10 |
| 25 | Canción 5 | Canción 5 | Canción 5 | 1.23 |
| 26 | Canción 5 | Canción 5 | Canción 5 | 1.29 |
| 27 | Canción 5 | Canción 5 | Canción 5 | 1.13 |
| 28 | Canción 5 | Canción 5 | Canción 5 | 0.93 |
| 29 | Canción 5 | Canción 5 | Canción 5 | 1.07 |
| 30 | Canción 5 | Canción 5 | Canción 5 | 1.76 |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

En base a la comparativa de la Tabla 4-13 se corrobora que existe una comunicación exitosa dado que se recibe la misma información enviada desde la aplicación móvil hacia los módulos con un tiempo de respuesta inmediato, mientras que al enviar la información del módulo maestro ESP32 hacia la aplicación móvil se genera una latencia que según (Banzi 2010) esta no representa inconvenientes al momento de su funcionamiento.

4.7 Consumo energético

La prueba tiene como objetivo verificar el consumo energético mediante la corriente medida por medio de una pinza amperimétrica KT-87N y sus datos técnicos consta en el Anexo M. Por lo cual se realiza la medición de consumo total energético del prototipo.

Los datos de las corrientes medidas del prototipo se muestran en la Tabla 4-14, en la Ilustración 4-14 se observa la toma de la corriente medida durante la reproducción de la canción 1, el resto de las medidas tomadas se evidencian en el Anexo N.

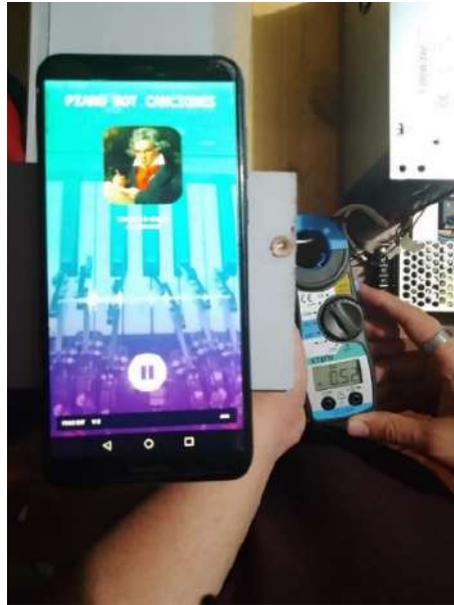


Ilustración 4-14. Corriente medida durante la reproducción de la canción 1

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Tabla 4-14. Consumo energético

| Número de muestras | Corriente medida [A] | Potencia [W] |
|--------------------|----------------------|--------------|
| Canción 1 | 0.52 | 10.58 |
| Canción 2 | 0.53 | 10.63 |
| Canción 3 | 0.54 | 10.68 |
| Canción 4 | 0.54 | 10.68 |
| Canción 5 | 0.55 | 10.73 |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

De la Tabla 4-14 se determina que el consumo de corriente máxima calculada con el equipo en funcionamiento es de 0.55 A y su potencia máxima es de 10.73 W.

4.8 Funcionamiento general del prototipo

Esta prueba tiene como objetivo validar el prototipo mediante la apreciación del pianista profesional Kenny Bonilla de gran trayectoria en el área de la música como se muestra en la Ilustración 4-15. Para lo cual se analiza los requerimientos establecidos anteriormente en la sección 3.1, y además se valora los parámetros de armonía, melodía y ritmo para cada canción.



Ilustración 4-15. Observación del prototipo por el pianista Kenny Bonilla

Realizado por: (Aguirre Marco y Meneses Jean 2023)

Según la valoración del pianista Kenny Bonilla después de escuchar las cinco canciones, determinó que el prototipo cumple al 100% y de una manera óptima con los parámetros fundamentales para la interpretación de canciones en un piano, además argumentó que el prototipo es novedoso y puede revolucionar el campo musical.

4.9 Análisis de costo del prototipo

Este análisis permite cuantificar el costo real del prototipo como se muestra en la Tabla 4-15, efectuando la suma de los costos de todos los elementos contemplando la parte electrónica, mecánica y estructural del prototipo.

De la Tabla 4-15, se determina que el prototipo está evaluado en \$1325 sin tomar en cuenta las horas de estudio, programación y construcción del prototipo. Del costo total se concluye que se obtuvo un 59.09% de gastos en componentes electrónicos y el 22.03% en estructurales.

Tabla 4-15. Análisis de costo del prototipo

| Descripción | Cantidad | Precio unitario [\$] | Precio total [\$] |
|---------------------------------------|----------|----------------------|-------------------|
| Tornillo de mesa lineal FSL40 | 2 | 250 | 500 |
| Controlador de motor DM542T | 2 | 35 | 70 |
| Servomotor LFD-01 9G | 8 | 9 | 72 |
| ESP32 | 2 | 14 | 28 |
| Sensores magnéticos | 2 | 2 | 4 |
| Fuente de alimentación T-250W 24V 10A | 1 | 59 | 59 |
| Fuente de alimentación RS25-5 | 1 | 20 | 20 |

| Descripción | Cantidad | Precio unitario [\$] | Precio total [\$] |
|-------------------------------|-----------------|-----------------------------|--------------------------|
| Cadena de arrastre | 2 | 16 | 32 |
| Estructura de manos robóticas | 2 | 80 | 160 |
| Teclado electrónico (Piano) | 1 | 250 | 250 |
| Elementos varios | - | 30 | 30 |
| Estructura del prototipo | - | 70 | 70 |
| Gastos extras | - | 30 | 30 |
| Total | | | \$ 1325 |

Realizado por: (Aguirre Marco y Meneses Jean 2023)

CONCLUSIONES

- Se realizó el diseño y construcción de manos robóticas programables aplicadas al área del entretenimiento en el campo musical, por medio de la revisión bibliográfica de robots pianistas y cumpliendo con los requerimientos establecidos para su desarrollo el cual incorpora motores paso a paso, servomotores, tarjetas de desarrollo ESP32, además posee una aplicación móvil para la selección de las canciones la cual tiene conexión Wi-Fi para su comunicación.
- De acuerdo con las pruebas de validación de los sensores magnéticos de la mano derecha el error relativo es de 1.43% y el valor de la mano izquierda es de 1.56%, por lo cual se determina que el error relativo es menor al 2% con lo que se considera que la distancia de los sensores es adecuada para el funcionamiento del prototipo.
- De acuerdo con la prueba de estabilidad del sensor magnético de la mano derecha se evidencio que el coeficiente de variación es de 3.06% y el valor de la mano izquierda es de 3.28%, por lo que se concluye que las mediciones de los sensores son fiables para el funcionamiento del prototipo.
- En base a la prueba de estabilidad del tiempo de reproducción de la primera canción se evidencio que el coeficiente de variación es de 0.38% y el valor de la última canción es de 0.29%, por lo que se concluye cuando el coeficiente de variación es menor al 5% las mediciones de los tiempos de reproducción son confiables para la correcta interpretación de las canciones ya que las variaciones son casi nulas.
- De acuerdo con la prueba de precisión de desplazamiento el error relativo en la tecla inicial de la mano derecha es de 1.59% y el valor en la mano izquierda es de 1.75%, por lo cual se determina que la precisión de desplazamiento es adecuada.
- De acuerdo con la prueba de precisión de desplazamiento el error relativo en la tecla final de la mano derecha es de 8.25% y el valor en la mano izquierda es de 8.89%, por lo cual se concluye que, a pesar de aumentar el error relativo en cada mano, este no interfiere al momento de interpretar la nota.
- En base a la prueba de comunicación se evidencio que existe una respuesta inmediata desde la aplicación móvil hacia el módulo maestro ESP32 y el módulo esclavo ESP32 llegando

siempre la información deseada, pero al momento de respuesta del módulo maestro ESP32 hacia la aplicación móvil se genera una latencia mínima la cual no genera problemas en la interpretación de las canciones.

- De acuerdo con la prueba de consumo energético del prototipo se determina que la corriente máxima medida es de 0.55 A y una potencia máxima de 10.73 W.
- En base a los proyectos investigados y la inexistencia de equipos comerciales se determina que no se puede realizar una comparación de costos para evaluar la rentabilidad del prototipo, por lo que el costo total es de \$1325 teniendo un 59.09% en gastos de componentes electrónicos y el 22.03% en gastos estructurales.

RECOMENDACIONES

- Incrementar el tiempo de pruebas del prototipo para identificar posibles mejoras en el diseño y construcción del prototipo de manos robóticas programables implementado.
- Se recomienda se estudie la posibilidad de implementar inteligencia artificial en el prototipo con la finalidad de implementar un aprendizaje autónomo para mejorar los tiempos y la ejecución al interpretar canciones.
- Se recomienda estudiar diferentes tipos de actuadores con niveles de ruido nulos para evitar interferencias acústicas al momento de interpretar una melodía, mejorar la velocidad de desplazamiento y facilitar la retroalimentación en la posición de las manos y dedos con la finalidad de mejorar la calidad del desplazamiento y accionamiento entre tecla
- Se recomienda el estudio de redes neuronales para implementar en el prototipo la capacidad de buscar, codificar e interpretar temas musicales específicos solicitados por medio de comandos de voz o al ser reproducida una melodía.
- Se recomienda incursionar, tomando como base esta investigación, en el desarrollo de robots enfocados al área del entretenimiento, que tengan la capacidad de adaptarse al uso de diferentes instrumentos musicales.
- Se recomienda patentar y difundir los resultados de los trabajos de investigación desarrollados en la Facultad con la finalidad de publicitar las capacidades de sus graduados, y captar recursos económicos para impulsar posibles emprendimientos.

BIBLIOGRAFÍA

ACOSTA, Robert et al., 2021. Caracterización de los motores paso a paso y su aplicación. Vol. 5, pp. 99-113.

ALBA PADILLA, Carlos Antonio y MUÑOZ GUILLEN, María Gabriela, 2017. Estudio y fabricación de una batería ácido plomo. *Temas de Ciencia y Tecnología*. Vol. 21, pp. 23-28.

ALBERTO BUITRAGO SUESCÚN, Yeison, CONSUELO PRIETO RAMÍREZ, Jennyfer y AUGUSTO PEÑA CORTÉS, César, 2012. Robot educativo de bajo costo que interpreta melodías en piano. [en línea]. Vol. 21, número 33, pp. 73-82. Recuperado a partir de: <https://www.redalyc.org/articulo.oa?id=413940772007>

ALFESSI, Pablo et al., 2020. *Robótica al servicio de la pandemia* [en línea]. Primera Ed. Chile. ISBN 9789567629503. Recuperado a partir de: <https://www.bcn.cl/observatorio/asiapacifico/noticias/publicacion-robotica-servicio-pandemia-2020>

ALVA, Jorgue Luis y ALCORTA, Natali Fiorella, 2020. *Sistemas embebidos. Guía metodológica para su desarrollo*. ISBN 9786124479137.

ANDROID STUDIO, 2020. Android Studio. [en línea]. 2020. Recuperado a partir de: <https://developer.android.com/studio/intro>.

ARBOLEDA, Andres Giovanni et al., 2014. *MÓDULO DE SERVOMOTOR 2. Medellín*.

ARDUINO, 2023. Arduino. [en línea]. 2023. Recuperado a partir de: <https://www.arduino.cc/reference/en/libraries/wifi/wificlient/>

ARIAS, Edison, 2016. *Implementación de un control de servomotores y motores a paso mediante Arduino-Matlab*.

AUTODESK, 2015. EAGLE / Fusion 360. [en línea]. 2015. Recuperado a partir de: <http://www.autodesk.com/products/3ds-max/overview>

BELL, David A., 2003. *Electronic Instrumentation and Measurements*. Second Edition. Ontario.

CABEZAS TERÁN, José Luis, 2016. *Diseño e implementación de un sistema electrónico de potencia híbrido para la carga de una batería tipo polímero de litio*. Quito.

CONTI, Francisco, 2005. Motores paso a paso. [en línea]. 29 abril 2005. Recuperado a partir de: https://books.google.com.ec/books?id=6yiXDwAAQBAJ&printsec=frontcover&hl=es&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false [accedido 19 abril 2023].

CONTRERAS, Bianca, 2022. *Base de datos en la nube* [en línea]. Recuperado a partir de: <https://www.computerweekly.com/es/definicion/Base-de-datos-en-la-nube>.

CORONA RAMÍREZ, Leonel, ABARCA JIMÉNEZ, Griselda y MARES CARREÑO, Jesús, 2014. Sensores y Actuadores.

CORPORATION, SolidWorks, 2002. SOLIDWORKS. [en línea]. 2002. Recuperado a partir de: <https://www.solidworks.com/es>

CORREA, Alejandra, 2021. AWS Amplify para desarrollo de aplicaciones web y móviles. [en línea]. 2021. Recuperado a partir de: <https://codster.io/blog/desarrollo-aplicaciones-aws-amplify/>

CUNNINGHAM, James et al., 2018. The Supernumerary Robotic 3rd Thumb for Skilled Music Tasks. *Proceedings of the IEEE RAS and EMBS International Conference on Biomedical Robotics and Biomechatronics*. Vol. 2018-Augus, número 644000, pp. 665-670. DOI 10.1109/BIOROB.2018.8487609.

FAHN, Chin Shyurng, TSAI, Cheng Feng y LIN, Yung Wei, 2017. Development of a novel two-hand playing piano robot. *International Conference of Control, Dynamic Systems, and Robotics*. Número 106, pp. 1-10. DOI 10.11159/cdsr17.106.

FIREBASE, 2023. Firebase. [en línea]. 2023. Recuperado a partir de: <https://firebase.google.com/docs/database?hl=es-419>

FUYUMOTION, 2010a. Brazo robótico de eje lineal de husillo de bolas compacto para trabajo pesado - Fuyu Technology Co., Ltd. [en línea]. 9 mayo 2010. Recuperado a partir de: <https://www.fuyumotion.com/es/heavy-duty-compact-ball-screw-linear-axis-robot-arm-product/> [accedido 17 mayo 2023].

