

UNIVERSIDAD CATÓLICA DE LA SANTÍSIMA CONCEPCIÓN

FACULTAD DE INGENIERÍA
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA



UCSC

Desarrollo de sistema basado en guante háptico para
el control de entornos inteligentes.

Roberto Esteban Vives Arriagada.

Informe de Habilitación Profesional
Ingeniería Civil Eléctrica

Profesor Patrocinante:
Dra. Silvia E. Restrepo M.

Profesores Guía:
Dr. Ricardo Bustos P.
Dra. Macarena Espinilla E.

Concepción, abril de 2025

UNIVERSIDAD CATÓLICA DE LA SANTISIMA CONCEPCION
Facultad de Ingeniería
Departamento de Ingeniería Eléctrica

Profesor Patrocinante:
Dra. Silvia E. Restrepo M.

Desarrollo de dispositivo biométrico para el control de entornos inteligentes

Roberto Esteban Vives Arriagada

Informe de Habilitación Profesional
para optar al Título de Ingeniero Civil Eléctrico

Abril 2025

Resumen

En este trabajo se presenta el desarrollo de un sistema de control háptico sensorizado, implementado en forma de un guante inteligente orientado al control de entornos domóticos mediante tecnologías de Internet de las Cosas (IoT). El sistema permite el reconocimiento de gestos manuales en tiempo real, facilitando la interacción del usuario con dispositivos del hogar inteligente de manera intuitiva y no intrusiva.

El guante incorpora sensores analógicos tipo potenciómetro, los cuales permiten medir el ángulo de flexión de las articulaciones de los dedos. Esta información es procesada para identificar gestos específicos que desencadenan acciones dentro del entorno controlado.

A diferencia de otros sistemas hápticos, comúnmente utilizados en aplicaciones médicas o de realidad virtual, esta propuesta se enfoca en una solución funcional y adaptable al ámbito de la domótica. El proyecto se estructura en tres fases: diseño e implementación del guante, construcción de un entorno de prueba con dispositivos representativos de una vivienda inteligente, y la integración del sistema con una plataforma IoT existente, específicamente Home Assistant, para la comunicación y control de los dispositivos conectados, así como la integración de una interfaz de usuario.

Finalmente, los resultados obtenidos durante la fase de pruebas validaron la viabilidad técnica del sistema propuesto. Se observó una respuesta rápida y precisa ante los gestos realizados con el guante, permitiendo la activación confiable de los actuadores asociados en el entorno simulado, control de cortina, control de cerradura de puerta, iluminación RGB, ventilación y alarma sonora. El sistema fue capaz de distinguir combinaciones complejas de movimientos sin pérdida de señal ni retardo perceptible, demostrando estabilidad en la comunicación mediante MQTT. La integración con dispositivos comunes en un entorno domótico permitió confirmar la funcionalidad del sistema como una interfaz intuitiva, precisa y eficaz para el control de escenarios inteligentes en tiempo real.

A los alumnos del pasado, presente y futuro de la carrera de Ingeniería Civil Eléctrica de la U.C.S.C.

Agradecimientos

Quiero agradecer a todas las personas que me han apoyado en mi carrera universitaria, en primer lugar, a mis padres, mi madre Jannette Arriagada, como mi padre Roberto Vives que siempre me dieron su apoyo, tanto en cuestiones académicas, y proyectos personales que terminaron guiándome al área de la carrera que termine eligiendo, a mis hermanos que fueron un pilar emocional fundamental a lo largo de todo este proceso. A mis tías, y primos que siempre me brindaron su apoyo en los momentos que necesité.

En segundo lugar, quiero agradecer a los amigos que hice a lo largo de esta carrera universitaria, que siempre fueron un apoyo y ayuda tanto en la parte emocional como académica, resaltando a María Ignacia Venegas, Juan Pablo Jara, Sebastián Basaur, Javier Barrientos, Barbara Robles y Skarlett Henríquez, que me acompañaron la mayor parte de mi carrera académica, así como a Camila Vallejos y Pablo Burgos que durante el último año siempre estuvieron para ayudarme.

Un importante agradecimiento a mi profesora guía Silvia Restrepo, con la que tuve el agrado de trabajar en diversos proyectos los últimos 3 años de carrera, y depositar su confianza en mis conocimientos y habilidades, y brindándome oportunidades tanto en la parte académica y de trabajo a lo largo del tiempo, así como a los profesores Ricardo Bustos, Ricardo Lizana, Samuel Vergara, que tuvieron en alto mis capacidades y me brindaron su apoyo en algún momento o a lo largo de la carrera.

Asimismo, deseo expresar agradecimiento al proyecto ANID FONDECYT N° 11221231 por el apoyo brindado, y financiamiento para poder llevar a cabo este proyecto.

Al Proyecto ING2030 (ING222010004), por el financiamiento otorgado para la realización de la pasantía en España en el marco de este trabajo. Asimismo, a la profesora Macarena Espinilla por su valiosa invitación, que me permitió ampliar mis conocimientos en áreas con las que no estaba familiarizado e integrarlos activamente en el desarrollo del proyecto.

Tabla de Contenidos

ABREVIACIONES.....	IX
CAPÍTULO 1. INTRODUCCIÓN	10
1.1. INTRODUCCIÓN GENERAL	10
1.2. TRABAJOS PREVIOS	11
1.2.1 automatización del hogar basada en IoT.....	11
1.2.2 Sistemas hápticos portátiles y guantes inteligentes	12
1.2.3 Reconocimiento de gestos de la mano	14
1.2.4 Exoesqueletos robóticos y control biométrico.....	15
1.2.5 Discusión	16
1.3. HIPÓTESIS DE TRABAJO	17
1.4. OBJETIVOS	17
1.4.1 Objetivo General	17
1.4.2 Objetivos Específicos.....	17
1.5. ALCANCES Y LIMITACIONES	17
1.6. TEMARIO Y METODOLOGÍA.....	18
CAPÍTULO 2. TECNOLOGÍA, COMPONENTES Y SOFTWARE.....	19
2.1. INTRODUCCIÓN	19
2.2. TECNOLOGÍA HÁPTICA	19
2.3. ENTORNOS INTELIGENTES.....	21
2.4. COMPONENTES Y SOFTWARE.....	22
2.4.1 Guante Háptico	23
2.4.2 Entorno de Pruebas.....	26
2.4.3 Servidor central.....	29
2.4.4 Plataformas y Software Utilizados	31
2.4.5 Materiales de construcción	32
CAPÍTULO 3. DISEÑO	35
3.1. INTRODUCCIÓN	35
3.2. GUANTE.....	35
3.2.1 Diseño Estructura física del guante.....	36
3.2.2 Diseño del sistema electrónico del dispositivo.	40
3.3. ENTORNO SIMULADO	44
3.3.1 Diseño Estructura Física del Entorno de Pruebas	44
3.3.2 Diseño del sistema electrónico del Entorno.	47
3.4. SERVIDOR CENTRAL.....	53
3.4.1 Selección de complementos para Home Assistant.	54
CAPÍTULO 4. IMPLEMENTACIÓN DE LOS SISTEMAS.....	56
4.1. INTRODUCCIÓN	56
4.2. GUANTE.....	56
4.2.1 Impresión de Placa PCB.....	57
4.2.2 Montaje del guante.	58
4.2.3 interacción entre guante y servidor central	59
4.3. ENTORNO SIMULADO.....	62
4.3.1 Impresión de Placa PCB.....	62
4.3.2 interacción entre el sistema domótico y servidor central.....	66
4.4. SERVIDOR CENTRAL.....	67
4.4.1 Configuraciones iniciales Home Assistant.....	68
4.4.2 Funcionamiento interno e interacción con el Guante y Entorno simulado	74
CAPÍTULO 5. RESULTADOS	76

5.1. INTRODUCCIÓN	76
CAPÍTULO 6. CONCLUSIONES.....	84
6.1.1 <i>Guante Háptico</i>	84
6.1.2 <i>Entorno Simulado</i>	84
6.1.3 <i>Servidor Central</i>	85
6.2. TRABAJO FUTURO	85
BIBLIOGRAFÍA	87

Abreviaciones

IoT	: Internet de las cosas (Internet of Things).
ICT.	: Tecnologías de la información y la Comunicación (Information and Communication Technology).
VCC	: Voltaje de alimentación (Voltage Common Collector).
GND	: Tierra (Ground).
FDM.	: Modelado por Deposición Fundida (Fused Deposition Modeling).
IDE	: Entorno de desarrollo integrado (Integrated Development Environment).
PCB.	: Placa de circuito impreso (Printed Circuit Board).
MQTT	: Transporte de Telemetría en Cola de Mensajes (Message Queuing Telemetry Transport).
LED	: Diodo emisor de luz (Light Emitting Diode).
RGB	: Rojo, verde, azul (Red, Green, Blue).
RAM	: Memoria de acceso aleatorio (Random Access Memory).
PLA	: Ácido Poliláctico (Polylactic Acid).

Capítulo 1. Introducción

1.1. Introducción General

La domótica y el Internet de las Cosas (IoT) están transformando la forma en que las personas interactúan con sus hogares, promoviendo entornos más eficientes, cómodos y personalizados. Actualmente, es común encontrar dispositivos que permiten automatizar y controlar funciones como la iluminación, la climatización o la seguridad, adaptándose a las necesidades del usuario y optimizando el uso de los recursos del hogar mediante la monitorización remota.

Este proyecto se enfoca en el desarrollo de un sistema y dispositivo de control manual para entornos inteligentes, implementado en forma de un guante sensorizado con tecnologías IoT. A diferencia de los sistemas tradicionales, que se basan en controles remotos o en automatizaciones dependientes de sensores ambientales, esta propuesta plantea una interfaz física e intuitiva basada en gestos biométricos de la mano, permitiendo un control más natural y flexible.

El primer paso del proyecto consiste en el diseño e implementación del guante sensorizado. Esto incluye el modelado tridimensional del dispositivo en el software Fusion360, la selección de los sensores y la integración electrónica necesaria en el guante háptico para el registro del movimiento de los dedos. Se emplean sensores analógicos para medir el ángulo de flexión de las articulaciones, con el fin de identificar diferentes posiciones y gestos característicos.

Posteriormente, se procederá a la validación del sistema mediante pruebas funcionales en un entorno controlado. Para ello, se definirá un conjunto de gestos específicos que permitirán activar distintos dispositivos domóticos, tales como sistemas de iluminación o actuadores eléctricos. Estas acciones estarán gestionadas a través de una plataforma servidor central, responsable de recibir, interpretar y distribuir las señales generadas por el guante sensorizado mediante protocolos de comunicación IoT. De esta forma, se evaluará tanto la viabilidad técnica del control háptico como su efectividad dentro de un ecosistema domótico interconectado, mediante la ejecución de pruebas prácticas que incluyeron la detección de movimientos individuales de cada articulación, la activación correcta de los actuadores en función de los gestos detectados y la estabilidad de la comunicación en tiempo real a través del servidor que integra Home Assistant.

1.2. Trabajos Previos

1.2.1 Automatización del hogar basada en sistemas IoT

- ♣ C. Stolojescu-Crisan, C. Crisan, and B.-P. Butunoi, "An IoT-based smart home automation system," *Sensors*, vol. 21, no. 11, p. 3784, May 2021. doi: 10.3390/s21113784.

Se propone un sistema de automatización del hogar denominado *qToggle*, basado en IoT, que interconecta sensores, actuadores y otras fuentes de datos mediante una API flexible. El sistema emplea dispositivos como ESP8266 y Raspberry Pi, y permite el control de electrodomésticos a través de una aplicación móvil, destacando por su facilidad de uso y escalabilidad.

- ♣ V. A. Orfanos, S. D. Kaminaris, D. Piromalis, and P. Papageorgas, "Smart Home Automation in the IoT Era: A Communication Technologies Review," *AIP Conference Proceedings*, vol. 2307, no. 1, p. 020054, Dec. 2020. doi: 10.1063/5.0032939.

En este artículo se revisan las tecnologías de comunicación utilizadas en la automatización de hogares inteligentes en la era del Internet de las Cosas (IoT). Se analizan las tecnologías alámbricas e inalámbricas, destacando las características técnicas, la seguridad, la interacción del usuario y los costos asociados. El objetivo del artículo es ayudar a los consumidores y profesionales a elegir la tecnología de automatización del hogar más adecuada para sus necesidades, proporcionando una comparación detallada entre soluciones abiertas y propietarias.

- ♣ C. Irugalbandara, A. S. Naseem, S. Perera, S. Kiruthikan, and V. Logeeshan, "A Secure and Smart Home Automation System with Speech Recognition and Power Measurement Capabilities," *Sensors*, vol. 23, no. 5784, pp. 1–15, Jun. 2023. doi: 10.3390/s23135784.

Se desarrolla un sistema de automatización del hogar seguro que opera de forma offline, incorporando reconocimiento de voz local y medición de consumo energético. La arquitectura busca resolver problemas de privacidad, conexión inestable y dependencia de servicios en la nube, especialmente en países en desarrollo.

- ♣ H. Yar, A. S. Imran, Z. A. Khan, M. Sajjad, and Z. Kastrati, "Towards Smart Home Automation Using IoT-Enabled Edge-Computing Paradigm," *Sensors*, vol. 21, no. 4932, pp.

1–23, Jul. 2021. doi: 10.3390/s21144932.

En este artículo se propone un sistema de automatización del hogar inteligente basado en el paradigma de computación en el borde habilitado por IoT. Se utiliza un dispositivo Raspberry Pi como unidad central de control para conectar y controlar diversos dispositivos y sensores en el hogar. El sistema proporciona control remoto y automatización de los electrodomésticos, asegurando la seguridad y reduciendo el consumo de energía. Además, se utiliza una nube local para almacenar datos sensibles, lo que mejora la privacidad y reduce el consumo de ancho de banda y costos de almacenamiento.

- ♣ W. N. A. A. W. Husni, M. Faisal, P. J. Ker, D. N. T. How, M. A. Hannan, and M. A. Salam, "Intelligent Home Automation System for Disabled People," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 9, no. 2, pp. 4142–4148, Dec. 2019. doi: 10.35940/ijeat.B4921.129219.

En este artículo se propone un sistema de automatización del hogar inteligente para personas con discapacidad, utilizando una Raspberry Pi 3 Modelo B y la aplicación Blynk. El sistema permite a los usuarios controlar electrodomésticos y monitorear la seguridad del hogar mediante un smartphone. Además, incorpora sensores PIR para detectar la presencia de intrusos, activando alarmas y capturando imágenes. El objetivo es proporcionar una solución más segura y fácil de usar en comparación con tecnologías anteriores como Bluetooth y ZigBee.

1.2.2 Sistemas hápticos portátiles y guantes inteligentes

- ♣ C. Pacchierotti, S. Sinclair, M. Solazzi, A. Frisoli, V. Hayward, and D. Prattichizzo, "Wearable Haptic Systems for the Fingertip and the Hand: Taxonomy, Review, and Perspectives," *IEEE Transactions on Haptics*, vol. 10, no. 4, pp. 580–600, Oct. 2017. doi:10.1109/TOH.2017.2689006.

En este artículo se presenta una revisión de los sistemas hápticos portátiles para la yema de los dedos y la mano, enfocándose en los desafíos de diseño relacionados con la portabilidad. Se analizan varias tecnologías utilizadas para proporcionar retroalimentación táctil y kinestésica, y se discuten los

principales desafíos para el desarrollo de interfaces hápticas portátiles en términos de ergonomía y usabilidad.

- ♣ K. Song, S. H. Kim, S. Jin, S. Kim, S. Lee, J. Kim, J.-M. Park, and Y. Cha, "Pneumatic actuator and flexible piezoelectric sensor for soft virtual reality glove system," *Scientific Reports*, vol. 9, no. 8988, pp. 1–11, 2019. doi: 10.1038/s41598-019-45422-6.

En este artículo se propone un sistema de guante orientado a realidad virtual (VR) con sensores piezoeléctricos flexibles además de actuadores neumáticos suaves. El sistema permite la detección de movimientos de los dedos y proporciona retroalimentación táctil en tiempo real, lo que permite al usuario interactuar con objetos virtuales. Se destaca la capacidad del actuador para ofrecer una respuesta rápida y precisa sin necesidad de un compresor de aire externo. El guante fue probado en un entorno de VR, permitiendo al usuario manipular piezas en un tablero de ajedrez virtual.

- ♣ M. F. Ahmed, N. Hasan, M. W. Ullah, M. H. Kabir y M. Rahman, "Elderly Safety and Communication: An IoT-Based Smart Glove for Fall Detection and Hand Gesture Communication", 3rd International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE), pp. 1–8, Abr. 2024. doi: 10.1109/ICAEEE62219.2024.10561702.

Se presenta un guante inteligente basado en IoT orientado a adultos mayores, que integra detección de caídas y comunicación mediante gestos de la mano. El sistema emplea un sensor MPU6050 conectado a un microcontrolador Arduino UNO y un módulo WiFi ESP8266, enviando alertas vía SMS y correo electrónico a través de la plataforma IFTTT. El guante permite transmitir mensajes predefinidos, como “necesito agua” o “ayuda urgente”, mediante movimientos direccionales de la mano. El evento se visualiza además en una pantalla LCD. La propuesta destaca por su bajo costo y facilidad de implementación, ofreciendo una solución asequible y eficaz para mejorar la seguridad y autonomía de personas mayores o con discapacidades del habla y movilidad.

- ♣ M. F. Ahmed y N. Hasan, "IoT-Enabled Smart Glove for Enhanced Elderly Care: A Health Monitoring Solution", *Preprint*, Research Square, pp. 1–25, Oct. 2024. doi: 10.21203/rs.3.rs-4956656/v1.

Este artículo presenta un sistema de guante inteligente enfocado en el monitoreo de adultos mayores, combinando tecnología IoT con sensores biométricos. El guante integra un acelerómetro giroscópico MPU6050 y un sensor MAX30100, que permiten detectar caídas y medir parámetros vitales como frecuencia cardíaca y saturación de oxígeno (SpO2). Además, el sistema permite enviar mensajes de emergencia mediante gestos de la mano, tales como "necesito comida" o "ayuda urgente", que se transmiten por WiFi a través de la plataforma IFTTT.