FUYUMOTION, 2010b. Actuador lineal micro con riel de guía lineal pequeño y liviano de perfil de aluminio con motor paso a paso - Fuyu Technology Co., Ltd. [en línea]. 9 mayo 2010. Recuperado a partir de: <https://www.fuyumotion.com/es/aluminum-profile-small-and-light-linear-guide-rail-micro-linear-actuator-with-stepper-motor-product/> [accedido 17 mayo 2023].

GARRIDO COBO, Juan, 2013. TFC Desarrollo de Aplicaciones Móviles.

GONZÁLEZ GAVILANES, Henry Fabricio y CARRILLO TRUJILLO, María Gabriela, 2019. *Implementación de un prototipo de robot sembrador de papa en terrenos sin inclinación para pequeños productores.* Riobamba.

GONZÁLEZ, María, FLORES, Yadira y MUÑOZ, Claudia, 2021. Panorama de la robótica educativa a favor del aprendizaje STEAM. Vol. 18, número 2. DOI 10.25267/Rev.

GONZALEZ, Oscar, 2017. Comparativa y análisis completo de los módulos Wifi ESP8266 y ESP32. [en línea]. 2017. Recuperado a partir de: <https://blog.bricogeek.com/noticias/electronica/comparativa-y-analisis-completo-de-los-modulos-wifi-esp8266-y-esp32/>

GUERRA, José, 2020. ESP32 Wifi y Bluetooth en un solo chip. [en línea]. 2020. Recuperado a partir de: <https://programarfacil.com/esp8266/esp32/>

HENAO, Antonio et al., 2003. *Piano Complementario NIVEL 1* [en línea]. Programa Editorial. Colombia. Recuperado a partir de: <https://books.google.es/books?hl=es&lr=&id=FkP68SMrq8wC&oi=fnd&pg=PA7&dq=Piano:+tocar+en+clave+de+DO&ots=mBxjRQbTYu&sig=2ho5m5BTSbe2SP9rfoaouCIFgRE#v=onepage&q&f=false> [accedido 10 mayo 2023].

IKISS, Jamal, 2020. *Sistema de Adquisición de Datos con ESP32.*

JEN-CHANG, Lin et al., 2014. Design of Piano-playing Robotic Hand. *IAES International Journal of Robotics and Automation (IJRA)*. Vol. 3, número 2, pp. 118-130. DOI 10.11591/ijra.v3i2.4269.

LÓPEZ, Sara, 2020. Firebase: qué es, para qué sirve, funcionalidades y ventajas. [en línea]. 2020. Recuperado a partir de: <https://digital55.com/blog/que-es-firebase-funcionalidades-ventajas-conclusiones/>

MARTÍN CASTILLO, Juan Carlos, 2017. *Electrónica.* Madrid: Editex S.A. ISBN 9788491610038.

MEDINA, Rosgaby, 2022. Estadísticas de la situación Digital en Ecuador 2021-2022. [en línea]. 2022. Recuperado a partir de: <https://branch.com.co/marketing-digital/estadisticas-de-la-situacion-digital-en-ecuador-2021-2022/> [accedido 29 mayo 2023].

MICROSOFT, 2022. CloudKit en Xamarin.iOS. [en línea]. 2022. Recuperado a partir de: <https://learn.microsoft.com/es-es/xamarin/ios/data-cloud/intro-to-cloudkit>

MORDORINTELLIGENCE, 2023a. Mercado de Robótica Industrial: Crecimiento, Tendencias, Impacto de Covid-19 y Pronósticos (2023 - 2028). [en línea]. 2023. Recuperado a partir de: <https://www.mordorintelligence.com/es/industry-reports/industrial-robotics-market>

MORDORINTELLIGENCE, 2023b. Mercado de Robótica de Asistencia: Crecimiento, Tendencias, Impacto de Covid-19 y Pronósticos (2023 - 2028). [en línea]. 2023. Recuperado a partir de: <https://www.mordorintelligence.com/es/industry-reports/assistive-robotics-market>

MORDORINTELLIGENCE, 2023c. Mercado de Robots Educativos: Crecimiento, Tendencias, Impacto de Covid-19 y Pronósticos (2023 - 2028). [en línea]. 2023. Recuperado a partir de: <https://www.mordorintelligence.com/es/industry-reports/educational-robot-market>

MORDORINTELLIGENCE, 2023d. Mercado Global de Robots Quirúrgicos: Crecimiento, Tendencias, Impacto de Covid-19 y Pronósticos (2023 - 2028). [en línea]. 2023. Recuperado a partir de: <https://www.mordorintelligence.com/es/industry-reports/surgical-robots-market>

MORDORINTELLIGENCE, 2023e. Mercado Robótica de Almacén de América Latina: Crecimiento, Tendencias, Impacto de Covid-19 y Pronósticos (2023 - 2028). [en línea]. 2023. Recuperado a partir de: <https://www.mordorintelligence.com/es/industry-reports/latin-america-warehouse-robotics-market>

MORENO, Javier Martín y RUIZ FERNANDEZ, Daniel, 2007. Informe Técnico: Protocolo ZigBee (IEEE 802.15.4). p. 36.

MUÑOZ SEGURA, Cristian David, 2016. *Diseño y desarrollo de un aplicativo móvil en la plataforma Android Studio para la empresa Allied Electronics & Service (Servicios Electrónicos Aliados) Perú*. Pereira: Universidad Tecnológica de Pereira.

NACIONAL, Asamblea, 2016. *Código Orgánico de la Economía Social de los Conocimientos, Creatividad e Innovación*. Ecuador.

NOVILLO, Johnny et al., 2018. *Arduino y el Internet de las Cosas*. ISBN 9788494915185.

ORNELAS, Ruben Castro, 2022. *Robotic Finger Hardware and Controls Design for Dynamic Piano Playing* [en línea]. Recuperado a partir de: <https://dspace.mit.edu/handle/1721.1/144673>

PAWLAK, Maxime y PAWLAK, Renaud, 2018. *Acordes de piano para dummies..* p. 356.

PÉREZ CHILQUINGA, Alex Paúl, 2019. *Estudio comparativo entre sensores de distancia, con diferente principio de medición, para determinar las especificaciones en la implementación de un scanner tridimensional*. Quito.

PÉREZ, Iván y MENDOZA, Hellyss, 2008. Desarrollo de una mano robótica con capacidad de manipulación. Vol. 7, pp. 1-28.

PHAN, ANNA y MALM, LUKAS, 2019. *Music reading and piano playing robot*.

PJRC, 2023. 32-Bit Products and Accessories. [en línea]. 2023. Recuperado a partir de: <https://www.pjrc.com/store/index.html>

PORTUONDO PAISA, Yoel y PORTUONDO MORET, Juan, 2010. La Repetibilidad y reproductibilidad en el aseguramiento de la calidad de los procesos de medición. *Tecnología Química* [en línea]. Vol. XXX, número 2, pp. 117-121. Recuperado a partir de: <http://www.redalyc.org/articulo.oa?id=445543770014>

RAMÍREZ, Piedad, 2022. ¿Cuáles son los sistemas operativos más usados en 2022? [en línea]. 4 enero 2022. Recuperado a partir de: <https://itsoftware.com.co/content/sistemas-operativos-mas-usados/> [accedido 16 mayo 2023].

REA, Mercy, 2019. *Desarrollo de un prototipo de robot para rehabilitación de pacientes con lesiones en rodillas controlado mediante dispositivos móviles*.

REDAGRÍCOLA, 2022. El mercado de los robots agrícolas alcanzaría los 26.700 millones de dólares en 2027. [en línea]. 2022. Recuperado a partir de: <https://www.redagricola.com/cl/el-mercado-de-los-robots-agricolas-alcanzaria-los-26-700-millones-de-dolares-en-2027/>

RITSCH, Winfried, 2011. Robotic piano player making pianos talk. *Proceedings of the 8th Sound and Music Computing Conference, SMC 2011*.

RODRIGUEZ, Dolores, 2019. Tarjetas Para Desarrollo De Hardware. [en línea]. 2019. Recuperado a partir de: <https://es.scribd.com/document/403155578/Tarjetas-Para-Desarrollo-De-Hardware-docx>

ROMERO RAMOS, Eva, 2016. *Estadística para todos*. Primera Edición. Madrid.

SALAZAR, Jordi, 2012. *Redes Inalámbricas* [en línea]. ISBN 9781447152927. Recuperado a partir de: https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01_R_ES.pdf

SANTILLÁN RIVADENEIRA, María José y MALDONADO JARAMILLO, Daniel Alexis, 2017. *Desarrollo de una aplicación móvil para la estimulación cognitiva de adultos mayores que padecen alzheimer en fases ligera y moderada utilizando la plataforma Andorid Studio*. Macas: Escuela Superior Politécnica de Chimborazo.

SECRETARÍA DE EDUCACIÓN SUPERIOR, CIENCIA, Tecnología e Innovación, sin fecha. Los once principios de INGENIOS. [en línea]. Recuperado a partir de: <https://www.educacionsuperior.gob.ec/los-once-principios-de-ingenios/>

TOPPER, Anthony y MALONEY, Thomas, 2019. *Piano-Playing Robotic Arm*.

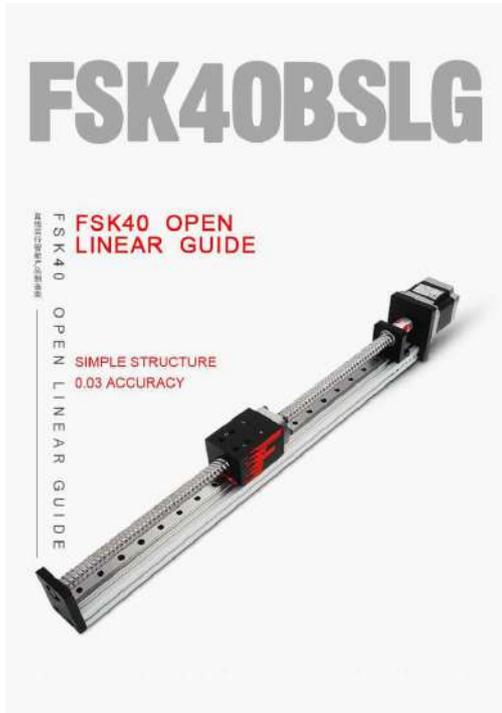
TORRENTE ARTERO, Óscar, 2013. *Arduino Curso Práctico de formación* [en línea]. España. Recuperado a partir de: www.FreeLibros.me

VILLASÍS-KEEVER, Miguel Ángel et al., 2018. Research protocol VII. Validity and reliability of the measurements. *Revista Alergia Mexico*. Vol. 65, número 4, pp. 414-421. DOI 10.29262/ram.v65i4.560.

XUKYO, 2018. Lista de servomotores. [en línea]. 11 septiembre 2018. Recuperado a partir de: <https://www.aranacorp.com/es/lista-de-servomotores/> [accedido 3 mayo 2023].

ANEXOS

ANEXO A: TORNILLO LINEAL FUYU FSL40



ALUMINUM ALLOY MATERIAL IS PREFERRED,
AND THE STRUCTURE IS COMPACT AND STRONG.
SMOOTH RUNNING AND DURABLE.
WITH THE CONTROLLER CAN ACHIEVE
HIGH-PRECISION LINEAR MOTION.
CUSTOMIZABLE LENGTH,
BUILD TWO-AXIS
AND THREE-AXIS POSITIONING SYSTEM,
PROVIDE INDUSTRY SOLUTIONS!



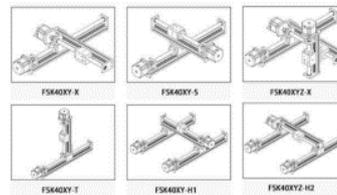
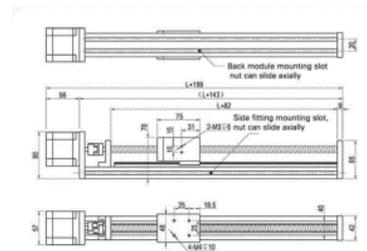
Special made motor with
max speed up to 1200rpm

XXXXXX



Oxidized Alu
to avoid rusting

XXXXXX



| | | | |
|---------------------------------------|-------------|-----|-----|
| Position Accuracy(mm) | ±0.03 | | |
| Ball Screw Diameter(mm) | φ12 | φ16 | φ16 |
| Lead(mm) | 4 | 5 | 10 |
| Max Horizontal Loading(N) | 25 | 25 | 20 |
| Max Vertical Loading(N) | 15 | 15 | 10 |
| Without Loading Speed(r/min) | 100 | 180 | 255 |
| Max Horizontal Full Speed(r/min) | 80 | 150 | 230 |
| Max Vertical Full Speed(r/min) | 40 | 90 | 50 |
| Max side mounted payload(N) | 5 | | |
| max side mounted speed(r/min) | 60 | 75 | 150 |
| Rated Acceleration(m/s ²) | 500 | | |
| Maximum thrust(N) | 471 | 376 | 188 |
| Storage temperature(°C) | -15°C~+60°C | | |

ANEXO B: FUENTE DE PODER T-250W 24V 10A

FUENTE DE ALIMENTACIÓN
24V / 10.4A / 250W

Fuente de alimentación conmutada
VCA/VCC 24V/250W
Ref. ALM317



El propósito de este documento es especificar los requisitos funcionales de la fuente de alimentación conmutada de tipo.