- ♣ S. B, P. P, P. Gupta, S. Aluvala and T. Stephan, "Enhanced Health Monitoring Using IoT-Embedded Smart Glove and Machine Learning," *2023 3rd International Conference on Innovative Sustainable Computational Technologies (CISCT)*, Dehradun, India, 2023, pp. 1-5, doi: 10.1109/CISCT57197.2023.10351358.

Se propone un sistema embebido de monitoreo de salud basado en un guante inteligente con conectividad IoT, diseñado para capturar y procesar parámetros fisiológicos como la temperatura corporal, frecuencia cardíaca y presión arterial. El guante integra sensores LM35, BMP180 e INVENTO, conectados a una placa Arduino Uno y un módem GSM. Los datos recolectados se transmiten a MATLAB, donde un modelo entrenado con el algoritmo *K-Nearest Neighbor* (KNN) clasifica el estado de salud del usuario como normal o anormal. En caso de detección de anomalías, el sistema genera prescripciones automáticas y las comunica tanto al usuario como al médico mediante mensajes.

1.2.3 Reconocimiento de gestos de la mano

- ♣ M. Oudah, A. Al-Naji, and J. Chahl, "Hand Gesture Recognition Based on Computer Vision: A Review of Techniques," *Journal of Imaging*, vol. 6, no. 8, pp. 73–89, Jul. 2020. doi: 10.3390/jimaging6080073.

En este artículo se propone una revisión de las técnicas de reconocimiento de gestos de la mano mediante visión por computadora. Se analizan diversos enfoques, como la segmentación de color de la piel, reconocimiento de movimientos y técnicas basadas en modelos de apariencia. El documento compara el rendimiento de estos métodos en términos de precisión, aplicabilidad y limitaciones en

escenarios reales.

- ♣ H. Luo, Y. -B. Lin, C. -C. Liao and Y. -H. Huang, "An IoT-Based Microservice Platform for Virtual Puppetry Performance," in *IEEE Access*, vol. 11, pp. 103014-103032, 2023, doi: 10.1109/ACCESS.2023.3315656.

Se desarrolla una plataforma basada en IoT y microservicios para la interpretación remota de marionetas virtuales. El sistema integra sensores inerciales y un guante inteligente para capturar los movimientos de la mano del usuario, los cuales son transmitidos en tiempo real a través de una arquitectura distribuida. Esta arquitectura está compuesta por servicios independientes conectados mediante MQTT y alojados en contenedores Docker, lo que facilita la escalabilidad y el mantenimiento. La propuesta permite representar de forma interactiva movimientos humanos en un entorno virtual 3D, con aplicaciones potenciales en entretenimiento, educación y rehabilitación.

1.2.4 Exoesqueletos robóticos y control biométrico

- ♣ A. Atzori, A. Gijsberts, C. Castellini, B. Caputo, A. Hager, S. Elsig, G. Giatsidis, F. Bassetto y H. Müller, "EMG-based hand gesture recognition: comparison of features and classifiers," *Biomedical Signal Processing and Control*, vol. 14, pp. 246–255, Oct. 2014. doi: 10.1016/j.bspc.2014.08.009.

En este artículo se presenta un modelo de mano impulsado por señales EMG, diseñado para identificar movimientos individuales de los dedos mediante técnicas de clasificación. La propuesta tiene como objetivo facilitar el control preciso de dispositivos robóticos o prótesis, permitiendo reproducir gestos específicos con alta exactitud. Para ello, se utilizan señales EMG registradas desde músculos del antebrazo, aplicando clasificadores como SVM, k-NN y redes neuronales artificiales. Además, se implementa un modelo dinámico de una mano virtual para validar los gestos identificados, logrando una correlación alta entre las trayectorias estimadas y los movimientos reales, lo cual evidencia la efectividad del sistema.

- ♣ K. Li, Z. Li, H. Zeng, and N. Wei, "Control of Newly-Designed Wearable Robotic Hand Exoskeleton Based on Surface Electromyographic Signals," *Frontiers in Neurorobotics*, vol.

15, no. 711047, Sep. 2021. doi: 10.3389/fnbot.2021.711047.

En este artículo se propone un exoesqueleto robótico portátil para la mano, controlado mediante señales electromiográficas superficiales (sEMG). El exoesqueleto cuenta con múltiples grados de libertad, permitiendo movimientos independientes de los dedos y el pulgar. Se utilizan varios clasificadores para interpretar las señales sEMG, logrando una alta precisión en el control de los movimientos de la mano, lo que puede ser útil para la rehabilitación de pacientes post-ictus.

1.2.5 Discusión

Como se ha revisado en los documentos y estudios analizados, los sistemas de automatización del hogar y los dispositivos portátiles han cobrado un interés creciente en los últimos años, impulsados por el avance de las tecnologías IoT, los sensores inteligentes y los desarrollos en interfaces hápticas.

Si bien los trabajos considerados no abordan exactamente el mismo enfoque que el presente proyecto, pueden agruparse en cuatro líneas temáticas que resultan fundamentales para su desarrollo:

En primer lugar, se identifica una tendencia hacia la creación de sistemas de automatización para hogares inteligentes mediante el uso de controladores y sensores interconectados. Estos trabajos exploran diferentes arquitecturas para el monitoreo y control remoto, destacando la importancia de la interoperabilidad, la escalabilidad y la eficiencia energética dentro de entornos IoT.

En segundo lugar, se aborda el reconocimiento de gestos como método de interacción con sistemas inteligentes. La mayoría de los enfoques analizados utilizan algoritmos de visión por computadora y aprendizaje automático. Aunque el proyecto actual se basa en sensores físicos y no en visión, los principios de segmentación, detección de patrones y clasificación son extrapolables y pueden adaptarse al diseño de algoritmos para sensores analógicos y digitales.

En tercer lugar, varios trabajos se centran en el diseño de exoesqueletos o dispositivos portátiles enfocados en la rehabilitación médica, particularmente mediante el uso de señales EMG. Aunque este enfoque se aleja de la domótica, se rescatan aportes valiosos en cuanto al diseño ergonómico, la integración sensorial y la interpretación de movimientos como base para generar acciones físicas o virtuales.

Finalmente, se resalta la preocupación por la seguridad y privacidad en los sistemas de automatización. Diversas propuestas destacan la importancia de operar de forma local u offline, evitando la dependencia de servicios en la nube y protegiendo los datos del usuario. Este aspecto es

clave en el diseño del presente proyecto, ya que se busca ofrecer un sistema confiable y seguro para los usuarios del guante sensorizado.

Actualmente, existe un creciente interés por implementar formas de interacción más naturales entre los usuarios y los sistemas domóticos. Frente a métodos tradicionales como interruptores físicos, interfaces móviles o comandos por voz, el uso de gestos manuales representa una alternativa directa y no intrusiva, que permite ejecutar acciones de manera más intuitiva. Este tipo de interacción busca reducir la dependencia de interfaces gráficas y facilitar el acceso a personas con distintos niveles de familiaridad tecnológica. En este contexto, el presente proyecto propone un sistema que permite el control del entorno inteligente mediante movimientos de la mano, mejorando la fluidez y personalización en el uso de tecnologías IoT dentro del hogar.

1.3. Hipótesis de Trabajo

La implementación de un sistema de control háptico mediante un dispositivo sensorizado tipo guante permitirá gestionar dispositivos en entornos domóticos de manera intuitiva y eficiente, optimizando la interacción entre el usuario y el entorno de hogar inteligente.

1.4. Objetivos

1.4.1 Objetivo General

Desarrollar un dispositivo sensorizado portátil capaz de detectar, gestos manuales, y que en base a ello se puedan controlar dispositivos a distancia dentro de entornos inteligentes.

1.4.2 Objetivos Específicos

- Investigar sobre sensores biométricos y entornos inteligentes.
- Identificar gestos manuales que podrían utilizarse como referencia de control.
- Identificar sensores a utilizar para recopilar información de los gestos asociados a las acciones.
- Desarrollar un prototipo del dispositivo biométricos.
- Implementar el dispositivo en el entorno de pruebas.

1.5. Alcances y Limitaciones

- Investigación del uso de sensores biométricos.
- Desarrollo de un guante sensorizado.

- Exploración de tecnologías de sensores.
- Análisis de aplicaciones en la interacción y control de dispositivos del hogar inteligente.
- Implementación de acciones en base a los gestos manuales.
- Validación en un entorno de pruebas de un hogar inteligente.

1.6. Temario y Metodología

En los capítulos que se desarrollarán en el presente documento a partir del capítulo dos estarán distribuidos de la siguiente manera:

- **Capítulo 2:** Se presentarán los conceptos, tecnologías, así como los componentes físicos, sensores, actuadores, junto con las herramientas y materiales.
- **Capítulo 3:** Se presentará el diseño de los sistemas desarrollados, detallando las motivaciones que llevaron a optar por el sistema adoptado.
- **Capítulo 4:** Se implementarán los dispositivos diseñados en el capítulo 2, y se probará su funcionamiento.
- **Capítulo 5:** Resultados.
- **Capítulo 6:** Conclusiones.

Capítulo 2. Tecnología, componentes y software

2.1. Introducción

En este capítulo se describirán las distintas tecnologías, componentes y software necesarios para la realización del proyecto, así como las plataformas y herramientas a utilizar en los capítulos posteriores.

2.2. Tecnología Háptica

Esta tecnología se basa en la retroalimentación háptica, cuyo principio de funcionamiento implica la generación de estímulos físicos, como vibración, presión y textura para replicar sensaciones táctiles reales. Los encargados de esta tarea son componentes específicos, que pueden definirse de la siguiente manera:

Actuadores hápticos: Son los encargados de generar las respuestas y estímulos táctiles al usuario, permiten percibir sensaciones táctiles/físicas mediante la interacción de lo que puede ser una interfaz. Los medios para ello pueden ser mediante vibraciones, mecanismos electromecánicos, piezoeléctricos, fluidos entre otros. Hay distintos tipos de acuerdo con su aplicación, que son principalmente, en el ámbito de la realidad virtual o en la medicina en los procesos de rehabilitación

Sensores hápticos: Son los encargados de recopilar los datos de las interacciones del usuario con el medio, como movimiento, fuerza, velocidad o presión ejercida, entre otras. Permiten captar detalles de las interacciones físicas del usuario, para luego procesarlos y utilizarlo para adaptar respuestas en dispositivos y/o sistemas. En aplicaciones avanzadas, permiten medir los gestos y movimientos de la mano, haciendo interacciones precisas en medios virtuales o simulados.

Interfaces hápticas: Permiten la comunicación entre el usuario y el sistema, generando y capturando señales táctiles. Estas interfaces, las cuales pueden ser por ejemplo guantes o controles hápticos, son los dispositivos donde están integrados los actuadores y/o sensores hápticos, y comunican e integran con el usuario.

Software de control: Es el encargado de procesar la información captada por los sensores, y

coordinar las respuestas a los actuadores en tiempo real. Los algoritmos que determinan las características de las respuestas de acuerdo con la interacción generada por el usuario, además de permitir configurar y personalizar diferentes aplicaciones con el sistema.

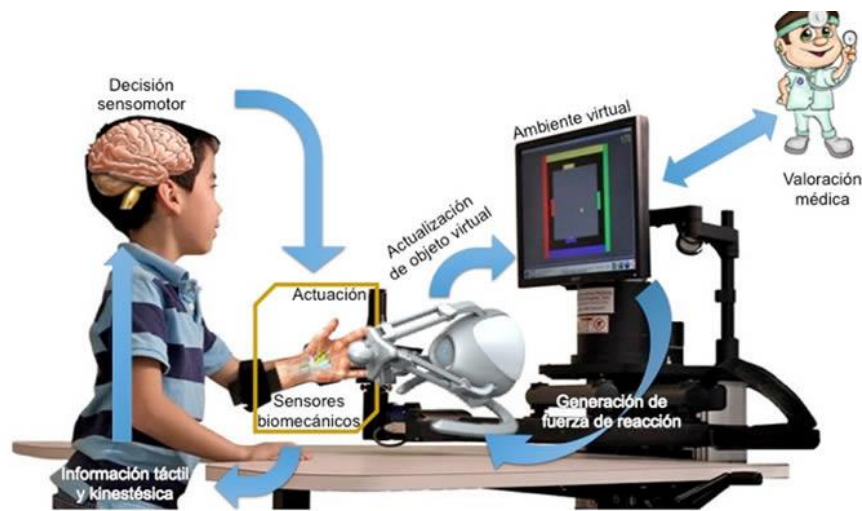


Fig. 2. 1 Ejemplo de Sistema Háptico aplicado a medicina [15]

Un punto importante para señalar es que según Spiceworks, una plataforma de profesionales en el sector ICT las tecnologías hápticas se pueden separar en 3 tipos, de acuerdo con la manera en que se comunican con el usuario (ver tabla [2.1]) [16]:

Vibración: Se envía una señal en forma de vibración al dispositivo que utiliza el usuario. Estas deben tener una razón específica, como lo podría ser una señal de alerta por algún tipo de problema en la actividad que se está realizando.

Presión de botones: Principalmente, se trata de botones mecánicos, en los que el usuario manda una señal intencional al sistema para obtener alguna respuesta e interactuar con la interfaz.

Quinestésicos: Corresponde a los dispositivos que van montados en el cuerpo del usuario, generan sensaciones físicas, como lo pueden ser la masa y el movimiento. Los guantes hápticos corresponden a esta categoría.

TABLA 2. 1 Tecnología háptica según su tipo

Tecnología	Tipo
Vibración	Actuador
Presión de botones	Sensor
Quinestésico	Sensor/Actuador

2.3. Entornos Inteligentes

Los entornos inteligentes, también denominados sistemas de domótica, se fundamentan en la automatización y control centralizado de distintos dispositivos dentro de espacios habitacionales o comerciales como se puede representar en la figura 2. 2. Su funcionamiento se basa en la integración de sensores, actuadores y tecnologías de comunicación, con el fin de gestionar de manera eficiente y personalizada aspectos clave del entorno como la iluminación, climatización, seguridad y consumo energético.

Estos sistemas permiten la interacción tanto local como remota, a través de plataformas de control que pueden incluir interfaces gráficas, asistentes virtuales o aplicaciones móviles, facilitando al usuario un acceso intuitivo y en tiempo real a las funciones del entorno.

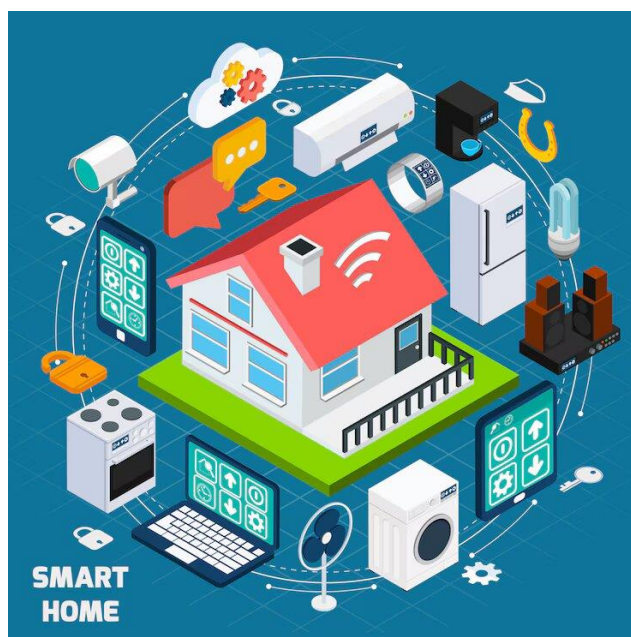


Fig. 2. 2 Representacion conexión de dispositivos en un entorno inteligente[17]

Entre las principales formas de interacción y control se pueden separar en [17]:

Control por voz: Mediante asistentes virtuales como Alexa, Google Assistant o Siri, el usuario puede emitir comandos para ejecutar acciones como encender luces o regular la temperatura.

Sistemas basados en sensores: Permiten automatizar respuestas según condiciones ambientales, como, por ejemplo, encender la luz al detectar presencia o cerrar persianas al medir una alta radiación solar.

Programación horaria: Facilita la ejecución automática de tareas rutinarias en horarios definidos, como encender el sistema de calefacción antes de la llegada del usuario o encender las luces exteriores de la casa cuando caiga la noche.

Control remoto: Brinda acceso al sistema desde cualquier lugar mediante dispositivos móviles o acceso web, permitiendo monitoreo y control a distancia.

Escenas y rutinas: Agrupan múltiples acciones en una sola instrucción, como un “modo noche” que apaga las luces, activar alarmas y ajusta la climatización.

Integración con otros sistemas: Los entornos inteligentes pueden interoperar con sistemas de seguridad, redes eléctricas inteligentes o plataformas IoT, ampliando sus capacidades y adaptabilidad.

2.4. Componentes y software

La elección de los componentes constituye una etapa crítica, ya que de ella depende el rendimiento, la precisión y la funcionalidad del sistema. Por este motivo, cada elemento ha sido evaluado considerando su compatibilidad eléctrica y mecánica, así como factores como el tamaño, la ergonomía y la adecuación para su uso en la mano.

El software desempeña un papel fundamental en este proyecto, tanto en la programación directa del dispositivo como en el diseño de los modelos 3D y las placas de circuito impreso (PCB) que conforman el guante. Asimismo, cumple un rol clave en la configuración e integración del

servidor con el resto de los dispositivos del sistema.

Esta sección se organiza en cuatro apartados principales: Guante háptico, Entorno de pruebas, Servidor central y Herramientas y materiales.

En la figura 2.3 se muestra de manera general el diagrama de interconexión del sistema, compuesto por tres bloques principales: el guante háptico, el servidor central y el entorno simulado. El guante integra sensores conectados a un multiplexor y un Arduino, que se comunica con el servidor central mediante Home Assistant. Este, a su vez, transmite los datos al entorno simulado, donde un segundo Arduino controla los actuadores en función de la información recibida.