Características de salida:

| | |
|------------------------------|------------------------------------|
| Salida nominal | 250W |
| Corriente nominal | 10.4A |
| Rango control | 0-10Vdc |
| Corriente máxima | 20A |
| Respaldo sobre V | 2000µs |
| Aplicación de salida | 24V - 35.2V |
| Regulación de línea (regul.) | < 0.2% |
| Tiempo de subida a carga | 1.5s, 20ms(200mA / 3s, 20ms(100mA) |

Características de entrada:

| | |
|-----------------------|---------------------------|
| Volaje de entrada | 100-240Vca, Conmutación |
| Frecuencia | 50/60 |
| Corriente típica Max | 1.60/1.00ca - 10A(2000VA) |
| Corriente de arranque | 10A(2000VA) |
| Corriente de fuga Max | < 0.70mA(100Vca) |

Electronica Múgica S.A. - C/ Miraflores, 101, 3er. Piso de Zonas 3802 - Algiers 38100
tel: (+31) 020 261 22 22 - (+31) 020 261 22 22 - www.electronicamugica.com

Fuente de alimentación conmutada
24V / 10.4A / 250W

Función de protección:

Autoprotección: Protección sobre corriente de salida 100%, ± 10%.

Cuando se alcanza un nivel crítico de corriente de salida se activa la acción de protección, evitando que se alcance el nivel crítico de corriente de salida. El momento en que se detecta la falta de protección.

Corriente máxima: 20A - 35.2V

Cuando el voltaje de salida alcanza el punto de saturación de salida, la protección corta automáticamente la salida. La función de auto-recuperación entre niveles de protección para permitir la recuperación de trabajo.

Requisito ambiental:

Temperatura de trabajo con y sin carga: 0°C a +40°C (2) a 80% RH (no condensación)

Temperatura de almacenamiento: -40°C a +85°C (3) a 95% RH (humedad relativa)

Seguridad:

Normativa de seguridad: IEC60950 / IEC60335 / IEC60730 y sus modificaciones.

Restricción de aislamiento: IEC60950. Clase II y III con una impedancia de entrada de aislamiento IEC60950. (III) Class II/III.

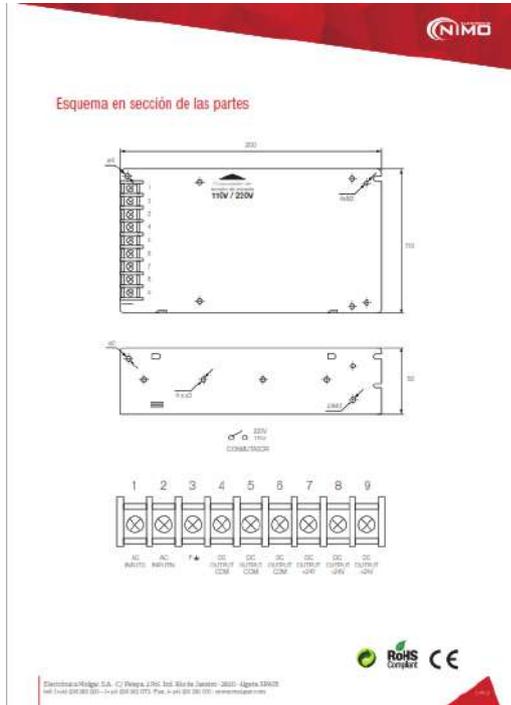
Certificaciones estándar:

CE, RoHS, REACH, IEC60950, IEC60335, IEC60730, IEC60950-1, IEC60950-2, IEC60950-3, IEC60950-4, IEC60950-5, IEC60950-6, IEC60950-7, IEC60950-8, IEC60950-9, IEC60950-10, IEC60950-11, IEC60950-12, IEC60950-13, IEC60950-14, IEC60950-15, IEC60950-16, IEC60950-17, IEC60950-18, IEC60950-19, IEC60950-20, IEC60950-21, IEC60950-22, IEC60950-23, IEC60950-24, IEC60950-25, IEC60950-26, IEC60950-27, IEC60950-28, IEC60950-29, IEC60950-30, IEC60950-31, IEC60950-32, IEC60950-33, IEC60950-34, IEC60950-35, IEC60950-36, IEC60950-37, IEC60950-38, IEC60950-39, IEC60950-40, IEC60950-41, IEC60950-42, IEC60950-43, IEC60950-44, IEC60950-45, IEC60950-46, IEC60950-47, IEC60950-48, IEC60950-49, IEC60950-50, IEC60950-51, IEC60950-52, IEC60950-53, IEC60950-54, IEC60950-55, IEC60950-56, IEC60950-57, IEC60950-58, IEC60950-59, IEC60950-60, IEC60950-61, IEC60950-62, IEC60950-63, IEC60950-64, IEC60950-65, IEC60950-66, IEC60950-67, IEC60950-68, IEC60950-69, IEC60950-70, IEC60950-71, IEC60950-72, IEC60950-73, IEC60950-74, IEC60950-75, IEC60950-76, IEC60950-77, IEC60950-78, IEC60950-79, IEC60950-80, IEC60950-81, IEC60950-82, IEC60950-83, IEC60950-84, IEC60950-85, IEC60950-86, IEC60950-87, IEC60950-88, IEC60950-89, IEC60950-90, IEC60950-91, IEC60950-92, IEC60950-93, IEC60950-94, IEC60950-95, IEC60950-96, IEC60950-97, IEC60950-98, IEC60950-99, IEC60950-100.

Requisito mecánico:

Consultar de entrada y salida: Clima de trabajo: Temperatura ambiente: 0°C a +40°C (2) a 80% RH (no condensación)

Electronica Múgica S.A. - C/ Miraflores, 101, 3er. Piso de Zonas 3802 - Algiers 38100
tel: (+31) 020 261 22 22 - (+31) 020 261 22 22 - www.electronicamugica.com



ANEXO D: SENSOR MAGNÉTICO

Door & Window Magnetic Sensor Switch for Arduino / IoT / Alarm System



MC-38 Wired Door/Window Sensor | Magnetic Switch | Home Alarm System. Reversible style (which means they can be "set into" for example: a door or window). The MC38 can be wired to your door, or window, any where you want a magnetic sensor to alarm when opened.

Metal shield anti fire ABS, the alarm sounds when the magnets separated. No external power supply is required— simply connect to wired or wireless alarm control panel (WSD and VAC, perm directly).

SPECIFICATIONS

- Constructing Made: N.C.
- Rated current: 100mA
- Rated voltage: 200VDC
- Operating distance: more than 15mm, less than 25mm
- Rated power: 3W
- Dimension: 28x15x0.9mm
- Cable Length: 30.2cm ± 12mm
- Switch output: normally closed (switch and magnet are together when the switch is closed)

FEATURES:

- Easy installation/Reliable performance
- Good characteristic of abrasion-proof
- Best Choice for you to protect family

For purchase & enquiries, please contact sales@synacorp.com.my or call 04-5041817

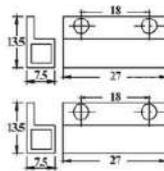
APPLICATION:

- Easy installation/Reliable performance
- Good characteristic of abrasion-proof
- Best Choice for you to protect family
- There are two types of reed switches: "normally open" reed switches and "normally closed" reed switches.
- The metal reeds on a normally open switch stay open when there is no magnet near the switch. In the presence of a magnetic field, the contacts of a normally-open reed switch will close. A normally-closed reed switch is closed when it is not near a magnet, as a magnet is brought close to it, a normally-closed switch will open.

PIN-OUTS:

The switch is non-polar, so you can plug in the wires in any way.

DIMENSIONS (MM):



For purchase & enquiries, please contact sales@synacorp.com.my or call 04-5041817

ANEXO F: CONTROLADOR PARA MOTOR PASO A PASO DM542T

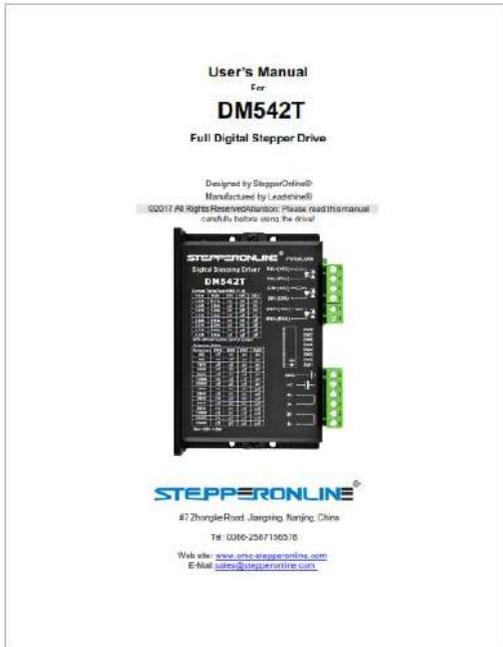


Table of Contents

- 1. Introduction, Features and Applications..... 1
 - 1.1 Introduction..... 1
 - 1.2 Features..... 1
 - 1.3 Applications..... 1
- 2. Specifications..... 1
 - 2.1 Electrical Specifications (T_a = 25°C/77°F)..... 1
 - 2.2 Operating Environment and other Specifications..... 2
 - 2.3 Mechanical Specifications (unit: mm (1 inch=25.4mm))..... 2
 - 2.4 Elimination of Heat..... 2
- 3. Pin Assignment and Description..... 3
 - 3.1 Connector P1 Configurations..... 3
 - 3.2 Connector P2 Configurations..... 3
- 4. Control Signal Connector (P1) Interface..... 3
 - 4.1 Connections of 4-lead Motor..... 4
 - 4.2 Connections of 6-lead Motor..... 4
 - 4.3 Half-Cool Configurations..... 4
 - 4.4 Full-Cool Configurations..... 5
 - 4.5 Connections of 8-lead Motor..... 5
 - 4.6 Series Connections..... 5
 - 4.7 Parallel Connections..... 5
- 5. Power Supply Selection..... 6
 - 5.1 Regulated or Unregulated Power Supply..... 6
 - 5.2 Multiple Drives..... 6
 - 5.3 Selecting Supply Voltage..... 6
- 6. Selecting Microstep Resolution and Drive Output Current..... 6
 - 6.1 Microstep Resolution Selection..... 6
 - 6.2 Current Settings..... 7
 - 6.3 Dynamic Current Setting..... 7
 - 6.4 Stallhold Current Setting..... 7
- 7. Wiring Notes..... 8
- 8. Typical Connection..... 8
- 9. Sequence Chart of Control Signals..... 9
- 10. Protection Functions..... 9
- 11. Frequently Asked Questions..... 10
 - 11.1 Problem Symptoms and Possible Causes..... 10

STEPPERONLINE® Full Digital Stepper Drive DM542T

1. Introduction, Features and Applications

Introduction

The DM542T is a fully digital stepper drive developed with advanced DSP control algorithms based on the latest motion control technology. It has achieved a unique level of system smoothness, providing optimal torque and fully mid-range scalability. Its motor auto-identification and parameter auto-configuration features offer quick setup to optimal modes with different motors. Compared with traditional analog drives, DM542T can drive a stepper motor at much lower noise, lower heating, and smoother movement. Its unique features make DM542T an ideal choice for high requirement applications.

Features

- Anti-Resonance provides optimal torque and fully mid-range scalability
- Motor auto-identification and parameter auto-configuration when power on, offer optimal responses with different motors
- Multi-Stepping allows a low resolution step input to produce a higher microstep output, thus offers smoother motor movement
- 15 selectable microstep resolutions including 400, 800, 1600, 3200, 6400, 12800, 25600, 1000, 2000, 4000, 5000, 10000, 10000, 20000, 25000
- Soft-start with an "auto" when powered on
- Input voltage: 20-29VDC
- 8 selectable peak current including 1.80A, 1.86A, 1.91A, 2.77A, 2.84A, 3.11A, 3.36A, 4.20A
- Pulse input frequency up to 200 KHz, TTL compatible and optically isolated input
- Automatic idle-current reduction
- Suitable for 2-phase and 4-phase motors
- Support PULLER mode
- Over-voltage and over-current protection

Application

Suitable for a wide range of stepper motor, size from NEMA17 to 24. It can be used in various kinds of machines, such as X-Y tables, engraving machines, blading machines, laser cutters, pick-place devices and so on. Particularly adapt to the applications with low noise, low heating, high speed and high precision.

2. Specifications

Electrical Specification (T_a = 25°C/77°F)

| Parameter | Min | Typical | Max | Unit |
|-----------------------------|-----|---------|--------------|------|
| Output Peak Current | 1.0 | - | 4.2 (10.93A) | A |
| Input Voltage Logic | +28 | -34 | +50 | VDC |
| Signal Current Pulse | 7 | 10 | 16 | mA |
| Input Frequency Pulse Width | 0 | - | 200 | μs |
| Isolation Resistance | 500 | - | - | MΩ |

1 | www.stepperonline.com | Email: sales@stepperonline.com | Web: www.stepperonline.com

STEPPERONLINE® Full Digital Stepper Drive DM542T

Operating Environment and other Specifications

| Using | Natural Cooling or Forced cooling | |
|-----------------------|-----------------------------------|--|
| | Environment | Avoid dust, oil flog and corrosive gases |
| Operating Environment | Ambient Temperature | 4T ~ 65T |
| | Humidity | 40%RH ~ 90%RH |
| | Operating Temperature | +10C ~ 42C |
| Storage Temperature | Vibration | 10MM/s (1/3 Sec) |
| | Weight | 280g (9.8oz) |

Mechanical Specifications (unit: mm (1 inch=25.4mm))

Figure 1: Mechanical specifications
*Recommend one side mounting for better heat dissipation

Elimination of Heat

- Drives reliable working temperature should be -42°C(113 F), and motor working temperature should be $+80$(176F).
- It is recommended to use automatic idle-current mode, namely current automatically reduce to 50% when motor stops, so as to reduce drive heating and motor heating.
- It is recommended to mount the drive vertically to maximize heat sink area. Use forced cooling method to cool the system if necessary.

2 | www.stepperonline.com | Email: sales@stepperonline.com | Web: www.stepperonline.com

ANEXO G: MICROCONTROLADOR ESP32

ESP32-WROOM-32 (ESP-WROOM-32)
Datasheet
Version 2.4



Espressif Systems

About This Guide

This document provides the specifications for the ESP32-WROOM-32(ESP-WROOM-32) module.

Revision History

For revision history of this document, please refer to the last page.