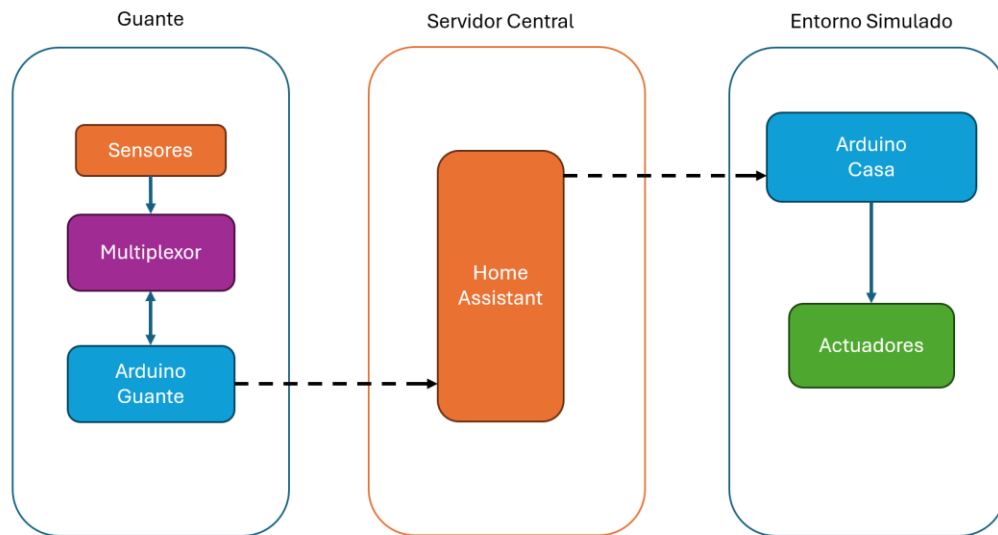


Fig. 2. 3 Diagrama General de interconexión sistema

2.4.1 Guante háptico

Se abarcarán los elementos electrónicos, mecánicos y de procesamiento que conforman el guante háptico desarrollado para este proyecto. Se incluyen tanto los sensores encargados de medir el ángulo de inclinación de los dedos, como los componentes necesarios a nivel lógico para captar estas mediciones y enviarlas al servidor central, así como el software y programas necesarios para su correcto funcionamiento.

En la Figura 2. 4, se muestra la posición aproximada de los componentes electrónicos del

guante háptico. En color azul se representan la posición y cantidad de potenciómetros tipo trimmer, los cuales permiten la medición de la flexión de las articulaciones. En color amarillo esta representado un demultiplexor de tipo 74HC4067. En naranja se muestra posición de un microcontrolador Arduino nano ESP32. La decisión de la cantidad, componentes, cantidad y ubicación de estos serán explicados de manera detallada en el capítulo 3.

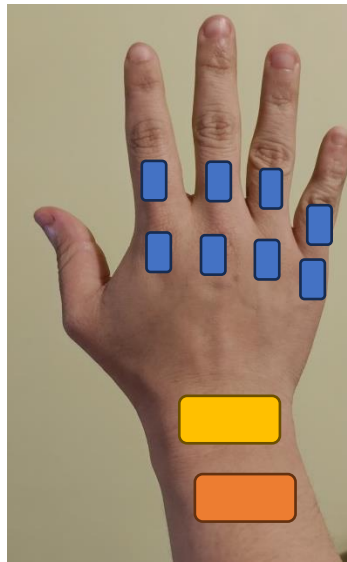


Fig. 2. 4 Posición de componentes guante háptico

En la Figura 2.5 se muestra un potenciómetro tipo trimmer junto a su pinout de conexión. El terminal 1 corresponde a la entrada de voltaje, con un rango de 0 [V] a 5 [V]. El terminal 2 es la salida de datos del potenciómetro, de tipo analógico, y el terminal 3 se conecta a tierra.

Este componente es el principal sensor del dispositivo desarrollado. Gracias a su parte rotativa, permite generar una señal analógica proporcional al ángulo de giro, lo que lo convierte en una opción adecuada para registrar el movimiento de las articulaciones de los dedos.

Se eligió el formato trimmer por su tamaño reducido, lo que facilita la incorporación de múltiples sensores sin comprometer el espacio o la comodidad del usuario. Además, ofrece buena precisión en las mediciones realizadas.

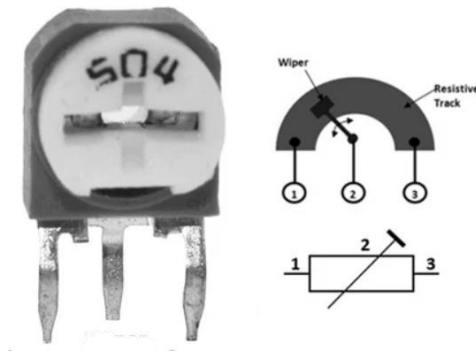


Fig. 2. 5 Potenciómetro tipo trimmer con su Pinout

El Arduino nano ESP32 mostrado en la figura 2. 6 es una placa de desarrollo basado en el microcontrolador ESP32, mientras que su distribución y estructura siguen la forma de un Arduino nano utilizando un espacio reducido, lo cual es ideal para un dispositivo que va unido al usuario. Esto permite que las comodidades y la fácil distribución de los pines de la placa trabajen en conjunto al microcontrolador ESP32, el cual además de contar con dos núcleos, para poder realizar hasta dos procesos en simultáneo, cuenta con conectividad bluetooth e internet mediante WIFI. Esta placa funciona a un nivel lógico de 3.3[V] a nivel interno, pero admite una alimentación en el rango de los 3 y 12 voltios.

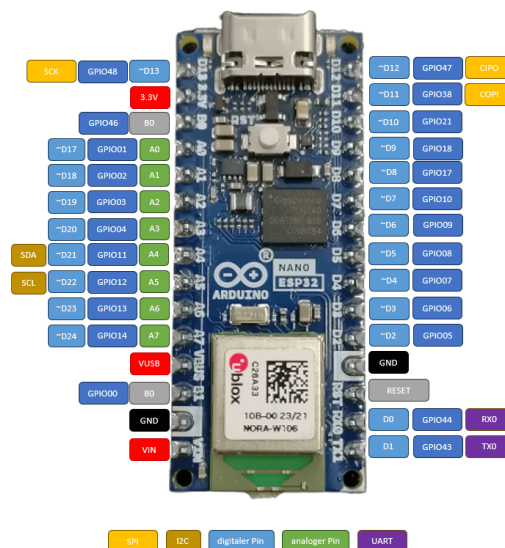


Fig. 2. 6 Arduino nano ESP32

En la figura 2. 7, se puede apreciar un módulo basado en el chip 74HC4067, el cual puede

operar como demultiplexor y multiplexor de 16 canales tanto de manera analógica como digital. Esto permite que, de un solo pin, denominado como SIG en el pinout, pueda llegar a comunicarse a 16 terminales, tanto como para enviar información a 16 dispositivos, o recibir datos de 16 sensores, lo que, en la práctica, permite aumentar en 15 pines de manera virtual las conexiones de algún dispositivo que lo esté utilizando de intermediario.

En el caso del proyecto es necesario usar dicho módulo, ya que el número de articulaciones en los dedos de la mano, cada una vinculada a un potenciómetro, supera el número de pines totales disponibles en la placa Arduino nano ESP32.



Fig. 2. 7 demultiplexor/multiplexor 74HC4067

2.4.2 Entorno de Pruebas

Se abordarán los elementos electrónicos y mecánicos que componen el entorno de pruebas utilizado para simular una vivienda inteligente a escala. Este entorno incluye actuadores como luces, ventiladores, cerraduras y cortinas automatizadas, los cuales responden a comandos enviados por el sistema central. Además, se incorporan los mecanismos de control necesarios para ejecutar las acciones generadas desde el guante, permitiendo validar el funcionamiento del sistema en condiciones similares a un entorno real.

En la Figura 2.8 se muestra, a modo ilustrativo, una representación esquemática de los principales componentes considerados en el entorno de pruebas. En ella se incluyen: una puerta con cerradura electrónica, una lámpara de pared para simular el sistema de iluminación, una cortina tipo roller accionada mediante motor, un ventilador de pie para la simulación de ventilación, y una bocina como elemento de alerta sonora. Todos estos dispositivos serán descritos en mayor detalle en el Capítulo 3, junto con su integración y funcionamiento dentro del sistema domótico desarrollado.

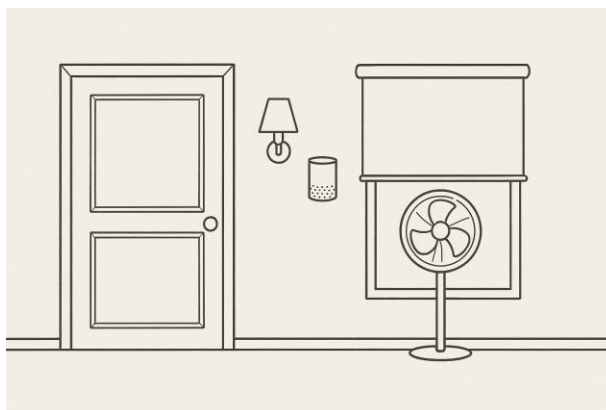


Fig. 2. 8 Representación de componentes a simular en una habitación

En la Figura 2. 9, se observa una cerradura eléctrica de tipo solenoide, diseñada para operar a 12 V. Este componente electromecánico permite controlar de forma electrónica el mecanismo de cierre y apertura de una puerta mediante la aplicación de una señal eléctrica. Cuando se aplica voltaje a la cerradura, el pestillo es retraído, desbloqueando la puerta; al retirar el voltaje, un resorte interno devuelve el pestillo a su posición original, bloqueando nuevamente el acceso.

La cerradura cuenta con una carcasa metálica resistente, un pestillo retráctil de movimiento lineal y un conector de dos pines para alimentación. Su estructura compacta y fácil integración la hacen adecuada para aplicaciones domóticas.



Fig. 2. 9 Cerradura eléctrica 12V

En la Figura 2.10 se presenta un motoreductor de corriente continua con reducción mecánica, diseñado para operar a un voltaje de 3 V y ofrecer una velocidad de salida aproximada de 25 revoluciones por minuto (RPM). Este tipo de motor incorpora una caja reductora metálica que

transforma la alta velocidad del motor original en una mayor fuerza de torque, permitiendo un movimiento lento, preciso y con mayor fuerza.

Gracias a su tamaño compacto, bajo consumo energético y eje metálico de salida, es ideal para aplicaciones de bajo torque. La estructura con engranajes metálicos garantiza durabilidad y resistencia bajo cargas mecánicas ligeras a medias.



Fig. 2. 10 Motoreductor de 3V 25RPM

En la Figura 2. 11, se muestra un LED RGB (Red-Green-Blue) de cátodo común con cuatro pines, correspondiente a cada uno de los colores primarios y una terminal común de conexión a voltaje. Este tipo de LED permite obtener una gran variedad de colores mediante la combinación proporcional de los tres canales básicos controlando la intensidad de corriente de cada uno.

Cada color puede ser regulado mediante modulación por ancho de pulso (PWM), lo que permite generar desde colores primarios hasta mezclas personalizadas. El encapsulado transparente favorece una emisión de luz clara y uniforme, útil tanto para señalización como para simulación de iluminación ambiental.



Fig. 2. 11 Led RGB cátodo común

En la Figura 2. 12, se muestra un ventilador de corriente continua (DC) de 5 V tipo brushless, comúnmente utilizado en sistemas de refrigeración de bajo consumo. Este ventilador opera a 0.1 A y está equipado con un conector de dos pines para su fácil integración en proyectos electrónicos.

Gracias a su tamaño compacto, bajo nivel de ruido y bajo consumo energético, es ideal para

sistemas embebidos donde se requiere simular o implementar ventilación localizada. Su rotor está impulsado por un motor sin escobillas (brushless), lo que le otorga mayor durabilidad y menor mantenimiento.



Fig. 2. 12 Ventilador 5V

En la Figura 2. 13, se presenta un buzzer pasivo de 5 V, un componente utilizado para generar sonidos o señales acústicas a través de la modulación de una señal digital externa. A diferencia de un buzzer activo, que emite un tono fijo al ser alimentado, el buzzer pasivo requiere una señal PWM (modulación por ancho de pulso) para determinar la frecuencia del sonido emitido, lo que permite una mayor versatilidad en su uso.

Este tipo de buzzer es comúnmente utilizado en sistemas de notificación o advertencia, ya que permite generar tonos personalizados y variados según la lógica de control del sistema. Funciona con bajo consumo de corriente y se integra fácilmente en sistemas controlados por microcontroladores.



Fig. 2. 13 Buzzer pasivo 5V

2.4.3 Servidor central

En la Figura 2.14 se presenta un esquema simplificado del flujo de datos del sistema, donde se destaca el rol del servidor central como intermediario entre el guante háptico y el entorno simulado.

Este servidor opera sobre una Raspberry Pi 5 y ejecuta la plataforma Home Assistant, encargada de interpretar las señales provenientes del guante y generar las acciones correspondientes en el entorno domótico. El funcionamiento detallado de esta plataforma y su integración con el sistema será abordado en el Capítulo 3.

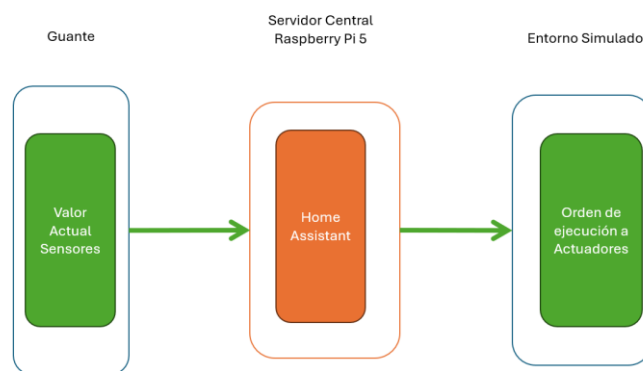


Fig. 2. 14 Esquema flujo de datos del sistema

Como servidor central se utilizó una Raspberry Pi 5 (ver figura 2. 15), una computadora de placa reducida (single-board computer) de última generación, que ofrece un alto rendimiento en un formato compacto. Está equipada con un procesador ARM Cortex-A76 de cuatro núcleos a 2.4 GHz, 8GB de memoria Ram, múltiples puertos USB 3.0, doble salida micro-HDMI, Ethernet Gigabit y soporte para sistemas operativos basados en Linux como Raspberry Pi OS o Ubuntu.

Esta placa es ampliamente utilizada como centro de procesamiento en sistemas IoT debido a su bajo consumo, conectividad avanzada y facilidad de integración con redes y periféricos. Además, permite el uso de lenguajes como Python y servidores ligeros como Flask, Mosquitto MQTT, o Home Assistant, lo que facilita el desarrollo de sistemas inteligentes y distribuidos.

permitiendo modelar con precisión las piezas que alojan los sensores y se ajustan a los dedos del usuario.

Además de permitir la simulación de movimientos en piezas móviles, destaca por su capacidad de exportar modelos a otras plataformas de manera sencilla. Gracias a su facilidad de uso, es una herramienta ideal para proyectos que requieren un prototipado rápido, ofreciendo también la ventaja de trabajar en la nube, lo que permite sincronizar los proyectos en múltiples dispositivos de forma eficiente.

Candle es un software de código abierto diseñado para el control de máquinas CNC, específicamente aquellas que utilizan firmware GRBL. Su interfaz intuitiva y liviana permite cargar, visualizar y ejecutar archivos de G-code de manera eficiente, facilitando el control de movimiento en sistemas de mecanizado.

En el contexto de este proyecto, Candle fue utilizado para controlar la máquina CNC durante el proceso de fabricación de la placa PCB del guante háptico, permitiendo verificar el correcto trazado de las pistas y supervisar en tiempo real la ejecución del mecanizado. Sus funciones destacadas incluyen el monitoreo en tiempo real de las coordenadas, la visualización previa del trazado y la creación automatizada de mapas de altura, aspecto fundamental para asegurar una impresión precisa de las pistas sobre la PCB.

Flatcam es una herramienta de software de código abierto orientada al procesamiento de archivos Gerber y Excellon, los cuales son ampliamente utilizados en el diseño y fabricación de circuitos impresos (PCB). Este programa ha sido especialmente desarrollado para transformar dichos archivos en trayectorias de mecanizado adecuadas para máquinas de control numérico computarizado (CNC), como routers o fresadoras.

2.4.5 Materiales de construcción

En la Figura 2. 16, se muestra la maquina CNC Ortur Aufero, una máquina de control numérico computarizado (CNC) de escritorio, utilizada principalmente para trabajos de grabado, fresado y corte en materiales como acrílico, madera, aluminio o placas PCB. Este tipo de máquina funciona mediante el control preciso de motores paso a paso que mueven un husillo rotativo a lo largo de tres ejes (X, Y, Z), siguiendo trayectorias generadas por software de diseño asistido por computadora (CAD/CAM).



Fig. 2. 16 Ortur CNC Aufero

El filamento PLA ilustrado en la figura 2. 17, corresponde a un material común en la impresión de piezas en 3D, el cual se derrite a una temperatura entre los 180°C y los 220°C. Es un material de fácil impresión, pero que a su vez entrega buenos resultados en calidad y resistencia. Es ideal para trabajar en prototipos en los que se requiere la impresión de varios modelos para probar en una sucesión rápida para irse ajustando progresivamente.



Fig. 2. 17 Filamento para impresión 3D PLA

Las impresoras de impresión 3D mostradas en la figura 2. 18, La CR6-SE y ender 3 v3 KE, son impresoras basadas en el sistema de FMD. Cuentan con una buena precisión para la elaboración de la estructura en la cual estará inserta la electrónica del proyecto, así como facilitar el prototipado de varios modelos de prueba para comprobar la ergonomía y la comodidad del uso.

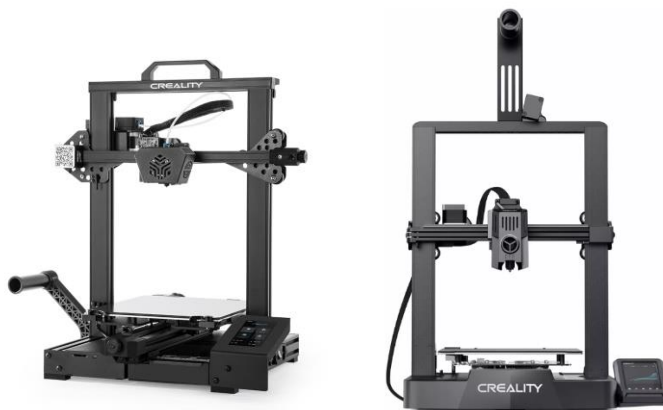


Fig. 2. 18 Impresoras 3D CR6-SE y ender 3 v3 KE

Capítulo 3. Diseño

3.1. Introducción

En este capítulo se expondrá el proceso, desde el diseño de la estructura física del circuito para poder realizar el sistema del guante háptico. Este diseño cuenta con el circuito esquemático, tablas de conexión junto a una explicación del funcionamiento de cada punto. Además, se explicará y mostrará el diseño de la estructura del guante en su papel de interaccionar con la mano del usuario, ya que al ser un dispositivo que va montado sobre el cuerpo se requiere algunas medidas precisas para poder funcionar de forma óptima.