Documentation Change Notification

Espressif periodically notifies to keep customers updated on changes to technical documentation. Please subscribe [here](#).

Certification

Download certificates for Espressif products from [here](#).

Disclaimer and Copyright Notice

Information in this document, including URL references, is subject to change without notice. THIS DOCUMENT IS PROVIDED AS-IS WITH NO WARRANTIES, WARRANTIES, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE. All liability, including liability for infringement of any proprietary rights, relating to use of information in this document is disclaimed. No license express or implied, by estoppel or otherwise, to any intellectual property rights are granted herein. The WiFi Alliance Member logo is a trademark of the WiFi Alliance. The Bluetooth logo is a registered trademark of Bluetooth SIG. All trade names, trademarks and registered trademarks mentioned in this document are property of their respective owners, and are hereby acknowledged. Copyright © 2018 Espressif Inc. All rights reserved.

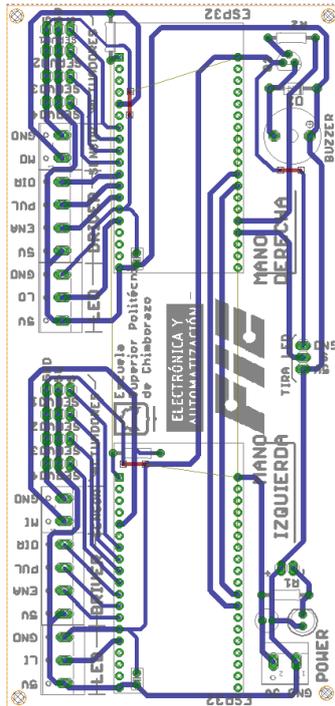
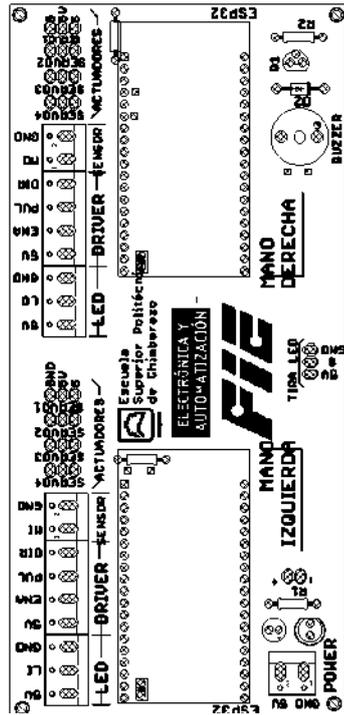
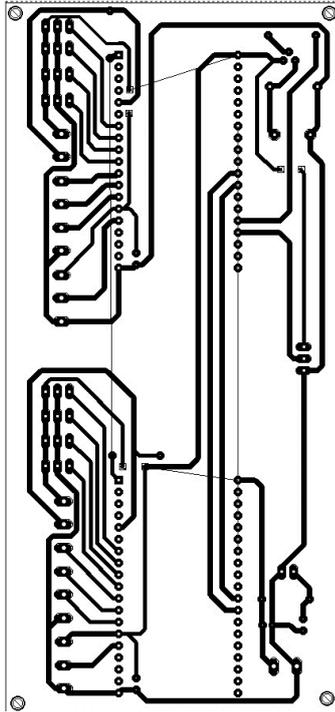
Contents

| | |
|-------------------------------------|----|
| 1 Overview | 1 |
| 2 Pin Definitions | 3 |
| 2.1 Pin Layout | 3 |
| 2.2 Pin Description | 3 |
| 2.3 Strapping Pins | 3 |
| 3 Functional Description | 6 |
| 3.1 CPU and Internal Memory | 6 |
| 3.2 External Flash and SPIRAM | 6 |
| 3.3 Crystal Oscillators | 6 |
| 3.4 RTC and Low-Power Management | 7 |
| 4 Peripherals and Sensors | 8 |
| 5 Electrical Characteristics | 9 |
| 5.1 Absolute Maximum Ratings | 9 |
| 5.2 WiFi Radio | 9 |
| 5.3 BLE Radio | 10 |
| 5.3.1 Receiver | 10 |
| 5.3.2 Transmitter | 10 |
| 5.4 Thermal Profile | 11 |
| 6 Schematics | 12 |
| 7 Peripheral Schematics | 13 |
| 8 Dimensions | 15 |
| 9 Learning Resources | 16 |
| 9.1 Must-Read Documents | 16 |
| 9.2 Must-Have Resources | 16 |
| Revision History | 17 |

List of Tables

| | | |
|---|---|----|
| 1 | ESP32-WROOM-32(ESP-WROOM-32) Specifications | 1 |
| 2 | Pin Definitions | 3 |
| 3 | Strapping Pins | 3 |
| 4 | Power Consumption by Power Modes | 7 |
| 5 | Absolute Maximum Ratings | 9 |
| 6 | WiFi Radio Characteristics | 9 |
| 7 | Receiver Characteristics — BLE | 10 |
| 8 | Transmitter Characteristics — BLE | 10 |

ANEXO H: ESQUEMAS PCB




```

derecha_wtfPro Ino
825
826 void der_dedo2() {
827   pixels.setPixelColor(2, pixels.Color(0, 90, 0));
828   pixels.show();
829   //Serial.println("TIEMPO: " + String(tempo));
830   servo2.write(v1);
831   //Serial.println("2 en 70");
832   delay(tempo);
833   servo2.write(vh);
834   //Serial.println("2 en 120");
835   delay(tempo2);
836   pixels.setPixelColor(2, pixels.Color(0, 0, 0));
837   pixels.show();
838 }
839
840 void der_dedo3() {
841   pixels.setPixelColor(1, pixels.Color(0, 0, 50));
842   pixels.show();
843   //Serial.println("TIEMPO: " + String(tempo));
844   servo2.write(v1);
845   //Serial.println("3 en 70");
846   delay(tempo);
847   servo2.write(vh);
848   //Serial.println("3 en 120");
849   delay(tempo2);
850   pixels.setPixelColor(1, pixels.Color(0, 0, 0));
851   pixels.show();
852 }
853
854 void der_dedo4() {
855   pixels.setPixelColor(0, pixels.Color(50, 50, 50));
856   pixels.show();
857   servo4.write(v1);
858   //Serial.println("4 en 60");
859   delay(tempo);
860   servo4.write(vh);
861   //Serial.println("4 en 120");
862   delay(tempo2);
863   pixels.setPixelColor(0, pixels.Color(0, 0, 0));
864   pixels.show();
865 }
866
867 void abajo() {
868   servo1.attach(servoPin);
869   servo1.write(v1);
870   servo2.attach(servoPin2);
871   servo2.write(v1);

```

```

derecha_wtfPro Ino
872   servo3.attach(servoPin3);
873   servo3.write(v1);
874   servo4.attach(servoPin4);
875   servo4.write(v1);
876   Serial.println("ABAJO");
877 }
878 void medio() {
879   servo1.attach(servoPin);
880   servo1.write(vh);
881   servo2.attach(servoPin2);
882   servo2.write(vh);
883   servo3.attach(servoPin3);
884   servo3.write(vh);
885   servo4.attach(servoPin4);
886   servo4.write(vh);
887   Serial.println("MEDIO");
888 }
889
890 void mover1() {
891   Serial.println("M1");
892   der_dedo1();
893   der_dedo2();
894   der_dedo3();
895   der_dedo4();
896   Firebase.setString(fbdo, "DATOS/SECUENCIA", "0");
897 }
898
899 void mover2() {
900   Serial.println("M2");
901   der_dedo1();
902   der_dedo2();
903   der_dedo3();
904   der_dedo4();
905   Firebase.setString(fbdo, "DATOS/SECUENCIA", "0");
906 }
907
908 void mover3() {
909   Serial.println("M3");
910   der_dedo1();
911   der_dedo2();
912   der_dedo3();
913   der_dedo4();
914   Firebase.setString(fbdo, "DATOS/SECUENCIA", "0");
915 }
916
917 void mover4() {
918   while (cancelon) {
919     Serial.println("ALEGRIA");

```

```

derecha_wtfPro Ino
920   tempo = 385;
921   tempo2 = 385;
922   delay(300);
923   stop();
924   if (cancelon == false) {
925     break;
926 }
927 //primer arreglo
928 buscar_nota("D5");
929 stop();
930 if (cancelon == false) {
931   break;
932 }
933 Serial2.println("Z1");
934 der_dedo2();
935 der_dedo3();
936 der_dedo4();
937 der_dedo1();
938 der_dedo2();
939 der_dedo3();
940 der_dedo4();
941 stop();
942 if (cancelon == false) {
943   break;
944 }
945 servo1.write(v1);
946 //Serial.println("1 en 70");
947 delay(tempo);
948 stop();
949 if (cancelon == false) {
950   break;
951 }
952 servo1.write(vh);
953 //Serial.println("1 en 120");
954 delay(380);
955 buscar_nota("C5");
956 Serial2.println("Z2");
957 stop();
958 if (cancelon == false) {
959   break;
960 }
961 der_dedo1();
962 der_dedo2();
963 der_dedo3();
964 der_dedo4();
965 stop();

```

```

derecha_wtfPro Ino
966   stop();
967   if (cancelon == false) {
968     break;
969 }
970 tempo = 380;
971 der_dedo1();
972 der_dedo2();
973 tempo = 400;
974 Serial2.println("Z3");
975 der_dedo1();
976 delay(180);
977 stop();
978 if (cancelon == false) {
979   break;
980 }
981 tempo = 385;
982 tempo2 = 385;
983 //segundo arreglo
984 buscar_nota("E5");
985 Serial2.println("Z4");
986 stop();
987 if (cancelon == false) {
988   break;
989 }
990 der_dedo2();
991 der_dedo3();
992 der_dedo4();
993 der_dedo1();
994 der_dedo2();
995 der_dedo3();
996 der_dedo4();
997 stop();
998 if (cancelon == false) {
999   break;
1000 }
1001 //der_dedo1();
1002 servo1.write(v1);
1003 //Serial.println("1 en 70");
1004 delay(tempo);
1005 servo1.write(vh);
1006 //Serial.println("1 en 120");
1007 stop();
1008 if (cancelon == false) {
1009   break;
1010 }

```

```

derecha_vifProino
1800   break;
1801   }
1802   delay(100);
1803   //mover linea1
1804   buscar_nota("C5");
1805   Serial2.println("Z5");
1806   stop();
1807   der_dedo1();
1808   der_dedo2();
1809   der_dedo3();
1810   stop();
1811   if (cancelon == false) {
1812     break;
1813   }
1814   tempo = 400;
1815   der_dedo1();
1816   stop();
1817   if (cancelon == false) {
1818     break;
1819   }
1820   delay(50);
1821   tempo = 150;
1822   der_dedo1();
1823   stop();
1824   if (cancelon == false) {
1825     break;
1826   }
1827   Serial2.println("Z0");
1828   der_dedo1();
1829   delay(200);
1830   stop();
1831   if (cancelon == false) {
1832     break;
1833   }
1834   //buscar nota("C5");
1835   //delay(500);
1836   //mover arreglo-----
1837   tempo = 185;
1838   tempo2 = 185;
1839   Serial2.println("Z7");
1840   der_dedo2();
1841   stop();
1842   if (cancelon == false) {
1843     break;
1844   }
1845   //buscar_nota("C5");
1846   //delay(500);
1847   //mover arreglo-----
1848   tempo = 185;
1849   tempo2 = 185;
1850   Serial2.println("Z7");
1851   der_dedo2();
1852   stop();
1853   if (cancelon == false) {
1854     break;
1855   }
1856

```

```

derecha_vifProino
1856   delay(100);
1857   der_dedo1();
1858   Serial2.println("Z8");
1859   der_dedo1();
1860   delay(100);
1861   stop();
1862   if (cancelon == false) {
1863     break;
1864   }
1865   //-----
1866   der_dedo2();
1867   delay(100);
1868   tempo = 130;
1869   tempo2 = 130;
1870   der_dedo3();
1871   stop();
1872   if (cancelon == false) {
1873     break;
1874   }
1875   der_dedo4();
1876   der_dedo3();
1877   stop();
1878   if (cancelon == false) {
1879     break;
1880   }
1881   delay(200);
1882   Serial2.println("Z7");
1883   der_dedo1();
1884   stop();
1885   if (cancelon == false) {
1886     break;
1887   }
1888   delay(50);
1889   //-----
1890   tempo = 185;
1891   tempo2 = 185;
1892   der_dedo2();
1893   //Serial2.println("Z10");
1894   delay(100);
1895   tempo = 130;
1896   tempo2 = 130;
1897   der_dedo3();
1898   der_dedo4();
1899   stop();
1900   if (cancelon == false) {
1901     break;
1902   }
1903

```

```

derecha_vifProino
1901   }
1902   }
1903   der_dedo3();
1904   delay(100);
1905   tempo = 100;
1906   Serial2.println("Z6");
1907   der_dedo1();
1908   stop();
1909   if (cancelon == false) {
1910     break;
1911   }
1912   delay(100);
1913   der_dedo1();
1914   tempo = 400;
1915   der_dedo2();
1916   der_dedo3();
1917   tempo = 500;
1918   Serial2.println("Z9");
1919   der_dedo1();
1920   stop();
1921   if (cancelon == false) {
1922     break;
1923   }
1924   delay(100);
1925   //teclad();
1926   //-----
1927   //mover linea1
1928   buscar_nota("B5");
1929   //delay(100);
1930   tempo = 185;
1931   tempo2 = 185;
1932   Serial2.println("Z4");
1933   stop();
1934   if (cancelon == false) {
1935     break;
1936   }
1937   der_dedo2();
1938   der_dedo3();
1939   der_dedo4();
1940   stop();
1941   if (cancelon == false) {
1942     break;
1943   }
1944   der_dedo4();
1945   der_dedo3();
1946   der_dedo2();
1947

```

```

derecha_vifProino
1946   }
1947   }
1948   der_dedo4();
1949   der_dedo2();
1950   //der_dedo1();
1951   //der_dedo3();
1952   serial.write(v);
1953   //Serial.println("i en 70");
1954   stop();
1955   if (cancelon == false) {
1956     break;
1957   }
1958   delay(tempo);
1959   serial.write(vh);
1960   //Serial.println("i an 120");
1961   delay(130);
1962   stop();
1963   if (cancelon == false) {
1964     break;
1965   }
1966   //mover linea1
1967   //mover linea1
1968   buscar_nota("C5");
1969   Serial2.println("Z5");
1970   stop();
1971   der_dedo1();
1972   der_dedo1();
1973   der_dedo2();
1974   stop();
1975   if (cancelon == false) {
1976     break;
1977   }
1978   delay(100);
1979   tempo = 135;
1980   der_dedo1();
1981   delay(100);
1982   tempo = 100;
1983   Serial2.println("Z10");
1984   stop();
1985   if (cancelon == false) {
1986     break;
1987   }
1988   der_dedo1();
1989   stop();
1990   if (cancelon == false) {
1991     break;
1992   }
1993

```



```

derecha_wiPro.ino
1559   if (cancel == false) {
1560     break;
1561   }
1562   tempo = 235;
1563   der_dedo2();
1564   delay(300);
1565
1566   tempo = 300;
1567   der_dedo2();
1568   delay(220);
1569   tempo = 335;
1570   start();
1571   if (cancel == false) {
1572     break;
1573   }
1574   der_dedo1();
1575   tempo = 450;
1576   Serial2.println("V13");
1577   stop();
1578   if (cancel == false) {
1579     break;
1580   }
1581   der_dedo1();
1582
1583   tempo = 515;
1584   //delay(100);
1585   tempo = 335;
1586   tempo2 = 60;
1587   buscar_nota("E5");
1588   stop();
1589   if (cancel == false) {
1590     break;
1591   }
1592   der_dedo4();
1593   tempo2 = 90;
1594   der_dedo3();
1595   tempo2 = 100;
1596   buscar_nota("F5");
1597   stop();
1598   if (cancel == false) {
1599     break;
1600   }
1601   tempo = 300;
1602   der_dedo3();
1603   delay(200);
1604
1605   tempo = 330;

```

```

derecha_wiPro.ino
1606   tempo2 = 125;
1607
1608   der_dedo3();
1609   Serial2.println("V11");
1610   stop();
1611   if (cancel == false) {
1612     break;
1613   }
1614   der_dedo3();
1615   tempo = 100;
1616   delay(10);
1617   der_dedo3();
1618   der_dedo2();
1619   der_dedo3();
1620   stop();
1621   if (cancel == false) {
1622     break;
1623   }
1624   delay(100);
1625   der_dedo1();
1626   der_dedo2();
1627   der_dedo3();
1628   delay(100);
1629   der_dedo1();
1630   stop();
1631   if (cancel == false) {
1632     break;
1633   }
1634   der_dedo2();
1635   der_dedo3();
1636   delay(100);
1637   der_dedo1();
1638   der_dedo2();
1639   Serial2.println("V11");
1640   stop();
1641   if (cancel == false) {
1642     break;
1643   }
1644   der_dedo3();
1645   stop();
1646   if (cancel == false) {
1647     break;
1648   }
1649   delay(600);
1650   start();
1651   if (cancel == false) {
1652     break;

```

```

derecha_wiPro.ino
1653   if (cancel == false) {
1654     break;
1655   }
1656   //SERIAL 2
1657   buscar_nota("E5");
1658   stop();
1659   if (cancel == false) {
1660     break;
1661   }
1662   tempo = 250;
1663   tempo2 = 220;
1664   der_dedo1();
1665   tempo = 250;
1666   Serial2.println("V4");
1667   stop();
1668   if (cancel == false) {
1669     break;
1670   }
1671   der_dedo1();
1672   buscar_nota("D5");
1673   stop();
1674   if (cancel == false) {
1675     break;
1676   }
1677   tempo = 300;
1678   tempo2 = 150;
1679   der_dedo1();
1680   der_dedo2();
1681   der_dedo2();
1682   der_dedo1();
1683   buscar_nota("E5");
1684   stop();
1685   if (cancel == false) {
1686     break;
1687   }
1688   der_dedo1();
1689   tempo = 200;
1690   tempo2 = 150;
1691   buscar_nota("E5");
1692   der_dedo1();
1693   delay(110);
1694   Serial2.println("V5");
1695   stop();
1696   if (cancel == false) {
1697     break;
1698   }
1699   der_dedo1();
1700

```

```

derecha_wiPro.ino
1701   }
1702   der_dedo2();
1703   delay(100);
1704   der_dedo1();
1705   delay(100);
1706   der_dedo2();
1707   tempo = 300;
1708   der_dedo3();
1709   delay(300);
1710   stop();
1711   if (cancel == false) {
1712     break;
1713   }
1714   tempo = 120;
1715   tempo2 = 60;
1716   der_dedo2();
1717   der_dedo3();
1718   tempo = 230;
1719   der_dedo2();
1720   tempo = 100;
1721   stop();
1722   if (cancel == false) {
1723     break;
1724   }
1725   der_dedo1();
1726   tempo = 250;
1727   der_dedo2();
1728   tempo = 210;
1729   tempo2 = 110;
1730   der_dedo1();
1731   tempo = 410;
1732   stop();
1733   if (cancel == false) {
1734     break;
1735   }
1736   Serial2.println("V8");
1737   der_dedo1();
1738   stop();
1739   if (cancel == false) {
1740     break;
1741   }
1742   delay(300);
1743   buscar_nota("C5");
1744   tempo = 230;
1745   tempo2 = 220;
1746   der_dedo1();

```

```

derecha_wifiPrimo
1743   tiempo = 250;
1744   stop();
1745   if (cancelon == false) {
1746     break;
1747   }
1748   Serial2.println("V4");
1749   der_dedo1();
1750   buscar_nota("DS");
1751   tiempo = 380;
1752   tempo2 = 150;
1753   der_dedo4();
1754   der_dedo1();
1755   der_dedo1();
1756   der_dedo1();
1757   buscar_nota("CS");
1758   stop();
1759   if (cancelon == false) {
1760     break;
1761   }
1762   der_dedo1();
1763   tiempo = 200;
1764   tempo2 = 150;
1765   buscar_nota("CS");
1766   stop();
1767   if (cancelon == false) {
1768     break;
1769   }
1770   der_dedo1();
1771   delay(110);
1772   Serial2.println("V5");
1773   stop();
1774   if (cancelon == false) {
1775     break;
1776   }
1777   der_dedo1();
1778   delay(180);
1779   der_dedo1();
1780   delay(180);
1781   der_dedo1();
1782   tiempo = 300;
1783   der_dedo1();
1784   delay(380);
1785   stop();
1786   if (cancelon == false) {
1787     break;
1788   }
1789   tiempo = 120;

```

```

derecha_wifiPrimo
1788   }
1789   tiempo = 120;
1790   tempo2 = 60;
1791   der_dedo1();
1792   der_dedo1();
1793   tiempo = 330;
1794   der_dedo4();
1795   tiempo = 100;
1796   der_dedo1();
1797   delay(280);
1798   stop();
1799   if (cancelon == false) {
1800     break;
1801   }
1802   tiempo = 250;
1803   der_dedo1();
1804   tiempo = 230;
1805   tempo2 = 110;
1806   der_dedo1();
1807   tiempo = 410;
1808   stop();
1809   if (cancelon == false) {
1810     break;
1811   }
1812   Serial2.println("V6");
1813   der_dedo1();
1814   Serial2.println("V6");
1815   Serial2.println("V6");
1816   stop();
1817   if (cancelon == false) {
1818     break;
1819   }
1820   Serial2.println("V6");
1821   stop();
1822   if (cancelon == false) {
1823     break;
1824   }
1825   delay(280);
1826   delay(480);
1827   stop();
1828   if (cancelon == false) {
1829     break;
1830   }
1831   tiempo = 210;
1832   buscar_nota("AA");
1833   stop();
1834   if (cancelon == false) {

```

```

derecha_wifiPrimo
1835     break;
1836   }
1837   delay(600);
1838   tiempo = 220;
1839   tempo2 = 180;
1840   der_dedo1();
1841   stop();
1842   if (cancelon == false) {
1843     break;
1844   }
1845   der_dedo1();
1846   tiempo = 220;
1847   tempo2 = 180;
1848   der_dedo1();
1849   stop();
1850   if (cancelon == false) {
1851     break;
1852   }
1853   buscar_nota("DS");
1854   Serial2.println("V10");
1855   stop();
1856   if (cancelon == false) {
1857     break;
1858   }
1859   der_dedo1();
1860   delay(280);
1861   stop();
1862   if (cancelon == false) {
1863     break;
1864   }
1865   tiempo = 230;
1866   tempo2 = 190;
1867   der_dedo1();
1868   der_dedo1();
1869   Serial2.println("V11");
1870   stop();
1871   if (cancelon == false) {
1872     break;
1873   }
1874   der_dedo1();
1875   buscar_nota("CS");
1876   der_dedo1();
1877   tiempo = 230;
1878   tempo2 = 190;
1879   buscar_nota("ES");
1880   stop();
1881   if (cancelon == false) {

```

```

derecha_wifiPrimo
1880   stop();
1881   if (cancelon == false) {
1882     break;
1883   }
1884   der_dedo1();
1885   tiempo = 320;
1886   tempo2 = 200;
1887   der_dedo1();
1888   tiempo = 220;
1889   tempo2 = 200;
1890   buscar_nota("CS");
1891   stop();
1892   if (cancelon == false) {
1893     break;
1894   }
1895   Serial2.println("V12");
1896   der_dedo1();
1897   }
1898   tiempo = 220;
1899   tempo2 = 200;
1900   delay(280);
1901   der_dedo1();
1902   buscar_nota("DS");
1903   stop();
1904   if (cancelon == false) {
1905     break;
1906   }
1907   Serial2.println("V13");
1908   der_dedo1();
1909   buscar_nota("CS");
1910   tiempo = 190;
1911   tempo2 = 190;
1912   der_dedo1();
1913   der_dedo1();
1914   delay(280);
1915   Serial2.println("V14");
1916   stop();
1917   if (cancelon == false) {
1918     break;
1919   }
1920   der_dedo1();
1921   der_dedo1();
1922   der_dedo1();
1923   buscar_nota("ES");
1924   der_dedo1();
1925   delay(280);
1926

```

```

derecha_wifiPro.ino
1927  der_dedo1();
1928  tempo = 420;
1929  tempo2 = 200;
1930  buscar_nota("D5");
1931  stop();
1932  if (cancelon == false) {
1933    break;
1934  }
1935  Serial2.println("V15");
1936  der_dedo1();
1937  tempo = 220;
1938  tempo2 = 200;
1939  stop();
1940  if (cancelon == false) {
1941    break;
1942  }
1943  buscar_nota("E5");
1944  der_dedo1();
1945  tempo = 320;
1946  tempo2 = 200;
1947  der_dedo1();
1948  delay(300);
1949  tempo = 220;
1950  tempo2 = 200;
1951  stop();
1952  if (cancelon == false) {
1953    break;
1954  }
1955  der_dedo1();
1956  tempo = 320;
1957  tempo2 = 200;
1958  der_dedo1();
1959  buscar_nota("D5");
1960  tempo = 220;
1961  tempo2 = 200;
1962  der_dedo1();
1963  tempo = 240;
1964  tempo2 = 200;
1965  Serial2.println("V16");
1966  stop();
1967  if (cancelon == false) {
1968    break;
1969  }
1970  der_dedo1();
1971  buscar_nota("C5");
1972  stop();

```

```

derecha_wifiPro.ino
1972  buscar_nota("C5");
1973  stop();
1974  if (cancelon == false) {
1975    break;
1976  }
1977  tempo = 120;
1978  tempo2 = 110;
1979  der_dedo1();
1980  der_dedo2();
1981  tempo = 200;
1982  tempo2 = 110;
1983  stop();
1984  if (cancelon == false) {
1985    break;
1986  }
1987  der_dedo1();
1988  delay(150);
1989  der_dedo1();
1990  delay(150);
1991  der_dedo2();
1992  tempo = 200;
1993  tempo2 = 200;
1994  stop();
1995  if (cancelon == false) {
1996    break;
1997  }
1998  der_dedo1();
1999  tempo = 400;
2000  tempo2 = 200;
2001  Serial2.println("V17");
2002  der_dedo1();
2003  stop();
2004  if (cancelon == false) {
2005    break;
2006  }
2007  buscar_nota("E5");
2008  stop();
2009  if (cancelon == false) {
2010    break;
2011  }
2012  tempo = 220;
2013  tempo2 = 200;
2014  der_dedo1();
2015  tempo = 420;
2016  tempo2 = 200;
2017  der_dedo1();
2018  buscar_nota("D5");

```

```

derecha_wifiPro.ino
2017  der_dedo1();
2018  buscar_nota("D5");
2019  stop();
2020  if (cancelon == false) {
2021    break;
2022  }
2023  tempo = 220;
2024  tempo2 = 200;
2025  der_dedo1();
2026  tempo = 650;
2027  tempo2 = 200;
2028  stop();
2029  if (cancelon == false) {
2030    break;
2031  }
2032  der_dedo1();
2033  delay(200);
2034  tempo = 620;
2035  tempo2 = 200;
2036  der_dedo2();
2037  //Serial2.println("V17");
2038  delay(200);
2039  Serial2.println("V18");
2040  der_dedo1();
2041  tempo = 115;
2042  tempo = 115;
2043  tempo = 115;
2044  tempo2 = 0;
2045  buscar_nota("E5");
2046  stop();
2047  if (cancelon == false) {
2048    break;
2049  }
2050  der_dedo1();
2051  tempo2 = 90;
2052  der_dedo3();
2053  tempo2 = 180;
2054  buscar_nota("C5");
2055  stop();
2056  if (cancelon == false) {
2057    break;
2058  }
2059  tempo = 180;
2060  tempo2 = 180;
2061  der_dedo1();
2062  delay(400);
2063