3.2. Guante

La primera etapa para poder desarrollar este dispositivo fue analizar las distintas circunstancias a las que podrían verse comprometidos los sensores y componentes, los cuales estarían sometidos a cambios constantes y limitaciones inherentes.

Como el principal sensor utilizado para determinar los movimientos de la mano sería un potenciómetro, que funciona como un sensor de posición angular de tipo analógico y mecánico, se optó por emplear un método de observación práctica: realizando movimientos controlados de las articulaciones para verificar que tipo de sensor pudiera registrar correctamente los desplazamientos esperados.

En la figura 3. 1 se pueden ver diversas posiciones de mano, así como la distribución ósea de las articulaciones. Al probar distintas combinaciones, se determinó que la mejor posición de los sensores es en la parte superior de los dedos, ya que de esa forma causan la menor interferencia en la mayoría de las posiciones y movimientos que puede tener una mano en diversas circunstancias. Además, la forma óptima de medir los movimientos y posiciones de la mano corresponde a cada articulación de manera individual.



Fig. 3. 1 Ejemplo de posiciones de la mano y su estructura

Después de analizar los movimientos de la mano, se decidió omitir en una primera etapa el dedo pulgar para el dispositivo. Esto debido que a diferencia del resto de los dedos requiere una mayor libertad de movimiento, lo que podría requerir más sensores o más complejidad en su desarrollo.

Se determinó que las articulaciones entre la Falange Distal y Falange Medial actúan de modo espejo, de manera que cuando una de ellas es flexionada, la otra lo hace en la misma medida. Esto es un punto importante al requerir solo un sensor, donde se le asigna el mismo valor de posición angular a la articulación espejo.

Con estas consideraciones, para esta instancia del desarrollo se consideró dos sensores por cada dedo, dando una cantidad de 8 sensores como mínimo para el dispositivo, el primero queda ubicado entre la articulación entre la Falange Medial y la Falange proximal y el segundo entre la Falange Proximal y el Hueso Metacarpiano.

Una vez decididos los sensores, se procede a determinar los pasos necesarios para diseñar la estructura del sistema que va acoplada directamente con la mano del usuario, y que actúa como intermediario entre esta y los sensores del guante. Para facilitar el prototipado de esta parte, se determinó que la forma óptima de probar el sistema es mediante la impresión de cada pieza mediante impresión 3D.

3.2.1 Diseño estructura física del guante

En la figura 3. 2 se ilustran algunos de los modelos de prueba más relevantes de la estructura

que sostienen los sensores sobre el dedo, los cuales después de ser diseñados en la aplicación de Fusion360 fueron impresos en 3D para probar tanto su movilidad, ergonomía, comodidad y libertad que da al dedo en conjunto. En primer lugar, cada diseño sufrió pequeños cambios con su antecesor considerando 3 factores importantes, siendo el asegurar que las piezas no se desprendan del dedo al realizar algunos movimientos. En segundo lugar, la comodidad del usuario que al realizar dichos movimientos no entorpezca los movimientos naturales de la mano. Y como tercer factor la robustez de las partes para no romperse en caso de realizar algún movimiento fuerte o brusco.

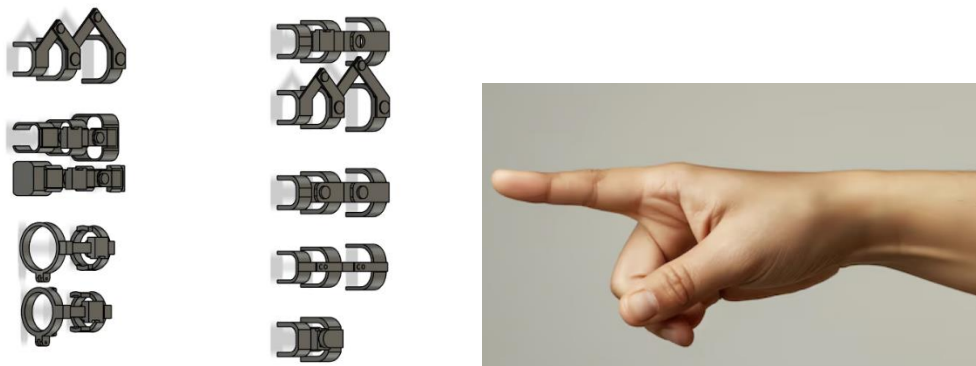


Fig. 3. 2 Posición de los modelos referentes al dedo índice

En la figura 3. 3 se puede apreciar el modelo elegido para ir acoplado en cada dedo utilizando como ejemplo el dedo índice, El modelo final fue construido seleccionando las partes óptimas de cada uno de los prototipos evaluados durante las pruebas. Se determinó que las secciones 1 y 4 de la imagen, encargadas de mantenerse sujetas al dedo, alcanzan un equilibrio ideal entre comodidad y firmeza cuando adoptan una forma de anillo.

Por otro lado, las secciones 2 y 3, ubicadas en la parte superior, cumplen la función de conectar ambos anillos y transmitir el movimiento desde uno al otro. Este movimiento es captado por un sensor situado entre las secciones 3 y 4. Se concluyó que la mejor configuración para esta conexión consiste en permitir la rotación de cada sección en los puntos de intersección. De esta forma, el sensor ubicado en la sección cuadrada entre las partes 3 y 4 actúa como eje de rotación y punto de unión, y sus lecturas se utilizarán como datos de entrada para el respectivo dedo.

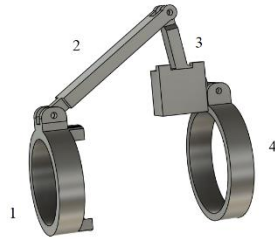


Fig. 3. 3 Diseño final respecto al dedo índice

Al analizar los grados de libertad con los que pueden rotar las articulaciones de los dedos de la mano, se determinó que estos movimientos oscilan en un rango entre 0° y 90° . Este análisis, junto con la disposición geométrica de las piezas 2 y 3 (ver figura 3.4), que vinculan el sensor con el dedo del usuario, requiere mantener una relación pitagórica para garantizar una medición precisa del ángulo de flexión.

Como se muestra en la figura 3.4, la disposición de estas piezas forma un triángulo rectángulo donde la pieza 2 actúa como la hipotenusa, mientras que la pieza 3 representa el cateto 2. Esta configuración asegura que el sensor registre una lectura de 90° cuando el dedo está completamente extendido (posición recta), y 0° cuando se encuentra en su punto máximo de flexión.

El largo de cada pieza fue determinado de la siguiente manera: el cateto 1 corresponde a la distancia entre el punto medio de cada anillo del dedo, el cateto 2 al largo necesario para evitar interferencias con los nudillos al flexionar el dedo, y la hipotenusa se calculó aplicando la fórmula pitagórica mostrada en la ecuación (3.1):

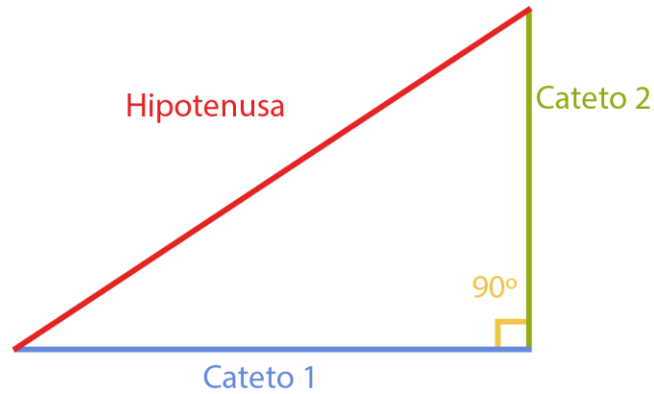


Fig. 3. 4 relación tamaño piezas de unión entre piezas y sensores

$$H^2 = C_1^2 + C_2^2 \quad (3.1)$$

En la figura 3. 5 se puede observar el modelo de la estructura encargada de unir las piezas del dedo índice con las de la mano. En este diseño, la pieza 3 actúa como soporte para el sensor, ubicado en la parte superior, siguiendo el mismo principio ilustrado previamente en la figura 2.4. Las piezas 1, 2, 3, 4, 5 y 6 corresponden al dedo índice, en donde las piezas 1 y 4, son los anillos que mantienen la estructura sujeta al dedo, y en donde las piezas 2, 3 y 5, 6 forman el mecanismo articulado que permite la flexión del dedo. Finalmente, la pieza 7 representa la estructura de sujeción a la mano, sirviendo como base estructural para mantener todas las secciones unidas de forma mecánica.

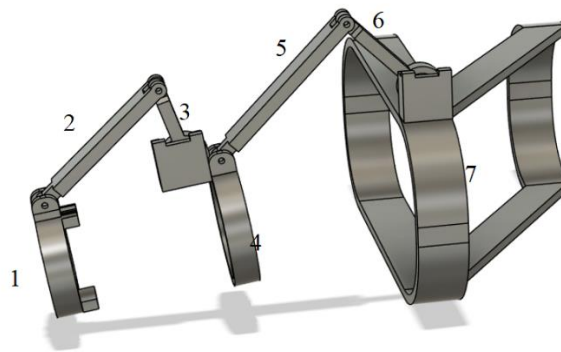


Fig. 3. 5 pieza unión mano junto a dedo índice

En las Tablas [3.1] y [3.2], y considerando la numeración mostrada en la Figura 3.6, se presentan las dimensiones calculadas de cada pieza articulada correspondiente a los dedos de la mano. De acuerdo con la disposición geométrica de la Figura 3.5, se toma como referencia el cateto 1 como

el área “vacía” entre cada conjunto de piezas.

En el caso de la Tabla [3.1], las medidas corresponden a las piezas 2 y 3 de la Figura 3.5, donde la pieza 2 representa la hipotenusa. Por su parte, la Tabla [3.2] muestra las dimensiones de las piezas 5 y 6, considerando a la pieza 5 como la hipotenusa.

Estas medidas fueron obtenidas utilizando un pie de metro digital, tomando las distancias entre los puntos medios de cada sección del dedo respectivo tanto en posición extendida como flexionada, asegurando así que las proporciones respetaran la relación pitagórica en ambos estados del movimiento.

TABLA 3. 1 Proporciones Pitágoras Piezas 2 y 3

Dedo	Cateto 1(mm)	Cateto 2(mm)	Hipotenusa(mm)
Indice Derecho	21.55	20.15	29.5
Corazón Derecho	27.12	21.32	34.5
Anular Derecho	24.12	19.32	30.9
Meñique Derecho	16.32	8.92	18.6

TABLA 3. 2 Proporciones Pitagóricas Piezas 5 y 6

Dedo	Cateto 1(mm)	Cateto 2(mm)	Hipotenusa(mm)
Indice Derecho	25.17	24.32	35
Corazón Derecho	25.17	24.32	35
Anular Derecho	30.72	27.15	41
Meñique Derecho	27.37	19.323	33.5

3.2.2 Diseño del sistema electrónico del dispositivo.

Considerando que la cantidad de sensores requeridos por el sistema podría incrementarse en futuras versiones del dispositivo, se decidió incorporar el módulo demultiplexor 74HC4067 como componente intermediario entre la placa de desarrollo Arduino y los sensores analógicos. Esta decisión se fundamenta en la limitación del número de pines analógicos disponibles en la placa Arduino, que asciende a solo 7 canales de entrada, cifra insuficiente para cubrir los requisitos mínimos

del sistema definido durante la etapa de diseño.

El uso del demultiplexor permite ampliar significativamente la cantidad de entradas disponibles, alcanzando hasta 22 canales analógicos. Esta expansión no solo satisface las necesidades actuales del sistema, sino que también otorga un margen adicional que facilita la integración de nuevos sensores o módulos en futuras iteraciones del proyecto. Además, esta solución mantiene una arquitectura eficiente en cuanto a utilización de pines, evitando la necesidad de recurrir a placas de mayor tamaño o complejidad, lo cual resulta ventajoso en términos de espacio físico y consumo energético.

En la figura 3.6 se puede ver de una manera simplificada la conexión entre los componentes que integran el guante háptico, en el que se puede ver como las señales de cada potenciómetro pasan en primer lugar por el multiplexor, el cual, mediante una señal por parte del Arduino, selecciona que canal estará habilitado, y en consecuencia se envía la medición del potenciómetro correspondiente al Arduino.

A partir del diagrama se puede determinar los pines a utilizar de cada dispositivo, así como su disposición de conexión entre ellos.

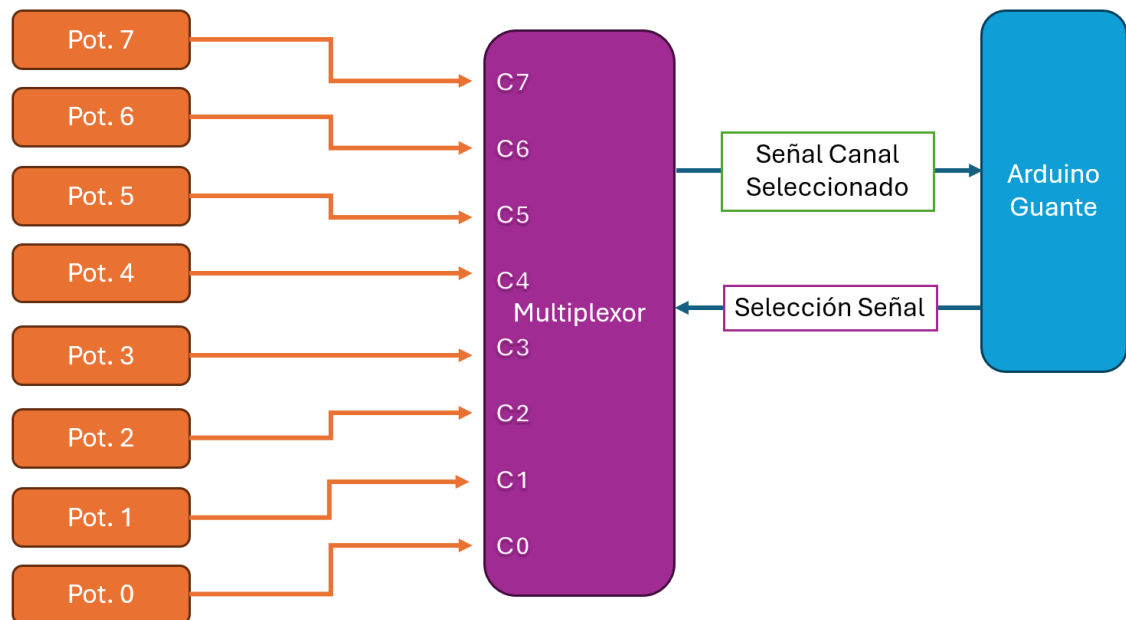


Fig. 3. 6 Diagrama de conexión de los dispositivos y sensores del Guante

La distribución de pines en la tabla [3.3] entre Arduino y el multiplexor 74HC4067, permite que la información de los sensores conectados al módulo 74HC4067 converja en el pin A0 de Arduino,

permitiendo a este último asignar la información de cada ángulo de forma interna y ordenada. Posteriormente pasa a la lectura del siguiente sensor, de acuerdo con las entradas selectoras S0, S1, S2, S3. Mediante la selección del canal.

TABLA 3. 3 Conexión Pines 74HC4067 y Arduino

Pin 74HC4067	Pin Arduino
GND	GND
VCC	3.3V
EN	GND
S0	D12
S1	D11
S2	D10
S3	D9
SIG	A0

Para garantizar el correcto funcionamiento de los 8 potenciómetros utilizados en el dispositivo, se estableció que los pines 1 y 3 de cada uno —correspondientes a VCC y GND, respectivamente se conectan en paralelo, es decir, todos comparten las mismas líneas de alimentación y tierra. La única conexión diferenciada corresponde al pin 2, el cual entrega la señal analógica medida. Este pin se conecta individualmente a uno de los canales de entrada del módulo demultiplexor 74HC4067. De esta manera, el pin 2 del potenciómetro n se conecta al canal n del demultiplexor, iniciando con el primer potenciómetro conectado al canal 0 y continuando secuencialmente hasta el octavo sensor, asignado al canal 7. Esta organización permite gestionar múltiples entradas analógicas mediante una única entrada del microcontrolador, esta disposición queda mostrada en la tabla [3.4].

TABLA 3. 4 Conexión entre un potenciómetro y 74HC4067

Potenciómetro “n”	74HC4067
Pin 1	VCC
Pin 2	C#
Pin 3	GND

El esquemático mostrado en la figura 3.7 representa las conexiones eléctricas entre los principales componentes del guante, detallando cómo se interrelacionan según lo especificado en las tablas [3.3] y [3.4]. En el diagrama también se incluye la conexión de una fuente de alimentación externa, como una batería, que permite la operación autónoma del sistema. Dado que el guante está diseñado para ser un dispositivo portátil y utilizarse directamente sobre el cuerpo del usuario, el uso de una fuente cableada resultaría poco práctico.

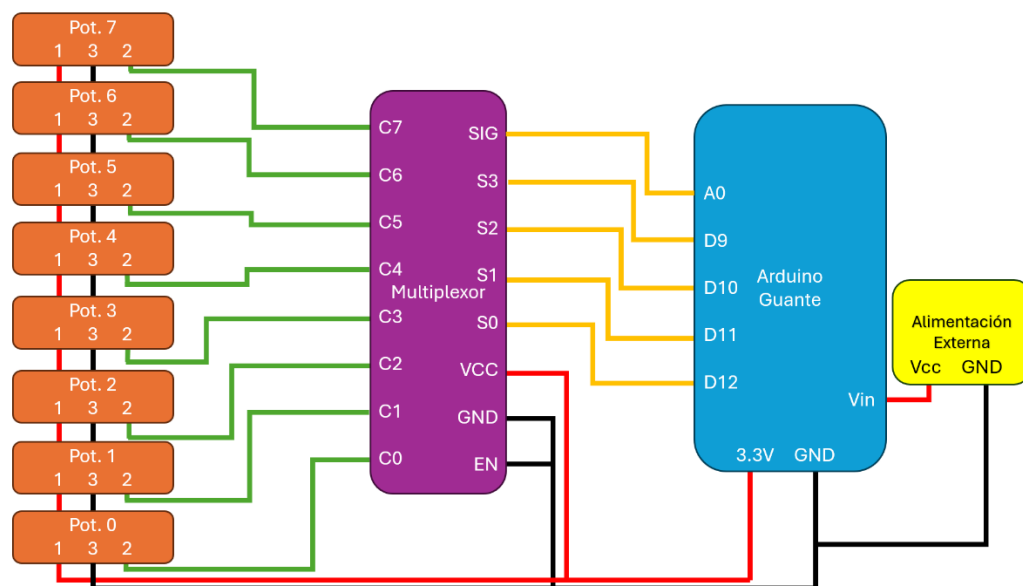


Fig. 3. 7 Esquemático Sistema Guante Háptico

En la figura 3.8 se presenta el diseño de la placa PCB desarrollada para el sistema electrónico del guante. A la izquierda se muestra el trazado de las pistas en el software flux.ai, mientras que a la derecha se visualiza una representación 3D de los componentes montados.

En este diseño se observa el uso de conectores tipo JST en el lateral izquierdo, destinados a la conexión de los 8 potenciómetros que actúan como sensores del guante. En la sección central se encuentra el módulo multiplexor 74HC4067, encargado de seleccionar cada uno de los canales. Finalmente, en el lado derecho se aprecia el espacio reservado para el Arduino Nano ESP32, el cual será montado sobre pines socket para facilitar su inserción y remoción. En la parte superior se añadió un conector destinado a la alimentación del circuito.

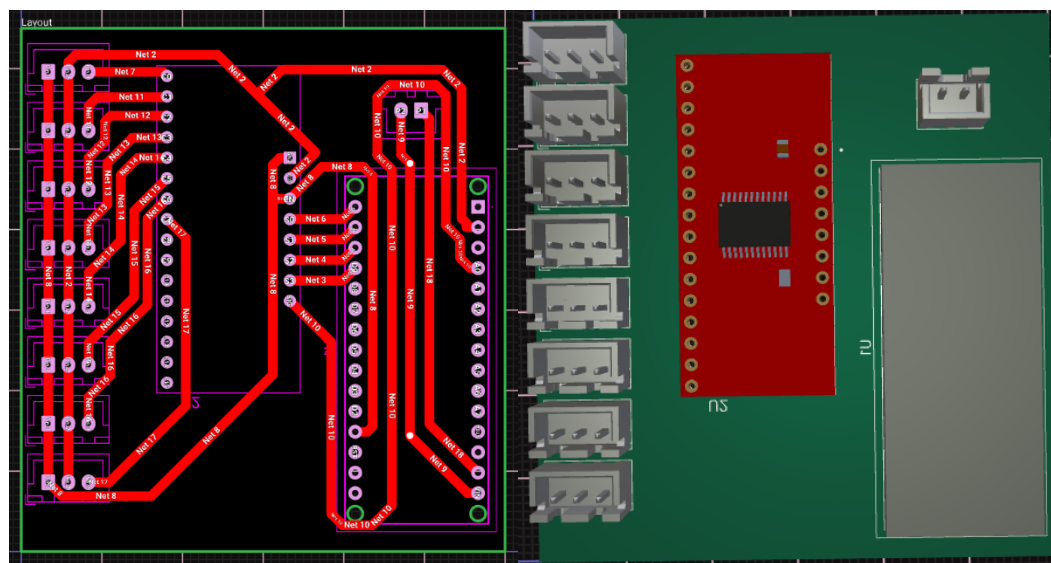


Fig. 3. 8 Circuito PCB del Guante Háptico

3.3. Entorno simulado

Para validar el funcionamiento del guante háptico, es necesario definir un entorno de pruebas que simulen las condiciones de un escenario de domótica realista. En esta etapa, se analizaron los diversos dispositivos y actuadores que se encuentran normalmente en un ambiente de domótica, así como la simplicidad de la realización de un modelo a escala de estos dispositivos, y su viabilidad eléctrica y de control para ser controlados por un microcontrolador Arduino.

El entorno se diseñó incorporando dispositivos representativos de un hogar inteligente, específicamente, iluminación, ventilación, cerradura, apertura de puerta y cortina, como también un sistema sonoro de notificaciones y/o alarma.

3.3.1 Diseño Estructura Física del Entorno de Pruebas

Dado que el entorno de pruebas se realizará a una escala personalizada, fue necesario diseñar cada una de sus partes. Para ello, se optó por el modelado en 3D de cada componente, con el objetivo de realizar posteriormente su impresión en PLA.

En la figura 3. 9, puede observarse el diseño de una cortina a escala, de tipo roller, a utilizar

como parte del entorno de pruebas para la simulación de un sistema de control de apertura y cierre automatizado. Este tipo de cortina se seleccionó por su simplicidad mecánica.

Las cortinas tipo roller poseen un mecanismo de funcionamiento de enrollado y desenrollado del material sobre un eje giratorio, lo cual permite un control suave y controlado, ideal para ser controlado mediante un motor DC.



Fig. 3. 9 Cortina a escala tipo roller

En la figura 3. 10 se observa el modelo 3D de una puerta a escala, diseñada específicamente para aplicarlo en el entorno simulado de pruebas. Esta estructura tiene por objetivo replicar de manera simplificada el funcionamiento de una puerta convencional, permitiendo validar la funcionalidad del sistema para la apertura y cierre mediante una cerradura eléctrica.

Consta de un sistema de bisagras y un tirador para permitir la movilidad de una puerta convencional, lo cual permitirá comprobar de manera óptima el funcionamiento de la cerradura eléctrica.



Fig. 3. 10 Puerta

En la figura 3. 11 se muestra el diseño de una caja a escala, desarrollada para contener en su interior una corrida de LEDs RGB, con el objetivo de simular un sistema de iluminación inteligente dentro del entorno de pruebas. Esta estructura actúa como una carcasa de protección y difusor de luz, permitiendo una mejor visualización de los colores proyectados, además de un diseño similar a sistemas de iluminación reales.

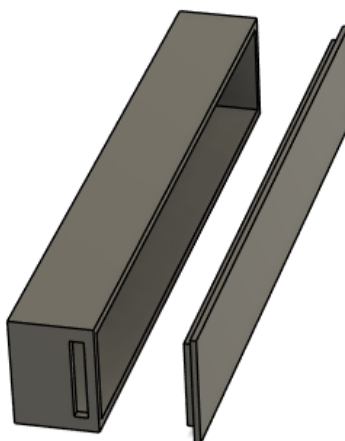


Fig. 3. 11 Contenedor Leds RGB

En la figura 3. 12 se presenta el diseño de un soporte destinado al montaje de un buzzer pasivo para la utilización del entorno de pruebas. La estructura a sido modelada para ajustarse con precisión al diámetro del buzzer, manteniéndolo firme en su posición durante las pruebas y permitiendo una

salida libre para el sonido generado. Además, cuenta con una abertura lateral para facilitar la conexión de los cables de alimentación.

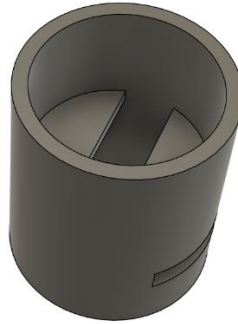


Fig. 3. 12 Soporte Buzzer

En la figura 3. 13 se presenta el diseño del soporte para un ventilador para el entorno simulado representa un sistema de ventilación o climatización. La estructura que se presenta en el diseño es un marco abierto que permite una fácil sujeción del ventilador a utilizar en una posición fija, así como permitir una circulación más eficiente del aire sin obstrucciones manteniendo una estabilidad estructural.

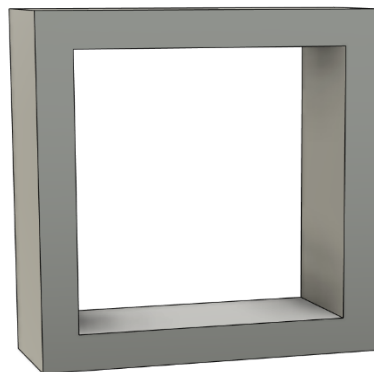


Fig. 3. 13 Soporte Ventilador

3.3.2 Diseño del sistema electrónico del Entorno.

En la figura 3. 14 se ilustra el esquema de conexión entre el microcontrolador Arduino y los distintos actuadores que conforman el entorno físico simulado. El diseño del sistema establece una

relación de control unidireccional, en el cual el Arduino emite señales de salida hacia estos actuadores sin requerir algún tipo de retroalimentación directa desde estos. Esto se debe a que las funciones asignadas a cada actuador no requieren una comprobación de estado en la etapa de validación.

Con el diagrama establecido se pueden seleccionar los pines de conexión entre el Arduino y los distintos Actuadores.

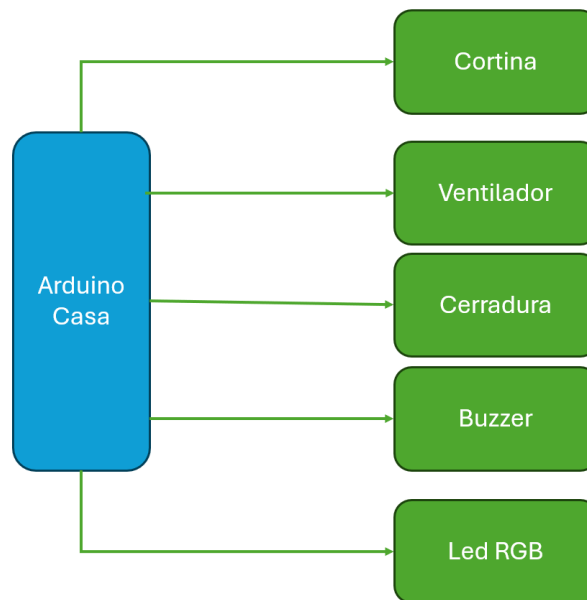


Fig. 3. 14 Conexión entre Arduino y los Actuadores del entorno simulado

Para el caso de la cortina, debido a que se utilizó un motor de tipo DC para el control de enrollado y desenrollado de la cortina, la mejor forma de lograr esto de manera directa desde un Arduino es mediante el uso de puentes H, cuya conexión está representada en la tabla [3.5], ya que estos además de permitir controlar el sentido de giro del motor, permiten el control de velocidad por medio de PWM. En este proyecto se decidió utilizar los pines A1 y A2 del Arduino, para controlar el sentido de giro del motor, así como el pin A3 para encender o apagar el funcionamiento del puente H, en el caso en que la cortina esté en reposo.

TABLA 3. 5 Conexión Pines Arduino y Puente H L293D

Pin Arduino	Pin L293D
GND	GND
VCC	Vcc1
A1	Input 1
A2	Input 2
A3	Enable 1,2

Una vez configurada la conexión entre el módulo de puente H (L293D) y el Arduino, la conexión del motor DC y el Puente H L293D (ver tabla [3.6]) corresponde a la conexión de las entradas de alimentación del motor con los pines de salida (Output 1 y 2) del módulo L293D, que se encargan de invertir su polaridad según lo requerido para el estado actual de la cortina. Para este caso se seleccionó que VCC y GND del motor DC, estén conectados a Output 1 y Output 2 respectivamente del módulo.

TABLA 3. 6 Conexión Pines Puente H L293D y motor DC

Pin motor DC	Pin L293D
VCC	Output 1
GND	Output 2

Para el caso de la cerradura electrónica implementada en el entorno de pruebas, se requiere una fuente externa de 12V, ya que los módulos disponibles en el laboratorio operan a ese voltaje. Debido a que el Arduino no puede entregar directamente dicho voltaje, se utiliza un relé como intermediario. Esto permite controlar el encendido y apagado de la cerradura desde el Arduino, al mismo tiempo que se aísla el circuito de 12V, evitando posibles daños por diferencias de tensión.

La conexión entre el relé y el Arduino se realiza utilizando los pines de alimentación del módulo y el pin digital 3 del Arduino, encargado de activar o desactivar el relé, como se muestra en la Tabla 3. 7.

TABLA 3. 7 Conexión Pines Arduino y Relé

Pin Arduino	Pin Relé
VCC	VCC
GND	GND
3	SIG

La conexión entre el relé y la cerradura (ver tabla [3.8]) consta de dos conexiones, en donde se decidió dejar el GND de la cerradura en el pin común del Relé, y el VCC en el normal abierto. Esto permite que, al enviar una señal lógica alta al relé, se energice y active la cerradura. En caso de requerir que la cerradura permanezca desenergizada o cerrada, basta con mantener la señal de activación en estado bajo. Este método previene un desgaste excesivo del componente mecánico del relé.

TABLA 3. 8 Conexión Pines Cerradura y Relé

Relé	Cerradura
Normal Abierto	VCC
Común	GND

Para el caso de la iluminación los LED RGB disponibles son del tipo cátodo común, lo que requerirá, que el pin VCC común de los 3 colores esté conectado directamente a VCC del Arduino, mientras que R, G, B estarán conectados a los pines 4, 5, 6 respectivamente (ver tabla [3.9]), funcionando con lógica negativa. Es decir, cuando el Arduino tenga los pines en bajo, el led RGB encenderá el respectivo color. Se utiliza una sola tabla de conexión, ya que, aunque se utilicen más led RGB para el entorno simulado, estos estarán conectados en paralelo.

TABLA 3. 9 Conexión Pines Arduino y LED RGB

Arduino	LED RGB
VCC	VCC
4	R(GND)
5	G(GND)
6	B(GND)

Para el caso del sistema sonoro de notificaciones y/o alarma, se utilizará un buzzer pasivo, que puede ser controlador mediante pulsos PWM para realizar distintos tonos al estar activado. El GND del buzzer estará directamente conectado a GND de Arduino. El pin VCC (ver tabla [3.10]) que en el caso del Buzzer es el de activación estará conectado al pin 7 de Arduino para su activación según sea requerido.

TABLA 3. 10 Conexión Pines Arduino y Buzzer

Arduino	Buzzer
7	VCC
GND	GND

Para el caso del ventilador, se seleccionó un modelo que opera a 5 V, controlado mediante una señal digital proveniente del Arduino Nano ESP32. Dado que este modelo de Arduino solo proporciona 3.3 V desde sus salidas, se recurre al pin VIN para obtener los 5 V necesarios, alimentando directamente el ventilador desde la línea de alimentación del microcontrolador.

La activación del ventilador se logra mediante el uso de un transistor configurado como interruptor. En esta disposición, el colector del transistor se conecta directamente a VIN, el emisor a VCC del ventilador, y la base al pin digital número 2 del Arduino. Cuando el microcontrolador activa este pin, el transistor entra en conducción, permitiendo el paso de corriente desde VIN hacia el ventilador (ver tabla [3.11]).

TABLA 3. 11 Conexión Pines Arduino, Ventilador y Transistor

Arduino	Ventilador	Transistor
GND	GND	NaN
NaN	VCC	Emisor
VIN	NaN	Colector
2	NaN	Base

El esquemático presentado en la figura 3.15 muestra las conexiones eléctricas entre los principales componentes del entorno simulado, detallando cómo se interrelacionan según lo definido

en las tablas [3.5], [3.6], [3.7], [3.8], [3.9], [3.10] y [3.11]. El diagrama incluye la distribución de señales desde el microcontrolador hacia los distintos actuadores del sistema, como el ventilador, el buzzer, la cerradura electrónica, el LED RGB y el motor DC. Además, se representa la integración de dos fuentes de alimentación externas: una fuente de 5 V destinada a la alimentación del microcontrolador y de periféricos de baja potencia, y una fuente de 12 V utilizada exclusivamente para accionar la cerradura, la cual requiere un mayor nivel de tensión para su correcto funcionamiento.

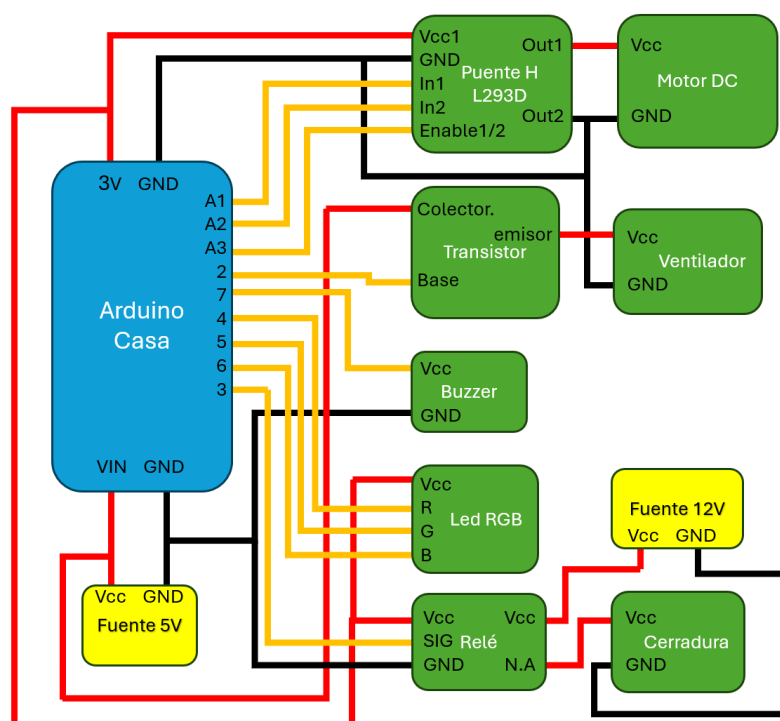


Fig. 3. 15 Esquemático del entorno simulado

En la figura 3. 16 se presenta el diseño del circuito impreso (PCB) correspondiente al sistema entorno simulado. En la parte izquierda de la imagen se puede observar en color rojo, las pistas de cobre las cuales están conectadas de acuerdo con las conexiones representadas en las tablas, y las líneas moradas representan el tamaño y disposición de los dispositivos.