```

```

derecha_wifiPro.ino
2064  tempo = 300;
2065  tempo2 = 220;
2066  der_dedo1();
2067  tempo2 = 300;
2068  Serial2.println("V9");
2069  stop();
2070  if (cancelon == false) {
2071    break;
2072  }
2073  der_dedo1();
2074  //delay(150);
2075  tempo = 115;
2076  tempo2 = 60;
2077  buscar_nota("E5");
2078  stop();
2079  if (cancelon == false) {
2080    break;
2081  }
2082  der_dedo1();
2083  tempo2 = 90;
2084  der_dedo3();
2085  buscar_nota("C5");
2086  //der_dedo2();
2087  Serial2.println("V9");
2088  stop();
2089  if (cancelon == false) {
2090    break;
2091  }
2092  tempo = 215;
2093  der_dedo2();
2094  delay(180);
2095  tempo = 160;
2096  stop();
2097  if (cancelon == false) {
2098    break;
2099  }
2100  der_dedo2();
2101  delay(220);
2102  tempo = 315;
2103  der_dedo1();
2104  tempo = 450;
2105  Serial2.println("V11");
2106  stop();
2107  if (cancelon == false) {
2108    break;
2109  }
2110

```

```

derecha_vf@Pro:ino
2109     break;
2110     }
2111     der_dedo1();
2112
2113     tempo = 115;
2114     //delay(100);
2115     tempo = 115;
2116     tempo2 = 90;
2117     buscar_nota("E5");
2118     stop();
2119     if (cancion == false) {
2120         break;
2121     }
2122     der_dedo4();
2123     tempo2 = 90;
2124     der_dedo1();
2125     tempo2 = 100;
2126     buscar_nota("C5");
2127     stop();
2128     if (cancion == false) {
2129         break;
2130     }
2131     tempo = 300;
2132     der_dedo3();
2133     delay(200);
2134
2135     tempo = 220;
2136     tempo2 = 125;
2137
2138     der_dedo3();
2139     Serial2.println("V111");
2140     stop();
2141     if (cancion == false) {
2142         break;
2143     }
2144     der_dedo3();
2145     tempo = 105;
2146     delay(10);
2147     der_dedo1();
2148     der_dedo2();
2149     der_dedo3();
2150     delay(100);
2151     stop();
2152     if (cancion == false) {
2153         break;
2154     }
2155     der_dedo1();

```

```

derecha_vf@Pro:ino
2156     der_dedo1();
2157     der_dedo1();
2158     delay(100);
2159     der_dedo1();
2160     der_dedo2();
2161     stop();
2162     if (cancion == false) {
2163         break;
2164     }
2165     der_dedo1();
2166     delay(100);
2167     der_dedo1();
2168     der_dedo2();
2169     Serial2.println("V1");
2170     der_dedo1();
2171     stop();
2172     if (cancion == false) {
2173         break;
2174     }
2175     Serial2.println("V7");
2176     apagarLuces();
2177
2178     delay(100);
2179     HW();
2180     break;
2181     }
2182     Serial2.println("V7");
2183     mostrarLuces();
2184     delay(100);
2185     HW();
2186     }
2187
2188     void shell() {
2189         Serial2.println("V111");
2190         der_dedo1();
2191         tempo = 115;
2192         tempo = 115;
2193         tempo2 = 60;
2194         buscar_nota("E5");
2195         der_dedo4();
2196         tempo2 = 90;
2197         der_dedo1();
2198         tempo2 = 100;
2199         buscar_nota("C5");
2200         tempo = 100;
2201         tempo2 = 100;
2202         der_dedo1();

```

```

derecha_vf@Pro:ino
2201     tempo2 = 100;
2202     der_dedo1();
2203     delay(400);
2204
2205     tempo = 300;
2206     tempo2 = 120;
2207     der_dedo1();
2208     tempo2 = 100;
2209     Serial2.println("V2");
2210     der_dedo1();
2211     //delay(100);
2212     tempo = 115;
2213     tempo2 = 90;
2214     buscar_nota("E5");
2215     der_dedo1();
2216     tempo2 = 90;
2217     der_dedo3();
2218     buscar_nota("C5");
2219     //der_dedo3();
2220     Serial2.println("V3");
2221     tempo = 215;
2222     der_dedo2();
2223     delay(300);
2224
2225     tempo = 160;
2226     der_dedo2();
2227     delay(200);
2228     tempo = 115;
2229     der_dedo1();
2230     tempo = 450;
2231     Serial2.println("V111");
2232     der_dedo1();
2233
2234     tempo = 115;
2235     //delay(100);
2236     tempo = 115;
2237     tempo2 = 60;
2238     buscar_nota("E5");
2239     der_dedo4();
2240     tempo2 = 90;
2241     der_dedo3();
2242     tempo2 = 100;
2243     buscar_nota("C5");
2244     tempo = 300;
2245     der_dedo3();
2246     delay(200);
2247

```

```

derecha_vf@Pro:ino
2248     delay(200);
2249
2250     tempo = 220;
2251     tempo2 = 125;
2252     der_dedo3();
2253     Serial2.println("V111");
2254     der_dedo3();
2255     tempo = 105;
2256     delay(10);
2257     der_dedo1();
2258     der_dedo1();
2259     delay(100);
2260     der_dedo1();
2261     der_dedo2();
2262     der_dedo3();
2263     delay(100);
2264     der_dedo1();
2265     der_dedo2();
2266     der_dedo3();
2267     der_dedo1();
2268     der_dedo2();
2269     der_dedo2();
2270     Serial2.println("V1");
2271     der_dedo1();
2272     }
2273
2274     void shell() {
2275         Serial2.println("V11");
2276         //der_dedo1();
2277         tempo = 115;
2278         tempo = 115;
2279         tempo2 = 90;
2280         buscar_nota("E5");
2281         der_dedo4();
2282         tempo2 = 90;
2283         der_dedo3();
2284         tempo2 = 100;
2285         buscar_nota("C5");
2286         tempo = 100;
2287         tempo2 = 100;
2288         der_dedo1();
2289         delay(400);
2290
2291         tempo = 300;
2292         tempo2 = 120;

```

```

derecha_wifiPro.ino
2293 der_dedo1();
2294 tempo2 = 180;
2295 Serial.println("V2");
2296 der_dedo1();
2297 //delay(150);
2298 tempo = 115;
2299 tempo2 = 60;
2300 buscar_nota("E5");
2301 der_dedo4();
2302 tempo2 = 90;
2303 der_dedo3();
2304 buscar_nota("C5");
2305 //der_dedo3();
2306 Serial.println("V3");
2307 tempo = 215;
2308 der_dedo2();
2309 delay(300);
2310
2311 tempo = 100;
2312 der_dedo3();
2313 delay(220);
2314 tempo = 315;
2315 der_dedo1();
2316 tempo = 150;
2317 Serial.println("V11");
2318 der_dedo1();
2319
2320 tempo = 115;
2321 //delay(180);
2322 tempo = 115;
2323 tempo2 = 60;
2324 buscar_nota("E5");
2325 der_dedo1();
2326 tempo2 = 90;
2327 der_dedo3();
2328 tempo2 = 180;
2329 buscar_nota("C5");
2330 tempo = 300;
2331 der_dedo3();
2332 delay(200);
2333
2334 tempo = 220;
2335 tempo2 = 125;
2336
2337 der_dedo3();
2338 Serial.println("V11");
2339 der_dedo3();

```

```

derecha_wifiPro.ino
2338 Serial.println("V11");
2339 der_dedo1();
2340 tempo = 185;
2341 delay(10);
2342 der_dedo1();
2343 der_dedo2();
2344 der_dedo1();
2345 delay(100);
2346 der_dedo1();
2347 der_dedo2();
2348 der_dedo3();
2349 delay(100);
2350 der_dedo1();
2351 der_dedo2();
2352 der_dedo3();
2353 delay(100);
2354 der_dedo1();
2355 der_dedo2();
2356 Serial.println("V11");
2357 der_dedo1();
2358
2359 void shd2() {
2360 buscar_nota("C5");
2361 tempo = 200;
2362 tempo2 = 220;
2363 der_dedo1();
2364 tempo = 250;
2365 Serial.println("V4");
2366 der_dedo1();
2367 buscar_nota("D5");
2368 tempo = 180;
2369 tempo2 = 150;
2370 der_dedo1();
2371 der_dedo3();
2372 der_dedo2();
2373 der_dedo1();
2374 buscar_nota("C5");
2375 der_dedo1();
2376 tempo = 200;
2377 tempo2 = 150;
2378 buscar_nota("C5");
2379 der_dedo1();
2380 delay(110);
2381 Serial.println("V5");
2382 der_dedo2();
2383 delay(100);
2384 der_dedo1();

```

```

derecha_wifiPro.ino
2385 delay(100);
2386 der_dedo2();
2387 tempo = 300;
2388 der_dedo3();
2389 delay(300);
2390
2391 tempo = 120;
2392 tempo2 = 60;
2393 der_dedo2();
2394 der_dedo3();
2395 tempo = 330;
2396 der_dedo4();
2397 tempo = 180;
2398 der_dedo3();
2399 delay(200);
2400 tempo = 250;
2401 der_dedo2();
2402 tempo = 210;
2403 tempo2 = 150;
2404 der_dedo1();
2405 tempo = 410;
2406 Serial.println("V6");
2407 der_dedo1();
2408
2409 void shd22() {
2410 buscar_nota("C5");
2411 tempo = 250;
2412 tempo2 = 220;
2413 der_dedo1();
2414 tempo = 250;
2415 Serial.println("V4");
2416 der_dedo1();
2417 buscar_nota("D5");
2418 tempo = 100;
2419 tempo2 = 150;
2420 der_dedo1();
2421 der_dedo3();
2422 der_dedo2();
2423 der_dedo1();
2424 buscar_nota("C5");
2425 der_dedo1();
2426 tempo = 200;
2427 tempo2 = 150;
2428 buscar_nota("C5");
2429 der_dedo2();
2430 delay(110);
2431 Serial.println("V5");

```

```

derecha_wifiPro.ino
2430 delay(110);
2431 Serial.println("V5");
2432 der_dedo2();
2433 delay(100);
2434 der_dedo1();
2435 delay(100);
2436 der_dedo2();
2437 tempo = 300;
2438 der_dedo3();
2439 delay(300);
2440
2441 tempo = 120;
2442 tempo2 = 60;
2443 der_dedo2();
2444 der_dedo3();
2445 tempo = 330;
2446 der_dedo4();
2447 tempo = 180;
2448 der_dedo3();
2449 delay(200);
2450 tempo = 250;
2451 der_dedo3();
2452 tempo = 210;
2453 tempo2 = 150;
2454 der_dedo1();
2455 tempo = 410;
2456 Serial.println("V6");
2457 der_dedo1();
2458 Serial.println("V7");
2459 Serial.println("V8");
2460 Serial.println("V9");
2461
2462
2463 void shd3() {
2464
2465 buscar_nota("A4");
2466 delay(600);
2467 tempo = 220;
2468 tempo2 = 180;
2469 der_dedo1();
2470 der_dedo3();
2471 tempo = 220;
2472 tempo2 = 180;
2473 der_dedo1();
2474 buscar_nota("D5");
2475 Serial.println("V10");
2476 der_dedo4();

```



```

MANO_ZQUERDA.ino
181 Serial.println("Inicio");
182 delay(1000);
183 Serial.println("Inicio");
184 delay(200);
185 habilitar2();
186 Serial.println("ACCION 3");
187 mover();
188 delay(100);
189 InputString = "";
190 obtener2();
191 }
192
193 if (InputString == "CAMBIO 4") {
194   habilitar2();
195   HC();
196   obtener2();
197   delay(100);
198   Serial.println("ACCION 4");
199   delay(200);
200   Serial.println("Inicio2");
201   mover2();
202   HC();
203   int tiempo = 100;
204   int tiempo2 = 100;
205   InputString = "";
206   obtener2();
207 }
208
209 if (InputString == "CAMBIO 5") {
210   habilitar2();
211   HC();
212   obtener2();
213   delay(100);
214   habilitar2();
215   delay(200);
216   mover2("A");
217   mover2("B");
218   mover2("C");
219   Serial.println("ACCION 5");
220   delay(100);
221   Serial.println("Inicio2");

```

```

MANO_ZQUERDA.ino
222
223 delay(100);
224 Serial.println("Inicio2");
225 mover();
226 HC();
227 int tiempo1 = 100;
228 int tiempo2 = 100;
229 InputString = "";
230 obtener2();
231 }
232
233 if (indice == "D") { //MOVER MOTOR Y PULSAM TECLA ESPECIFICA, ESTOY: DISEÑ
234   if (indice == 1) {
235     habilitar2();
236     buscar_rotas(rotas4);
237     obtener2();
238     InputString = "";
239   }
240 }
241
242 if (InputString == "E") { //MOVER MOTOR SEGUN POSICION ESPECIFICAS
243   Serial.println("MOVIMOS");
244   posos2 = InputString.substring(1, total2());
245   habilitar2();
246   mover();
247   obtener2();
248 }
249
250 if (InputString == "F") { //CAMBIAR TIEMPO DE PANTUFER SERVO.
251   tiempo = InputString.substring(1, total2());
252   Serial.println("Tiempo servo: " + String(tiempo));
253 }
254
255 if (InputString == "G") { //REINICIAR POSICION
256   Serial.println("moviendo a inicio");
257   p2 = false;
258   habilitar2();
259   HC();
260   mover2();
261   obtener2();
262 }
263
264 if (InputString == "H") { //MOVER SIGUIE PAGOS
265   posos2 = InputString.substring(1, total2());
266   Serial.println("P: ");
267   Serial.println(posos2);
268   Serial.println("1 - ");