En la parte derecha se puede visualizar una representación 3D de cómo se vería la placa PCB en físico, en donde el recuadro gris representa el Arduino nano ESP32, y los conectores JST las conexiones a los distintos actuadores.

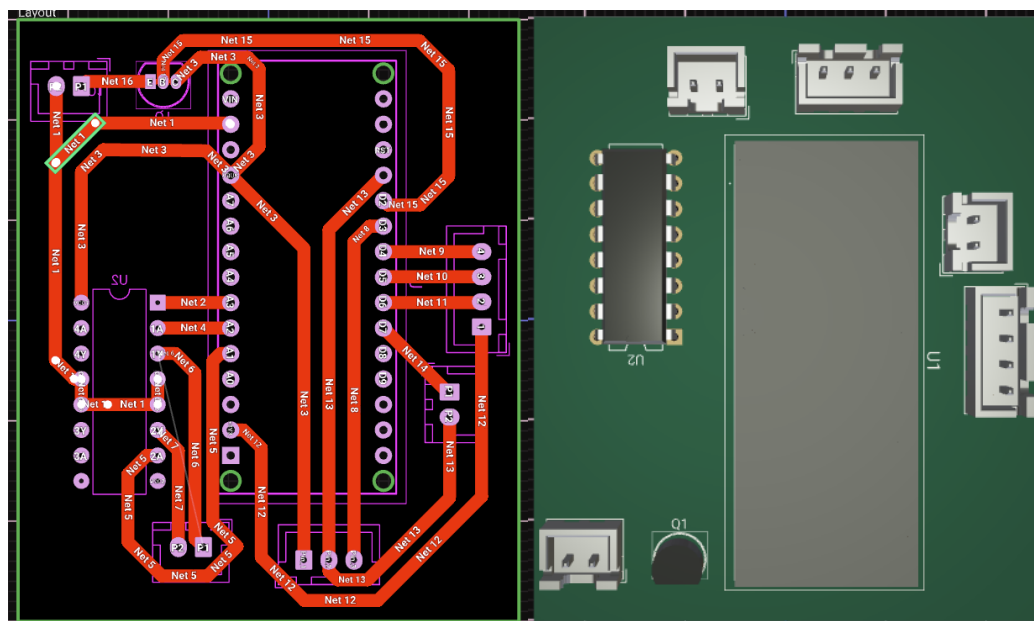


Fig. 3. 16 Circuito PCB del entorno simulado

3.4. Servidor Central

Para lograr la integración y coordinación entre el guante háptico y el entorno simulado, es necesario definir un servidor central, que actúe como núcleo lógico del sistema. Este servidor tiene como propósito principal gestionar las comunicaciones entre los distintos módulos del proyecto.

En esta etapa, se consideraron distintos factores como la compatibilidad con protocolos de comunicación, la estabilidad del sistema y la capacidad de procesamiento. Se optó por implementar una Raspberry Pi 5 como base física del servidor, dada su versatilidad, reducido tamaño y eficiencia energética. Este servidor permite ejecutar de forma local un sistema de automatización doméstica mediante el software Home Assistant, el cual facilita la integración de dispositivos y el desarrollo de automatizaciones personalizadas.

Con el objetivo de aprovechar al máximo las capacidades del servidor central basado en Home Assistant, se consideró la integración de distintos complementos esenciales que permiten ampliar su funcionalidad y adaptarlo a los requerimientos del sistema desarrollado. Entre estos, se encuentran herramientas clave como un gestor de dominios personalizado, que permite el acceso remoto y seguro mediante un subdominio dinámico; un bróker MQTT fundamental para la comunicación entre dispositivos IoT de manera ligera y eficiente; un visor de registros, indispensable para el monitoreo y detección de errores en tiempo real; y finalmente un editor de código que permita editar y gestionar

la configuración del sistema directamente desde el navegador con una experiencia sencilla y fácil de utilizar.

3.4.1 Selección de complementos para Home Assistant.

Para poder acceder a la plataforma de Home Assistant de manera remota se utilizó DuckDNS, el cual es un complemento utilizado en Home Assistant para la gestión de nombres de dominio dinámico (DDNS). Este servicio permite asignar un nombre de dominio personalizado al servidor central, facilitando el acceso remoto, además de mantener una conexión estable que permite acceder al sistema de manera segura sin comprometer la integridad de los datos o la privacidad del usuario.

Para el correcto funcionamiento fue necesario realizar configuraciones específicas en el router de red utilizado, tales como la habilitación del reenvío de puertos (port forwarding) hacia el dispositivo en el que se ejecuta Home Assistant. Esto significó reasignar los puertos de entrada que utiliza DuckDNS, el cual de manera fija utiliza el 422, mientras que home assistant se aloja en el puerto 8123. Finalmente, para evitar que el router cambiara la ip local del dispositivo se decidió asignarle una IP fija con la estructura de 192.168.0.99 para evitar desconexiones en caso de reinicio de alguno de estos dispositivos.

Se utilizó el protocolo Mosquitto, un complemento para Home Assistant que implementa un bróker MQTT (Message Queuing Telemetry Transport), que permite el envío y recepción de datos de manera ligera entre el servidor central y dispositivos que tengan conectividad IoT. Es un protocolo basado en un modelo de publicador/subscriptor, lo que lo hace ideal para el control de actuadores por su eficiencia en conexiones inestables, pudiendo mantener una alta frecuencia en la comunicación de los datos.

La implementación del broker MQTT mediante Mosquitto permite establecer un canal de comunicación entre el guante háptico y el servidor central por el modelo de suscripción y publicación. Para su correcto funcionamiento es necesario la creación de usuarios con sus propias contraseñas y puertos seguros de acceso. En este caso como se disponía de conectarse a 2 dispositivos, el guante, y el entorno simulado se creó dentro de Home Assistant al usuario Mosquitto y Mosquitto2 respectivamente para cada uno de los dispositivos. Para la comunicación, se habilitó el puerto 1883 para las conexiones entre el guante, el servidor central y el entorno simulado, para mantener una comunicación bidireccional.

Para una mejor visualización y debug de la plataforma se implementó el complemento Log Viewer, de Home Assistant que permite visualizar de manera estructurada y en tiempo real los registros del sistema. Este complemento es especialmente útil para realizar pruebas para las conexiones y automatizaciones del sistema, ya que proporciona de una manera sencilla y directa sobre los errores y conexiones que tiene el sistema en tiempo real. Esto permite realizar un seguimiento detallado del comportamiento del sistema, identificar los fallos de comunicación, así como validar el correcto funcionamiento de los scripts.

Para poder programar las configuraciones y automatizaciones del servidor se utilizó el complemento Studio Code Server, una extensión para el desarrollo de configuraciones de Home Assistant. Este complemento proporciona una interfaz del editor Visual Studio Code directamente desde el sistema, permitiendo editar archivos del entorno, como lo son `configuration.yaml`, `automations.yaml`, `scripts` y otros elementos del sistema de manera remota y eficiente.

El uso resulta clave en proyectos personalizados como el de la presente tesis, ya que centraliza el acceso y edición del sistema desde cualquier navegador, sin necesidad de herramientas adicionales externas. En particular la disponibilidad de Studio Code Server permitió realizar ajustes rápidos, pruebas y mantenimientos sin la necesidad de acceder físicamente al dispositivo.

En la Figura 3.17 se ilustra la interacción entre los principales complementos integrados en Home Assistant. Se representa gráficamente cómo cada complemento se comunica con la plataforma, así como las interacciones que ocurren entre ellos. Además, se muestra cómo algunos de estos complementos envían información en paralelo al visor de registros (*Log Viewer*), lo cual permite la supervisión del sistema en tiempo real durante su operación.

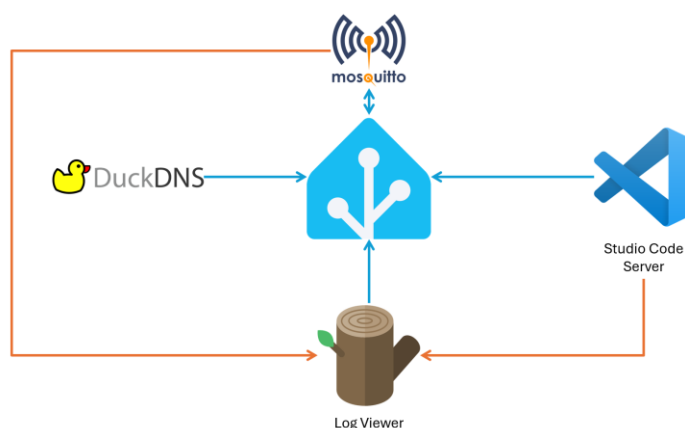


Fig. 3. 17 Representación interacción Home Assistant con complementos

Capítulo 4. Implementación de los sistemas

4.1. Introducción

En este capítulo se detalla el proceso de implementación física y funcional de los tres sistemas diseñados a lo largo del proyecto: el guante háptico, el entorno simulado y el servidor central. La finalidad de esta etapa es integrar todos los componentes electrónicos, mecánicos y de software previamente definidos, con el objetivo de validar su operatividad bajo condiciones más cercanas a las reales y su capacidad para interactuar entre sí de manera eficiente.

La implementación del guante háptico considera el montaje de los sensores en la estructura ergonómica previamente diseñada para ser capaz de capturar con precisión los movimientos de la mano, los cuales serán utilizados como los comandos de entrada para el control del entorno simulado. Por su parte, el entorno simulado incluye una serie de dispositivos domóticos representativos, como los ya mencionados, cortina, cerradura, sistema de iluminación, ventilación y alertas todos adaptados a escala, y montados en la casa a escala ubicada en el Laboratorio de Redes y Sistemas Inteligentes de la UCSC.

Finalmente, el servidor central basado en una Raspberry Pi ejecutando Home Assistant, se configura para recibir, interpretar y gestionar las señales provenientes del guante, así como para coordinar las acciones de los actuadores presentes en el entorno.

Esta sección describe paso a paso la conexión física de los dispositivos, la configuración de software y red, y las pruebas iniciales para permitir asegurar la correcta funcionalidad del sistema global.

4.2. Guante

La implementación del guante háptico constituye una etapa fundamental del proyecto, ya que representa el dispositivo principal de interacción entre el usuario y el entorno inteligente, además como el objetivo principal y central propuesto al comienzo de este proyecto. Esta fase abarca desde la fabricación física del guante hasta su integración funcional con el servidor central, permitiendo la lectura y transmisión de datos en tiempo real a una alta tasa de datos.

4.2.1 Impresión de Placa PCB

En la figura 4.1 se muestra el modelo de la placa PCB correspondiente al guante háptico, visualizado en el software Flatcam. Este diseño fue utilizado para generar los archivos necesarios para su fabricación mediante una máquina CNC.

La imagen representa el circuito encargado de conectar los sensores del guante, permitiendo la lectura de los potenciómetros a través del multiplexor y su envío al microcontrolador. Las líneas azules delgadas indican las trayectorias que la máquina CNC seguirá para eliminar cobre superficial y definir las pistas del circuito. Los puntos rojos corresponden a perforaciones para pines y conectores, mientras que las líneas semitransparentes verdes representan el recorrido lógico para realizar dichas perforaciones con precisión.

Finalmente, la línea azul exterior más gruesa delimita el contorno que será recortado para separar la placa del resto del material, obteniendo así un formato compacto. Esta placa será ensamblada junto con los elementos físicos del guante háptico, integrándose al sistema general y permitiendo su comunicación con el servidor central.

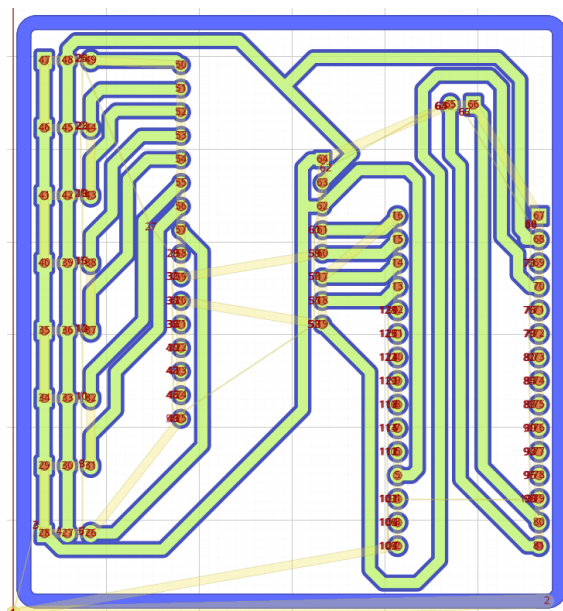


Fig. 4. 1 Modelo PCB Guante en Flatcam

En la figura 4.2 se presenta la placa PCB ya fabricada y ensamblada, visualizada desde ambas caras para apreciar su estructura completa. A la izquierda se observa la parte inferior de la placa, en

donde se encuentran las pistas de cobre fresadas mediante la máquina CNC, las cuales establecen las conexiones eléctricas entre los distintos componentes del sistema. A la derecha, se muestra la cara superior de la PCB, donde se han montado los conectores tipo JST y los pines tipo socket.

Los conectores JST, visibles alineados en la parte izquierda de la imagen derecha, permiten establecer una conexión segura y ordenada con los sensores que se integrarán al guante háptico. Por otro lado, los pines socket —ubicados al centro y la derecha— han sido incorporados con el objetivo de facilitar el montaje y desmontaje del Arduino Nano ESP32 y del módulo multiplexor 74HC4067, sin necesidad de soldaduras permanentes.

Esta configuración modular, reflejada en la disposición ordenada de los componentes sobre la placa, no solo favorece la organización y mantenimiento del sistema, sino que también permite su reutilización y adaptación progresiva durante el desarrollo del prototipo.

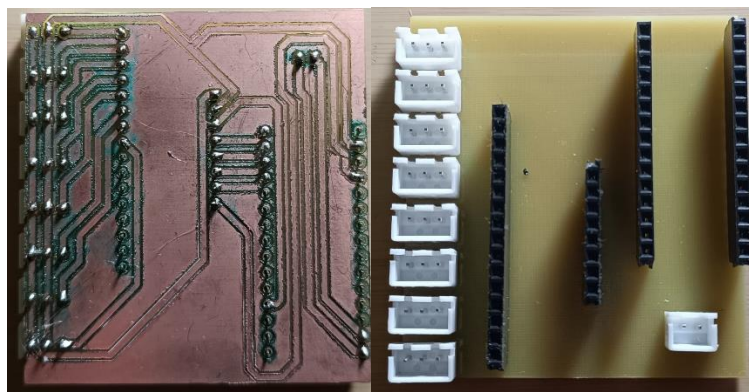


Fig. 4. 2 Imagen de la PCB con conectores JST y Pin Socket Guante

4.2.2 Montaje del guante.

En la Figura 4. 3 se muestra el montaje de los componentes físicos del guante. En el lado derecho se observa la estructura impresa en 3D que se ajusta a la mano, donde están montados los potenciómetros ubicados en las secciones correspondientes a las articulaciones que se deben medir. Estos potenciómetros están conectados mediante cables a la parte izquierda de la estructura, que corresponde al soporte fijado al brazo. Esta sección cuenta con un espacio destinado al montaje de la placa PCB y las baterías necesarias para energizar el circuito.

En la placa PCB se logra apreciar cómo van ubicados el Arduino Nano ESP32, y el

multiplexor, así como la conexión de la placa con las baterías, que cuenta con un interruptor para encender y apagar el dispositivo.

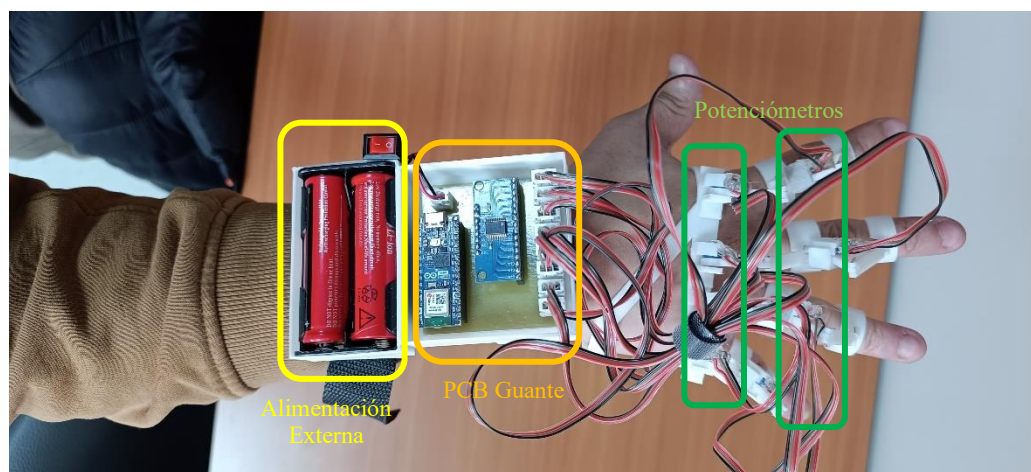


Fig. 4. 3 Montaje del guante con sus componentes

4.2.3 Interacción entre guante y servidor central

En la figura 4.4, se puede observar el esquema de conexión e interacción del guante háptico con el servidor central, mediante una conexión inalámbrica a internet mediante Wi-Fi representada por una línea punteada verde, en el cual por medio de MQTT publica desde arduino los estados actuales de cada potenciómetro, que son escuchados por Home Assistant.

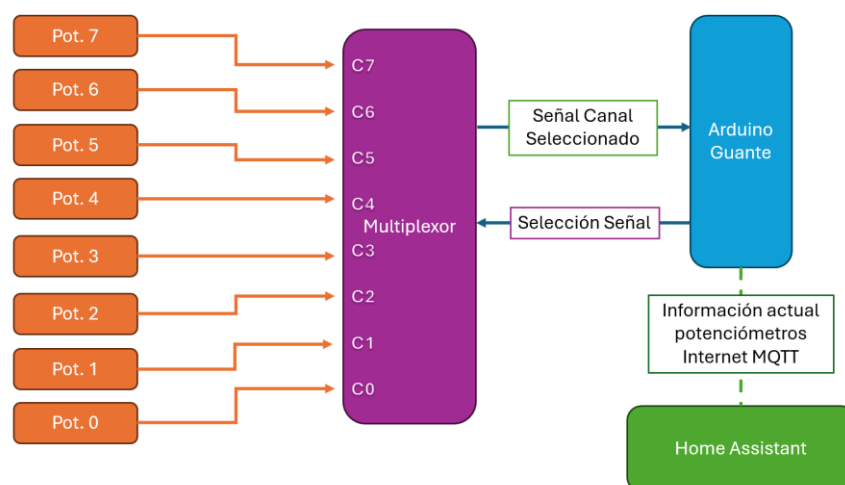


Fig. 4. 4 Interacción entre guante y servidor usando Home Assistant

En la figura 4.5, se puede observar el funcionamiento lógico interno que realiza Arduino para que opere el guante, en donde el recuadro verde indica el encendido o inicio de la lógica del dispositivo, los recuadros azules representan el funcionamiento interno del Arduino, y las líneas punteadas y el color naranja representan la interacción con el servidor central de Home Assistant por medio de MQTT.

En un inicio el Arduino realiza dos tareas de inicialización, en el cual realiza una conexión a la red de internet por medio de Wifi, seguido de iniciar el protocolo MQTT con sus debidas configuraciones de usuario y dirección web para su correcta vinculación y comunicación con el servidor central.

Por el lado del funcionamiento interno inicializa los pines utilizados para su comunicación con el multiplexor y los potenciómetros, los cuales una vez inicializados proceden a realizar una calibración inicial. Mediante un indicador LED RGB integrado en el Arduino Nano ESP32, se guía a la persona que porta el guante durante el proceso de calibración: el color rojo indica que debe mantener las articulaciones de la mano completamente extendidas para registrar el valor máximo de referencia, mientras que el color verde señala que debe flexionar todas las articulaciones de los dedos para registrar los valores mínimos correspondientes. Con estas mediciones se procede a tomar como un 0, las medidas mínimas de cada potenciómetro, y 90 las medidas máximas, para una mejor visualización de los 90 grados que tiene de libertad cada articulación.

Una vez calibrado cada potenciómetro se procede a seleccionar un canal del demultiplexor asignado a cada potenciómetro. Luego se toman 10 mediciones de cada uno para limpiar el posible

ruido que estas mediciones puedan tener, usando método de la mediana para entregar un solo valor.

Una vez que se obtiene el valor sin ruido de cada potenciómetro se procede a comprobar la conexión del guante a internet y al servidor central. En caso de estar conectados se enviarán los valores actuales al servidor central. En caso contrario se pasará por una función de reconexión con internet y el servidor central, para posteriormente enviar los valores en tiempo real a servidor.

En todo momento el guante entra dentro de un bucle en donde regresa a seleccionar los canales y lectura de potenciómetros como se muestra en la figura.

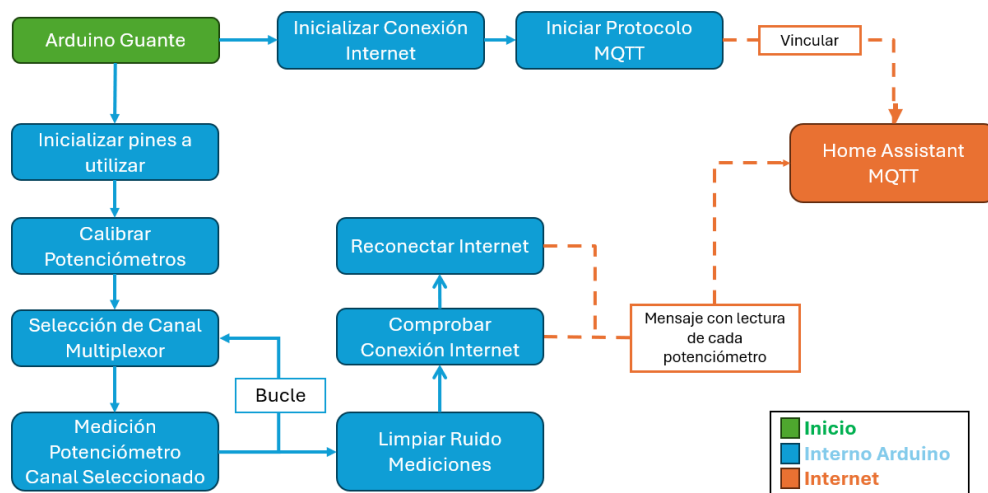


Fig. 4. 5 Funcionamiento interno del Guante

4.2.4 Selección de acciones guante

La Tabla 4.1 presenta la asignación de acciones asociadas a cada potenciómetro integrado en el guante háptico, en función de la posición del dedo correspondiente (estirado o flexionado). Cada potenciómetro está vinculado a una articulación específica, y su lectura permite identificar el gesto realizado por el usuario.

Las acciones definidas cubren distintas funcionalidades del sistema domótico simulado: control de la cortina automatizada (potenciómetro 0), activación del ventilador (potenciómetro 1), apertura o cierre de la cerradura electrónica (potenciómetro 2), y encendido o apagado de la señal sonora de alerta (potenciómetro 3). A su vez, los potenciómetros 4, 5 y 6 están destinados al encendido y apagado de luces RGB de color rojo, verde y azul respectivamente, permitiendo así simular

escenarios de iluminación personalizados. El potenciómetro 7 no fue asignado a ninguna acción específica en esta versión del prototipo, quedando reservado para futuras expansiones o configuraciones del sistema.

TABLA 4. 1 Acciones según posición de los dedos

Dedo	Dedo Estirado	Dedo Flexionado
Potenciómetro 0	Abrir cortina	Cerrar cortina
Potenciómetro 1	Encender ventilador	Apagar ventilador
Potenciómetro 2	Abrir cerradura	Cerrar cerradura
Potenciómetro 3	Activar Sonido	Desactivar sonido
Potenciómetro 4	Prender luz roja	Apagar luz roja
Potenciómetro 5	Prender luz verde	Apagar luz verde
Potenciómetro 6	Prender luz azul	Apagar luz azul
Potenciómetro 7	NaN	NaN

4.3. Entorno Simulado

En esta sección se detalla el proceso de implementación física del entorno simulado, el cual representa un sistema domótico a escala. Se abordan las etapas de fabricación de los dispositivos, el ensamblaje de los componentes electrónicos y estructurales, y su conexión con el sistema de control central. Esta implementación tiene como propósito validar el funcionamiento del sistema completo a partir de las órdenes generadas por el guante sensorizado y gestionadas a través del servidor central.

4.3.1 Impresión de Placa PCB

El proceso de elaboración de la placa PCB correspondiente a este sistema contempla su fabricación mediante una máquina Router CNC. Dicho circuito fue diseñado con el objetivo de maximizar el aprovechamiento del espacio disponible y facilitar una distribución ordenada de las conexiones. Una vez completado el fresado del diseño, se procederá al montaje de los distintos

componentes, incorporando conectores específicos en función de los requerimientos eléctricos y estructurales de cada sección del sistema.

En la figura 4. 6, se puede ver la visualización del modelo diseñado previamente para la placa PCB en el software de Flatcam, el cual se utilizó para la generación de los archivos necesarios para la fabricación mediante la máquina CNC. El diseño mostrado corresponde al circuito desarrollado previamente en la aplicación web de Flux.

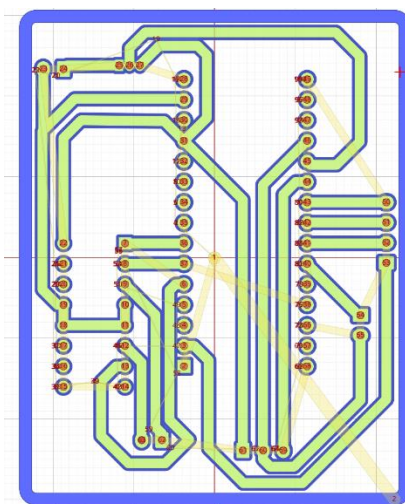


Fig. 4. 6 Modelo PCB en Flatcam

En la figura 4. 7 se presenta la placa PCB ya fabricada y ensamblada, visualizada desde ambas caras para apreciar su estructura completa. A la izquierda se observa la parte inferior de la placa, en donde se encuentran las pistas de cobre responsable de la comunicación entre los pines de los dispositivos.

En la imagen de la derecha se pueden ver los conectores tipo JST para las conexiones de los distintos actuadores, así como el módulo L293D, y el transistor 2n2222 que se decidieron soldar directamente en la PCB. Se logran apreciar los pines Socket para la conexión del Arduino ESP32 Nano.

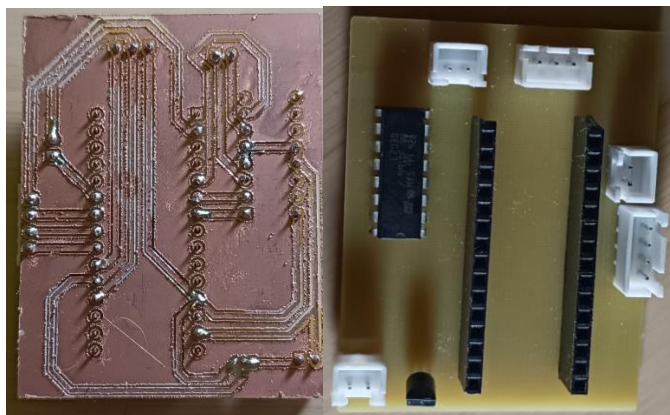


Fig. 4. 7 Imagen de la PCB con conectores JST y Pin Socket Entorno Simulado

En la figura 4. 8, se pueden ver los componentes del entorno simulados montados en la casa a escala ubicadas en el Laboratorio de Redes y Sistemas Inteligentes de la UCSC. En la parte superior, de izquierda a derecha se pueden identificar, el buzzer, la cortina y los leds RGB, en la parte inferior derecha se puede apreciar la cerradura electrónica junto a la puerta impresa en 3D. En la figura 4. 9 puede visualizarse con la cortina abierta la ubicación del ventilador.

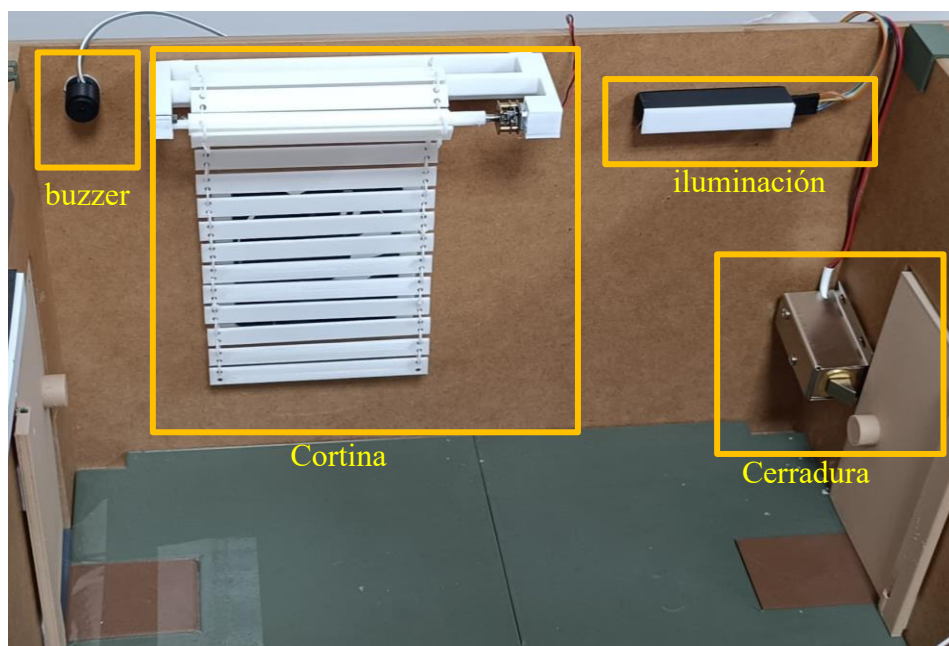


Fig. 4. 8 Entorno simulado implementado Cortina Cerrada



Fig. 4. 9 Entorno simulado con ventilador visible

En la figura 4. 10 se observa la parte posterior de la pared mostrada previamente en la figura 4. 8 y 4. 9. Se muestra la caja de conexiones del entorno simulado, la cual alberga en su interior la fuente de alimentación de 12V necesaria para el funcionamiento de la cerradura electrónica, así como la PCB encargada de gestionar los actuadores. En esta placa se encuentra conectado el microcontrolador, el cual emite las señales de control correspondientes a cada uno de los actuadores del sistema.



Fig. 4. 10 Conexiones eléctricas entorno simulado

4.3.2 Interacción entre el sistema domótico y servidor central

En la figura 4. 11, se puede observar el esquema de conexión e interacción del sistema con el servidor central mediante comunicación MQTT, en donde se puede apreciar la conexión inalámbrica mediante Wi-Fi por parte de Home Assistant con el Arduino mediante líneas puntuadas. Esta comunicación consiste en el envío de los valores booleanos de los potenciómetros del guante, los cuales de acuerdo con la señal de cada potenciómetro serán enviadas instrucciones a cada actuador.

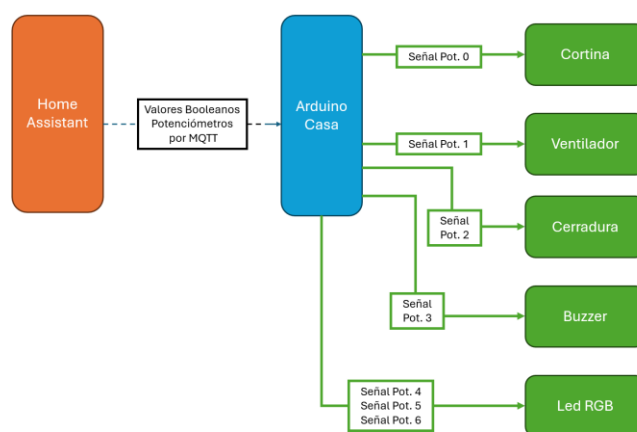


Fig. 4. 11 Interacción entre sistema domótico y servidor usando Home Assistant

En la figura 4. 12, se presenta el funcionamiento lógico interno que lleva a cabo el microcontrolador Arduino para operar el entorno simulado. En el diagrama, el recuadro verde representa el punto de inicio del sistema, mientras que los recuadros celestes detallan los procesos internos realizados por Arduino. Por su parte, los elementos en color naranja y las líneas punteadas indican la comunicación con el servidor central Home Assistant a través del protocolo MQTT.

El flujo comienza con la inicialización de los pines que se utilizarán para la comunicación entre el Arduino y los actuadores, seguido de la conexión a internet del dispositivo mediante Wi-Fi, para posteriormente iniciar el protocolo MQTT en modo escucha de las publicaciones de Home Assistant sobre los valores booleanos de los potenciómetros.

Una vez iniciado este proceso el Arduino entra en un estado de espera de mensajes por parte de Home Assistant, el cual al recibir el mensaje sobre el estado lógico de alguno de los valores booleanos este envía el valor correspondiente que ha recibido el cambio a la función de control correspondiente.

En el caso de la función encargada del control de la cortina este funciona en caso de recibir una señal de ON, enrollar la cortina por un tiempo determinado para abrirla, mediante la conexión entre el módulo L293D y el motor DC conectado a la misma, mientras que al recibir una señal de OFF desenrolla la cortina por un tiempo determinado para cerrarla.

En el caso de las funciones de los actuadores de control del ventilador, cerradura y buzzer, sus respectivas funciones emiten una señal de 5V en los pines conectados a los respectivos actuadores cuando se recibe un ON del booleano correspondiente, y una señal de 0V cuando se recibe una señal de OFF para indicar que dejen de funcionar.

En el caso de las funciones de los leds RGB, estos al funcionar con lógica negativa, cuando se recibe una señal de ON del booleano correspondiente, el arduino en los pines correspondientes envía una señal de 0V para encenderlos, mientras que con la señal de OFF emite una salida de 5V para apagar el led correspondiente.

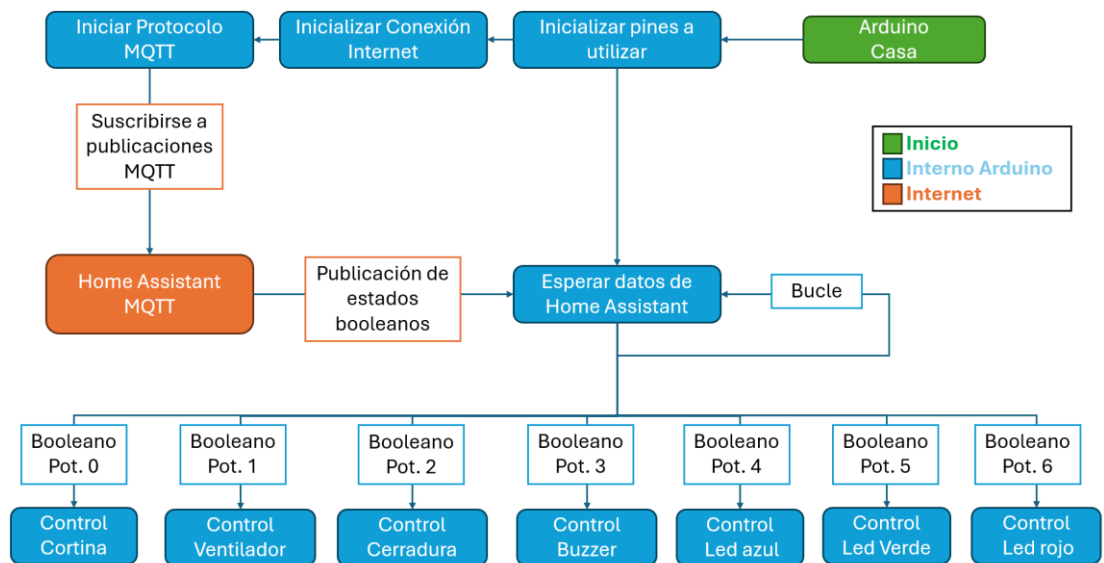


Fig. 4. 12 Funcionamiento interno del entorno simulado

4.4. Servidor Central

En esta sección se presenta la implementación del servidor central, para la integración del guante háptico y el entorno simulado. El proceso comienza con la preparación del hardware, en este

caso Raspberry Pi 5.