```

```

MANO_ZQUERDA.ino
275 Serial.println(posos2);
276 Serial.println("1 - ");
277 Serial.println(tiempo2);
278 }
279
280 if (InputString == "I") { //CAMBIAR TIEMPO DE PASO
281   tiempo1 = InputString.substring(1, total2());
282   // Hacer algo con el angulo...
283   Serial.println("1");
284   Serial.println(posos2);
285   Serial.println("1 - ");
286   Serial.println(tiempo2);
287 }
288
289 if (InputString == "D") { // Si el primer caracter es "g" (grados)
290   getD();
291 }
292
293 if (InputString == "A") { // Si el primer caracter es "g" (grados)
294   getA();
295 }
296
297 if (InputString == "B") { // Si el primer caracter es "g" (grados)
298   getB();
299 }
300
301 if (InputString == "C") { // Si el primer caracter es "g" (grados)
302   getC();
303 }
304
305 if (InputString == "E") { // Si el primer caracter es "g" (grados)
306   getE();
307 }
308
309 }
310
311 void habilitar1() { //ACTIVAR MOTOR
312   delay(100);
313   digitalWrite(motor, HIGH);
314   delay(100);
315 }
316
317 void desarmar1() { //DESARMAR MOTOR
318   delay(100);
319   digitalWrite(motor, LOW);
320   delay(100);

```

```

MANO_ZQUERDA.ino
321
322 }
323
324 void init_servo1() { //MOTOR
325   pinMode(servo1Color(1), servo1.Color(150, 0, 0));
326   pinMode(servo1);
327   servo1.write(90);
328   delay(tiempo3);
329   servo1.write(120);
330   delay(tiempo4);
331   pinMode(servo1Color(2), servo1.Color(0, 0, 0));
332   pinMode(servo1);
333 }
334
335 void init_servo2() { //MOTOR
336   pinMode(servo2Color(2), servo2.Color(0, 50, 0));
337   pinMode(servo2);
338   servo2.write(90);
339   delay(tiempo3);
340   servo2.write(120);
341   delay(tiempo4);
342   pinMode(servo2Color(2), servo2.Color(0, 0, 0));
343   pinMode(servo2);
344 }
345
346 void init_servo3() { //MOTOR
347   pinMode(servo3Color(1), servo3.Color(0, 0, 50));
348   pinMode(servo3);
349   servo3.write(90);
350   delay(tiempo3);
351   servo3.write(120);
352   delay(tiempo4);
353   pinMode(servo3Color(1), servo3.Color(0, 0, 0));
354   pinMode(servo3);
355 }
356
357 void init_servo4() { //MOTOR
358   pinMode(servo4Color(0), servo4.Color(50, 50, 50));
359   pinMode(servo4);
360   servo4.write(90);
361   delay(tiempo3);
362   servo4.write(120);
363   delay(tiempo4);
364   pinMode(servo4Color(0), servo4.Color(0, 0, 0));
365   pinMode(servo4);
366 }
367
368 void inicio() { //BIBLIOTECAS
369   Serial.println("INICIO");

```

```

MANO_IZQUIERDA.ino
367 void inicio() { //DEBES ARRIBA
368   Serial.println("INICIO");
369   servo5.attach(servoPin5);
370   servo5.write(v12);
371   servo6.attach(servoPin6);
372   servo6.write(v12);
373   servo7.attach(servoPin7);
374   servo7.write(v12);
375   servo8.attach(servoPin8);
376   servo8.write(v12);
377 }
378
379 void abajo() { //DEBES ABAJO
380   servo5.attach(servoPin5);
381   servo5.write(v2);
382   servo6.attach(servoPin6);
383   servo6.write(v2);
384   servo7.attach(servoPin7);
385   servo7.write(v2);
386   servo8.attach(servoPin8);
387   servo8.write(v2);
388   Serial.println("ABAJO");
389 }
390
391 void medio() { //DEBES MEDIO
392   servo5.attach(servoPin5);
393   servo5.write(v2);
394   servo6.attach(servoPin6);
395   servo6.write(v2);
396   servo7.attach(servoPin7);
397   servo7.write(v2);
398   servo8.attach(servoPin8);
399   servo8.write(v2);
400   Serial.println("MEDIO");
401 }
402
403 void mover1() { //SECUENCIA DE SERVO 1
404   Serial.println("M1");
405   Iz_dedo1();
406   Iz_dedo2();
407   Iz_dedo3();
408 }
409
410 void mover2() { //SECUENCIA DE SERVO 2
411   Serial.println("M2");
412   Iz_dedo1();
413   Iz_dedo2();
414 }

```

```

MANO_IZQUIERDA.ino
414   Iz_dedo4();
415 }
416 void mover3() { //SECUENCIA DE SERVO 3
417   Serial.println("M3");
418   Iz_dedo1();
419   Iz_dedo4();
420   Iz_dedo3();
421   Iz_dedo2();
422 }
423
424 void mover4() { //CONJUNTO ALERERIA ACERBA/ARREBENTADO
425   Serial.println("lister");
426   delay(100);
427   Serial.println("lister");
428   Serial.println("lister");
429 }
430
431 while (1) {
432   if (Serial2.available() > 0) {
433     // Leer el mensaje
434     String mensaje = Serial2.readStringUntil('\n');
435     mensaje.trim();
436     if (mensaje == "Z1") {
437       servo5.write(v2); //dedo 1
438       servo7.write(v2); //dedo 2
439       LucasDoble1();
440       delay(2400);
441       servo5.write(v12);
442       servo7.write(v12);
443       apagarLuces();
444       delay(150);
445       buscar_notas("C4");
446     }
447     if (mensaje == "Z2") {
448       servo5.write(v2);
449       servo7.write(v2);
450       LucasDoble1();
451       delay(1600);
452       servo5.write(v12);
453       servo7.write(v12);
454       apagarLuces();
455       delay(150);
456       buscar_notas("D4");
457     }
458     if (mensaje == "Z3") {
459       servo5.write(v2);
460       servo7.write(v2);
461       LucasDoble1();
462     }

```

```

MANO_IZQUIERDA.ino
459   servo7.write(v2);
460   LucasDoble1();
461   delay(400);
462   apagarLuces();
463   servo5.write(v12);
464   servo7.write(v12);
465   delay(150);
466   //SERVIDO ARRIBA
467   buscar_notas("E4");
468 }
469 if (mensaje == "Z4") {
470   servo5.write(v2);
471   servo7.write(v2);
472   LucasDoble1();
473   delay(1400);
474   apagarLuces();
475   servo5.write(v12);
476   servo7.write(v12);
477   delay(150);
478   buscar_notas("C4");
479 }
480 if (mensaje == "Z5") {
481   servo5.write(v2);
482   servo7.write(v2);
483   LucasDoble1();
484   delay(1000);
485   apagarLuces();
486   servo5.write(v12);
487   servo7.write(v12);
488   delay(110);
489 }
490 if (mensaje == "Z6") {
491   servo5.write(v2);
492   servo7.write(v2);
493   LucasDoble1();
494   delay(100);
495   apagarLuces();
496   servo5.write(v12);
497   servo7.write(v12);
498   //funcion arreglo
499   tespol = 185;
500   tespol = 185;
501   delay(150);
502   buscar_notas("D4");
503 }
504 }
505 }

```

```

MANO_IZQUIERDA.ino
506 if (mensaje == "Z7") {
507   servo5.write(v2);
508   servo7.write(v2);
509   LucasDoble1();
510   delay(400);
511   apagarLuces();
512   servo5.write(v12);
513   servo7.write(v12);
514   //funcion arreglo
515 }
516 if (mensaje == "Z8") {
517   servo5.write(v2);
518   servo7.write(v2);
519   LucasDoble1();
520   delay(1800);
521   apagarLuces();
522   servo5.write(v12);
523   servo7.write(v12);
524 }
525 if (mensaje == "Z9") {
526   servo5.write(v2);
527   servo7.write(v2);
528   LucasDoble1();
529   delay(1200);
530   apagarLuces();
531   servo5.write(v12);
532   servo7.write(v12);
533   //funcion arreglo
534 }
535 if (mensaje == "Z8") {
536   servo5.write(v2);
537   servo7.write(v2);
538   LucasDoble1();
539   delay(1300);
540   apagarLuces();
541   servo5.write(v12);
542   servo7.write(v12);
543 }
544 if (mensaje == "Z9") {
545   servo5.write(v2);
546   servo7.write(v2);
547   LucasDoble1();
548   delay(600);
549   apagarLuces();
550   servo5.write(v12);
551   servo7.write(v12);
552   delay(130);

```



```

MANO_IZQUIERDA.ino
1830 }
1831 iz_dedo2();
1832 iz_dedo1();
1833 Serial2.println("C7");
1834 stop();
1835 if (inputString == "STOP") {
1836   apagarLuces();
1837   inicio();
1838   fincancion();
1839 }
1840 break;
1841 }
1842 iz_dedo2();
1843 iz_dedo1();
1844 Serial2.println("C8");
1845 stop();
1846 if (inputString == "STOP") {
1847   apagarLuces();
1848   inicio();
1849   fincancion();
1850 }
1851 break;
1852 }
1853 iz_dedo2();
1854 iz_dedo1();
1855 Serial2.println("C9");
1856 stop();
1857 if (inputString == "STOP") {
1858   apagarLuces();
1859   inicio();
1860   fincancion();
1861 }
1862 break;
1863 }
1864 iz_dedo2();
1865 iz_dedo1();
1866 Serial2.println("C10");
1867 stop();
1868 if (inputString == "STOP") {
1869   apagarLuces();
1870   inicio();
1871   fincancion();
1872 }
1873 break;
1874 }
1875 }
1876 }

```

```

MANO_IZQUIERDA.ino
1875 fincancion();
1876 break;
1877 }
1878 iz_dedo2();
1879 iz_dedo1();
1880 iz_dedo2();
1881 iz_dedo1();
1882 stop();
1883 if (inputString == "STOP") {
1884   apagarLuces();
1885   inicio();
1886   fincancion();
1887 }
1888 break;
1889 }
1890 iz_dedo2();
1891 iz_dedo1();
1892 stop();
1893 if (inputString == "STOP") {
1894   apagarLuces();
1895   inicio();
1896   fincancion();
1897 }
1898 break;
1899 }
1900 }
1901 }
1902 }
1903 }
1904 }
1905 }
1906 }
1907 }
1908 }
1909 }
1910 }
1911 }
1912 }
1913 }
1914 }
1915 }
1916 }
1917 }
1918 }
1919 }
1920 }
1921 }

```

```

MANO_IZQUIERDA.ino
1922 apagarLuces();
1923 inicio();
1924 fincancion();
1925 }
1926 break;
1927 }
1928 }
1929 }
1930 }
1931 }
1932 }
1933 }
1934 }
1935 }
1936 }
1937 }
1938 }
1939 }
1940 }
1941 }
1942 }
1943 }
1944 }
1945 }
1946 }
1947 }
1948 }
1949 }
1950 }
1951 }
1952 }
1953 }
1954 }
1955 }
1956 }
1957 }
1958 }
1959 }
1960 }
1961 }
1962 }
1963 }
1964 }
1965 }
1966 }
1967 }
1968 }

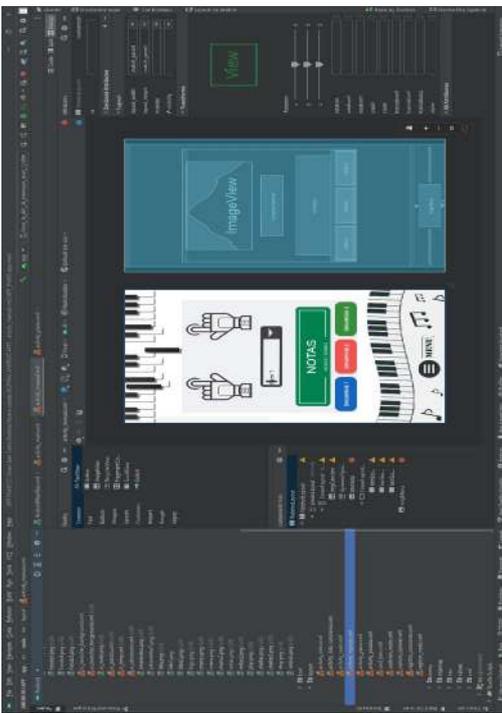
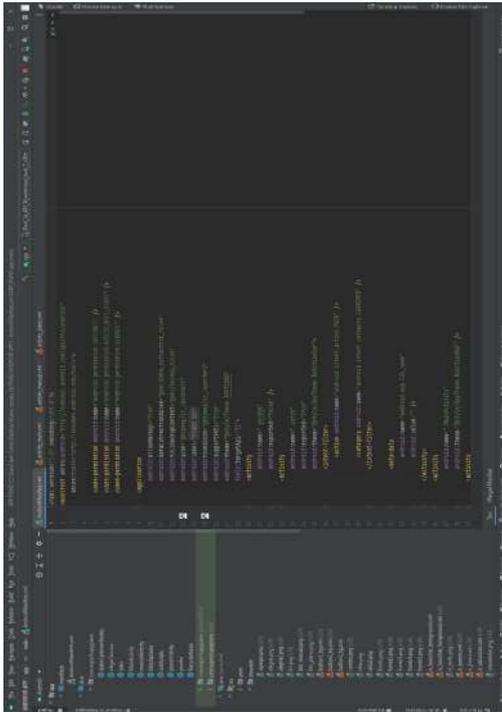
```

```

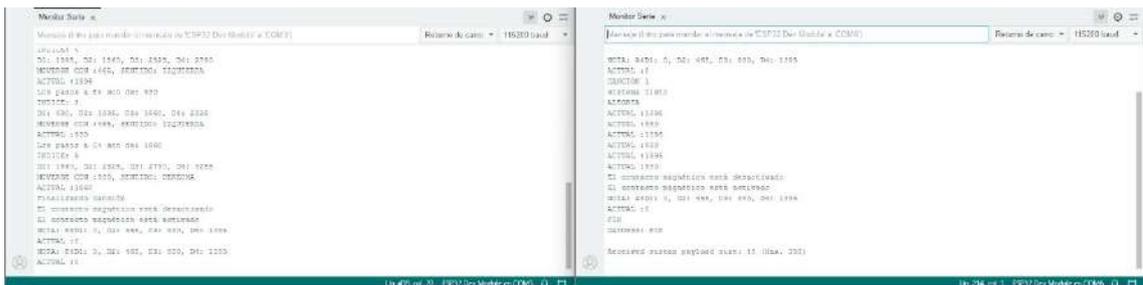
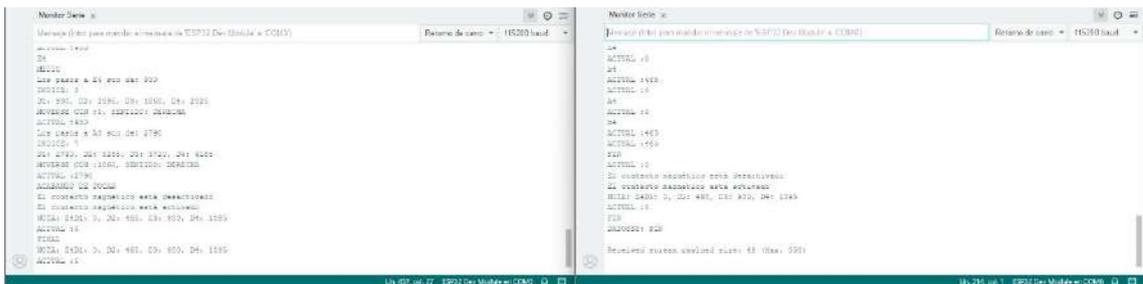
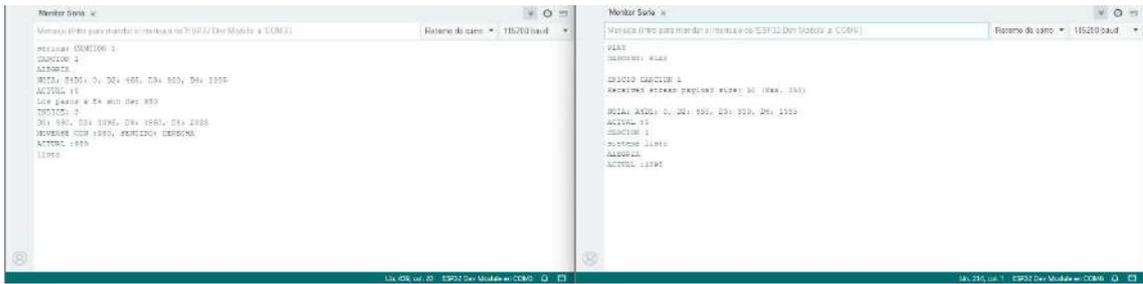
MANO_IZQUIERDA.ino
1967 if (inputString == "STOP") {
1968   inicio();
1969   fincancion();
1970 }
1971 }
1972 }
1973 }
1974 }
1975 }
1976 }
1977 }
1978 }
1979 }
1980 }
1981 }
1982 }
1983 }
1984 }
1985 }
1986 }
1987 }
1988 }
1989 }
1990 }
1991 }
1992 }
1993 }
1994 }
1995 }
1996 }
1997 }
1998 }
1999 }
2000 }
2001 }
2002 }
2003 }
2004 }
2005 }
2006 }
2007 }
2008 }
2009 }
2010 }
2011 }
2012 }
2013 }

```


ANEXO K: APLICACIÓN MÓVIL



ANEXO L: PRUEBAS DE COMUNICACIÓN VISUALIZADAS EN EL MONITOR SERIAL IDE





ANEXO M: HOJA DE DATOS DE LA PINZA AMPERIMÉTRICA

The meter is completely portable, LCD, 3½ digit clamp meter. It has rugged design, is easy to hold in operator's hand and convenient to use.

1. Safety Information

- 1.1 Read the following safety information carefully before attempting to operate or service the meter.
- 1.2 To avoid damages to the instrument do not exceed the maximum limits of the input values show in the technical specifications tables.
- 1.3 Never measure current while the test leads are inserted into the input jacks.
- 1.4 Do not use the meter or test leads if they look damaged. Use extreme caution when working around bare conductors or bus bars.
- 1.5 Caution when working with voltages above 60VDC or 30VAC RMS. Such voltages pose a shock hazard.

2. Operating Features

| | |
|------------|---------------|
| AC Current | 0.01A to 400A |
| AC Voltage | 1V to 450V |
| DC Voltage | 1V to 600V |

The meter display is a liquid crystal assembly providing a readily display in all light conditions. The decimal point is automatically positioned, and the polarity sign (minus) is lighted for negative DC

measurement (plus is understood if no sign appears), so that the display is direct reading in units selected at the rotary switch. Overrange measurements are indicated by blanking all but the MSD, decimal point, and polarity sign (if negative). In addition the display includes a low battery indication, if low battery is indicated, operator should replace the used battery with new one.

3. Specifications

The following Specifications assume a one year calibration cycle and an operating temperature of 64°F to 82°F (18°C to 28°C) at relative humidity up to 80% unless otherwise noted.

3.1 AC Current (Average sensing, calibrated to rms of sine wave)

| Range | Resolution | Accuracy(50Hz - 60Hz) |
|-------|------------|-------------------------------|
| 20A | 10mA | ± (3% of reading +5 digits) |
| 200A | 100mA | ± (2.5% of reading +5 digits) |
| 400A | 1A | ± (3% of reading +5 digits) |

(Overload protection: 400A on all ranges)

3.2 AC Voltage (Average sensing, calibrated to rms of sine wave)

| Range | Resolution | Accuracy(50Hz - 500Hz) |
|-------|------------|--------------------------|
| 450V | 1V | ± (2% reading +4 digits) |

Input impedance: 9MΩ

Overload protection: 450V AC/DC on all ranges.

3.3 DC Voltage

| Range | Resolution | Accuracy |
|-------|------------|---------------------------|
| 600V | 1V | ± (1.2% reading +3 digit) |

Overload protection: 600V DC/peak AC on all ranges
Input impedance: 9MΩ

3.4 Diode Test

Test current: <1.2mA
Open circuit voltage: ≤ 3.2V
Overload protection: 300V DC/peak AC
Application: Semiconductor P - N junction good or bad test
Display reading approx diode forward, voltage value.

3.5 Continuity Test:

Buzzer sound: <75Ω
Overload Protection: 300V DC/peak AC

3.6 Resistance

| Range | Resolution | Accuracy |
|--------|------------|------------------------------------|
| 200K Ω | | ± (1.5% of reading +5 digit) 100 Ω |

3.7 Environment

Temperature
Normal operation: 18°C to 28°C (64°F to 82°F)
Usable condition: 0°C to 50°C (32°F to 122°F)
Storage: -20°C to 60°C (-30°F to 140°F)
battery removed and < 80% RH
Relative Humidity: max 80%

3.8 Function characteristics

Measurement method: Dual slope integration

Reading Rate : 3 reading/sec

Polarity: Automatic, indicated minus, assumed plus
Overload indication: Blanking of all digits, except MSD, decimal point and sign

Power requirements: 1.5V x 2

Battery indication: Display indicates **LO B T** when approximately 20% of battery life remains

Display: LCD, 3½ digit (1999 count)

Data hold: All function and ranges with this feature

Dimension: 150 x 63 x 28mm (L x W x H)

approx

Weight: 145 grams (including battery, approx)

3.9 Accessories

Instruction manual

Test leads

AAA 1.5V x 2

4. Operation and Recalibration

1. Transformer Jaws:

Pick up the AC current flowing through the conductor.

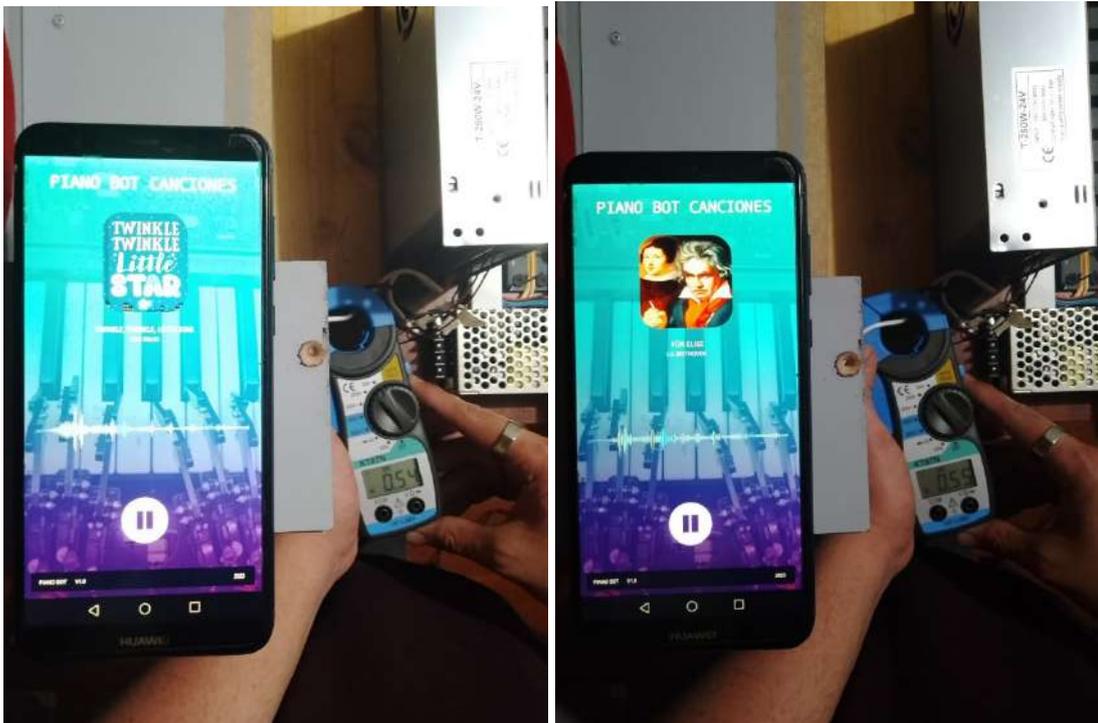
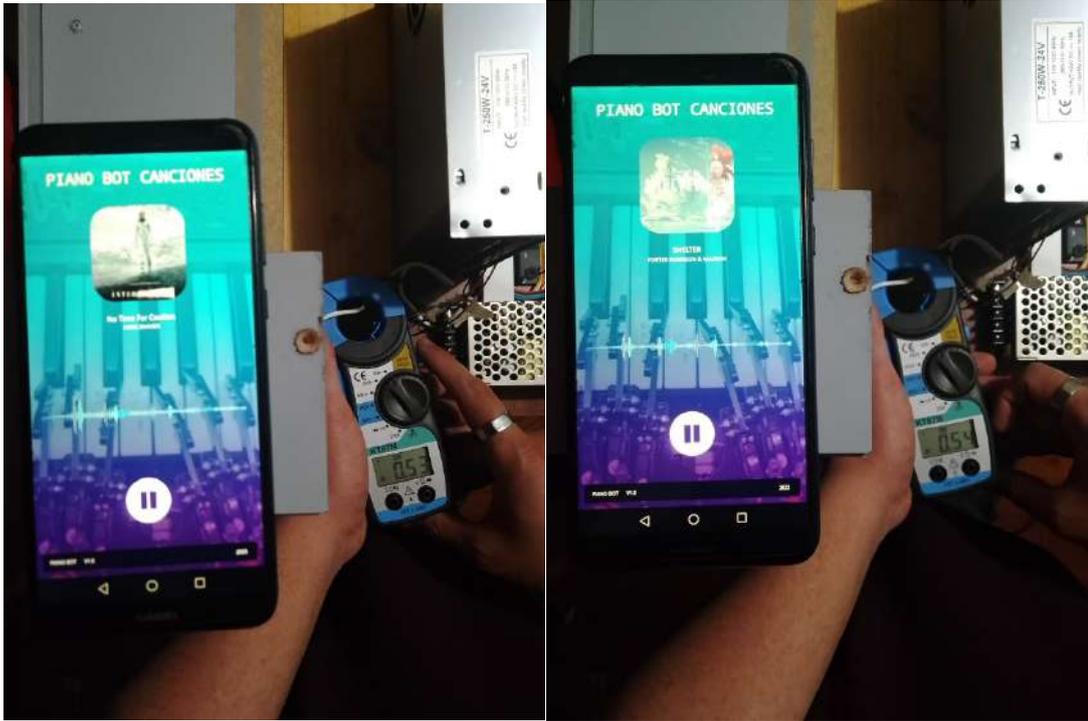
2. Trigger:

Press the lever to open the transformer jaws. When the lever is released, the jaws will close again.

3. Data Hold Switch:

A push switch, (push on, do not pull to select function). All function and ranges with this feature.

ANEXO N: PRUEBAS DE CONSUMO ENERGÉTICO





ESCUELA SUPERIOR POLITÉCNICA DE
CHIMBORAZO

DIRECCIÓN DE BIBLIOTECAS Y RECURSOS DEL
APRENDIZAJE



UNIDAD DE PROCESOS TÉCNICOS
REVISIÓN DE NORMAS TÉCNICAS, RESUMEN Y BIBLIOGRAFÍA

Fecha de entrega: 04/ 10 / 2023

| |
|---|
| INFORMACIÓN DE LOS AUTORES |
| Nombres – Apellidos: Marco Fabricio Aguirre Salas – Jean Carlos Meneses Rivera |
| INFORMACIÓN INSTITUCIONAL |
| Facultad: Informática y Electrónica |
| Carrera: Electrónica y Automatización |
| Título a optar: Ingeniero en Electrónica y Automatización |
| f. Analista de Biblioteca responsable: Ing. Fernanda Arévalo M. |

~~X BERTHA QUINTANIZA~~