La configuración inicial constituye en la instalación del sistema operativo Home Assistant OS, para posteriormente la configuración de los complementos y configuraciones necesarias para la comunicación entre los dispositivos.

En la figura 4. 13, se puede observar la interfaz del programa Raspberry Pi Imager, herramienta utilizada para la instalación del sistema operativo Home Assistant OS, en Raspberry Pi 5. Este proceso consiste en seleccionar el modelo del dispositivo, el sistema operativo compatible, en el caso de este proyecto Home Assistant OS, y el medio de almacenamiento en el cual estará grabada la imagen, que en este caso es la tarjeta microSD de la Raspberry Pi.

Una vez insertada esta tarjeta en la Raspberry Pi 5, está ya ejecuta el sistema operativo de forma funcional, siempre y cuando se mantenga conectada a la red.



Fig. 4. 13 Instalador sistema Home Assitant OS

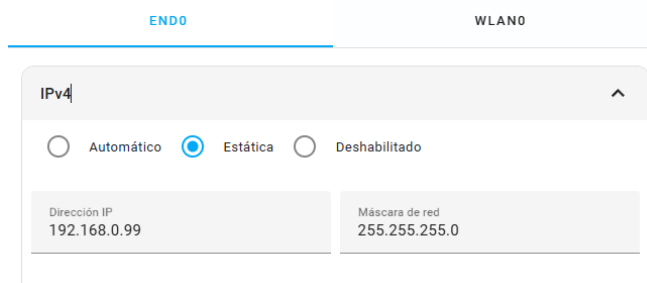
4.4.1 Configuraciones iniciales Home Assistant

Una vez instalado correctamente el sistema operativo en la Raspberry Pi, es necesario realizar una serie de configuraciones iniciales en Home Assistant. Estas configuraciones permiten establecer los parámetros básicos de funcionamiento del sistema, tales como la creación del usuario administrador, la conexión a la red local, la detección automática de dispositivos compatibles, y la instalación de complementos para la integración y control del entorno inteligente.

En la figura 4. 14, se puede observar la interfaz de configuración de red del sistema Home Assistant, en el cual se establece una dirección IP estática para el servidor central. Esta configuración

es fundamental para asegurar que el dispositivo mantenga siempre la misma dirección dentro de la red local, lo que facilita la comunicación entre distintos dispositivos y evita conflictos de conexión. Para ello se selecciona una IP estática con terminación .99 junto con su correspondiente máscara de red, garantizando una referencia fija de acceso desde otros dispositivos parte del sistema domótico.

Configura las interfaces de red



Configura las interfaces de red

ENDO WLANO

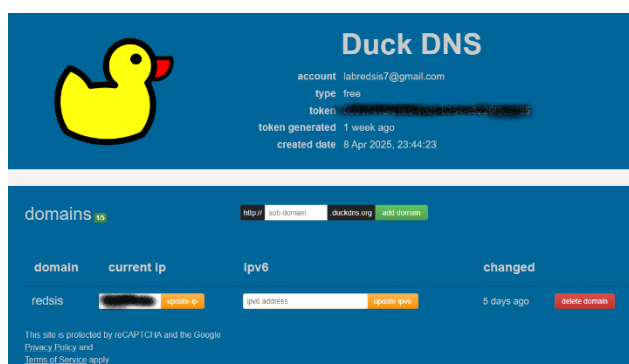
IPv4

☐ Automático
 ☒ Estática
 ☐ Deshabilitado

Dirección IP: 192.168.0.99
 Máscara de red: 255.255.255.0

Fig. 4. 14 Configuración IP

En la figura 4.15, se puede observar la interfaz web del servicio DuckDNS, en donde se llevó a cabo la configuración del dominio personalizado asignado como servidor central. En esta plataforma se registró el dominio redsis.duckdns.org, que será utilizado como dirección de acceso remoto al sistema, además de entregar un token de acceso necesario para su vinculación por el lado del servidor.



Duck DNS

account: labredsis7@gmail.com
 type: free
 token: [redacted]
 token generated: 1 week ago
 created date: 8 Apr 2025, 23:44:23

domains 15

[http:// \[redacted\] duckdns.org](#)
[add domain](#)

domain	current ip	ipv6	changed
redsis	[redacted] update ip	ipv6 address update ipv6	5 days ago delete domain

This site is protected by reCAPTCHA and the Google
 Privacy Policy and
 Terms of Service apply.

Fig. 4. 15 Configuración servicio DuckDNS

En la figura 4. 16, se puede observar la interfaz de configuración del complemento DuckDNS

por el lado de Home Assistant. En esta sección se ingresó el dominio personalizado asignado (redsis.duckdns.org), así como el token de autorización generado en la cuenta de usuario en la plataforma de DuckDNS. Además, se configuran parámetros para los términos de uso y configuración automática de certificados SSL, responsable de establecer una conexión segura (HTTPS) en el servidor y dispositivos externos.

Duck DNS

Opciones

redsis.duckdns.org

Domains

A list of DuckDNS subdomains registered under your account. An acceptable naming convention is "my-domain.duckdns.org".

Token*

The DuckDNS authentication token found at the top of the DuckDNS account landing page. The token is required to make any changes to the subdomains registered to your account.

Alias

A list aliases of domains configured on the "domains" option.

Let's Encrypt

```

1 accept_terms: true
2 algo: secp384r1
3 certfile: fullchain.pem
4 keyfile: privkey.pem
5

```

Configure Let's Encrypt options

Seconds*

300

The number of seconds to wait before updating DuckDNS subdomains and renewing Let's Encrypt certificates.

Fig. 4. 16 Configuración DuckDNS Home Assistant

En la figura 4 .17, se presenta la interfaz de configuración del complemento Mosquitto Broker, que actúa como intermediario en la comunicación entre dispositivos mediante el protocolo MQTT, utilizando el puerto 8883 para comunicarse. En esta instancia, se definió el nombre de usuario con su respectiva contraseña para el acceso al servicio, así como se configuraron los certificados de seguridad SSL previamente generados por DuckDNS.

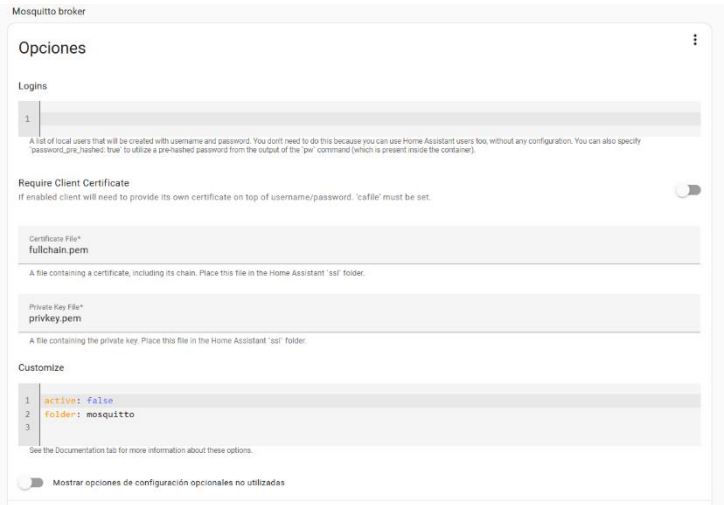


Fig. 4. 17 Configuración Mosquitto Broker

En la figura 4. 18, se muestra la configuración de redirección de puertos realizadas en el router, necesaria para permitir el acceso remoto seguro a los servicios del servidor central. En esta configuración se habilitó el puerto 443 para redirigir las solicitudes entrantes del dominio generado en DuckDNS al puerto 8123 en donde está alojada la interfaz de control de Home Assistant, así como el puerto 8883 para la comunicación por medio del protocolo MQTT.

Servidores Virtuales

Descripción	Puerto Entrante	Tipo	Dirección IP Privada	Puerto Local
☐				
☐ Home Assitant	443-443	TCP	192.168.0.99	8123-8123
☐ mqtt	8883-8883	Both	192.168.0.99	8883-8883

Fig. 4. 18 Configuración Puertos Router

En la figura 4. 19, se presenta la configuración del complemento Visual Studio Code Server, que a diferencia de los complementos anteriores solo se es necesario indicar que se inicie en cuanto el servidor se encuentre activo, y panel lateral de la interfaz de control para un fácil acceso.

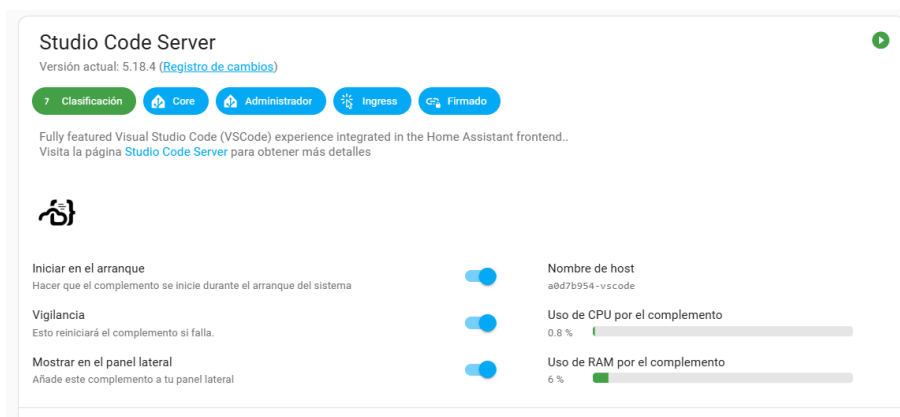


Fig. 4. 19 Configuración Studio Code Server

En la figura 4. 20, se presenta la configuración del complemento Log Viewer, que al igual que Visual Studio Code Server, solo requiere la indicación de que empiece a operar desde el momento de inicio del servidor, así como un acceso por el panel lateral para facilitar su utilización cuando sea requerido.

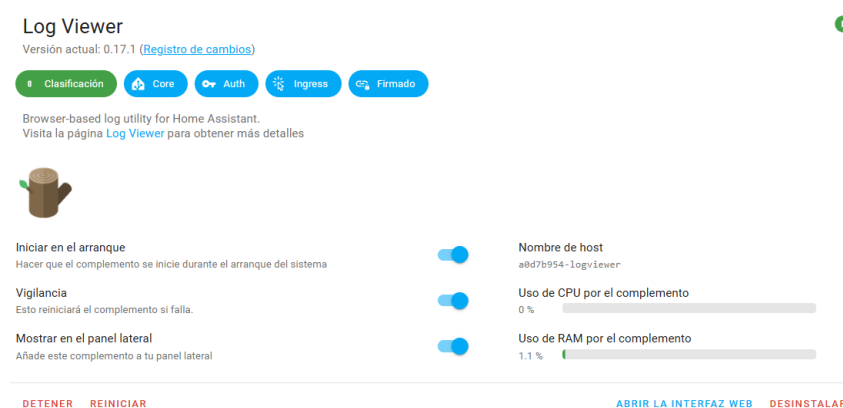


Fig. 4. 20 Configuración Log Viewer

En la figura 4. 21, se presenta el panel de control o DashBoard correspondiente al guante háptico, diseñado en la interfaz de Home Assistant. Este panel permite visualizar y simular los estados de cada sensor del guante, asignando una sección para cada dedo, en forma de dos representaciones: la parte superior representa un estado booleano de cada articulación, en donde se muestra como apagado mientras el dedo está flexionado, y encendido mientras se encuentre estirado. Además, en la parte inferior se puede ver en forma de deslizadores o sliders, el ángulo de la articulación de manera precisa en valores que van de 0 a 90. Esto ayuda a visualizar y comprobar la precisión en la cual el

guante puede detectar la posición de flexión en tiempo real de cada articulación. En la tabla [4.1]



Fig. 4. 21 DashBoard del Guante

En la tabla [4.1] se presenta la correspondencia entre los componentes del Dashboard y los sensores utilizados en el guante háptico. Los nombres asignados a cada interruptor o deslizador indican tanto el dedo correspondiente como la posición del potenciómetro. Los elementos identificados con el número 1 hacen referencia a los potenciómetros ubicados en la primera falange, mientras que los identificados con el número 2 corresponden a los potenciómetros montados sobre el nudillo.

TABLA 4. 1 relación componentes Dashboard con Sensores

Interruptor/Sliders	Potenciómetro
Índice 1	0
Índice 2	1
Corazón 1	2
Corazón 2	3
Anular 1	4
Anular 2	5
Meñique 1	6
Meñique 2	7

4.4.2 Funcionamiento interno e interacción con el Guante y Entorno simulado

En esta sección se detalla el funcionamiento del servidor central como núcleo de procesamiento e intermediación entre el guante háptico y el entorno simulado. Este subcapítulo se enfoca en describir como el servidor central, mediante Home Assistant, interpreta los datos recibidos desde el guante usando el protocolo MQTT, y en función de esta información, ejecuta las acciones correspondientes de manera interna para enviar las señales de control hacia el dispositivo del entorno simulado.

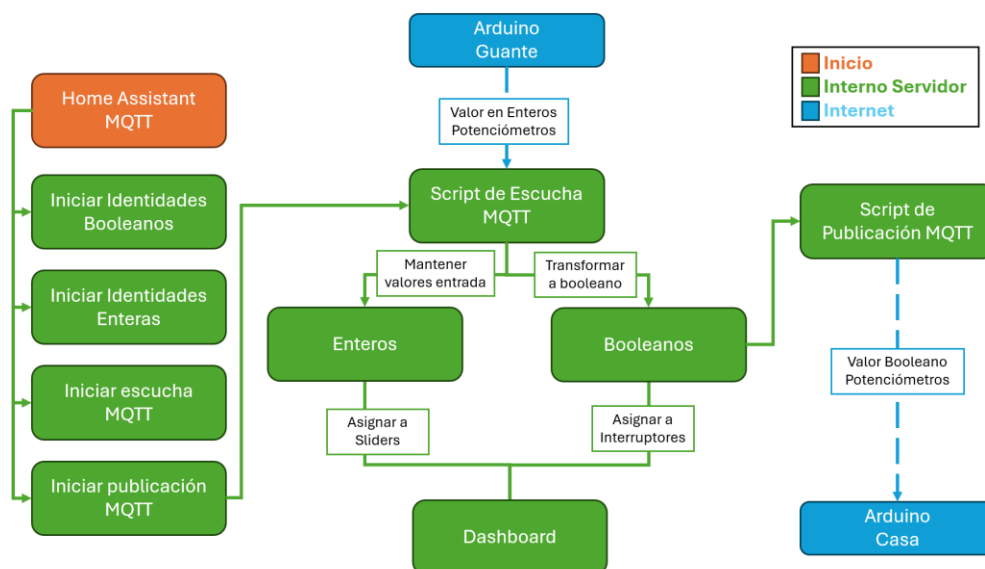


Fig. 4. 22 Funcionamiento interno de Home Assistant

En la figura 4.22 se presenta el funcionamiento interno del sistema Home Assistant, en relación con la recepción de datos provenientes del guante háptico y su posterior envío hacia el entorno simulado. El proceso comienza con la definición de entidades booleanas y enteras, las cuales son esenciales para la visualización y el control desde el panel de usuario (dashboard).

A continuación, se ejecuta un script que permite escuchar los mensajes recibidos por el protocolo MQTT, así como otro script que publica los cambios detectados en las entidades booleanas mediante suscripciones también por MQTT.

Una vez inicializados estos parámetros y funciones, el servidor entra en modo escucha, esperando un mensaje tipo array de 8 elementos, cada uno representando un valor numérico correspondiente al ángulo de una articulación del guante, en un rango de 0 a 90 grados. Estos datos, enviados por el Arduino a través de MQTT, permiten conocer la posición en tiempo real de cada dedo.

El script de escucha procesa estos valores y los asigna tanto a entidades enteras como booleanas. Las entidades enteras almacenan el valor exacto de cada articulación, mientras que para las booleanas se aplica una lógica de umbral: si el valor es menor o igual a 45, se interpreta como "apagado" (falso), y si es mayor a 45, como "encendido" (verdadero).

Finalmente, estos valores se reflejan en la interfaz del dashboard, actualizando los elementos visuales correspondientes. Las variables enteras se representan mediante sliders y permiten una visualización continua del movimiento, mientras que los cambios en las variables booleanas son utilizados por el script de publicación para enviar órdenes de activación al entorno simulado, en función del estado de los interruptores.

Capítulo 5. Resultados

5.1. Introducción

Este capítulo presenta los resultados obtenidos tras la implementación e integración de los sistemas desarrollados: el guante háptico, el entorno simulado y el servidor central. Se exponen las pruebas realizadas para validar el correcto funcionamiento de cada módulo, así como la interacción entre ellos mediante el protocolo MQTT y el entorno de automatización Home Assistant.

Se describen las observaciones registradas durante las fases de operación de cada componente, evaluando el comportamiento del sistema en tiempo real, la estabilidad de la comunicación entre dispositivos y la respuesta ante distintas acciones ejecutadas desde el guante. Los resultados permiten identificar tanto los logros alcanzados como los posibles márgenes de mejora para futuras iteraciones del proyecto.

En la figura 5. 1 puede apreciar el guante háptico siendo utilizado en distintos ángulos y disposiciones. Las imágenes muestran cómo el dispositivo se ajusta a la mano del usuario, evidenciando la correcta integración de los sensores, la estructura impresa en 3D y la caja contenedora de componentes electrónicos. Esta disposición permite observar tanto el diseño ergonómico del sistema como la flexibilidad de movimiento que ofrece, asegurando una lectura precisa de los gestos y posiciones de cada dedo.

Se destaca la integración de los componentes mecánicos y electrónicos en una configuración relativamente compacta, donde los componentes 3D como los componentes adheridos a están correctamente montados, garantizando una interacción entre los dedos y las articulaciones que se están midiendo con su respectivo potenciómetro. Al realizar pruebas con distintos movimientos por el lado mecánico se pudo comprobar que al mover una articulación ya sea en un ángulo que se puede considerar pequeño (5° - 10°) o más amplios (45° - 90°), la parte móvil de cada sensor responde de buena manera y no se logran ver pérdidas de movimiento.

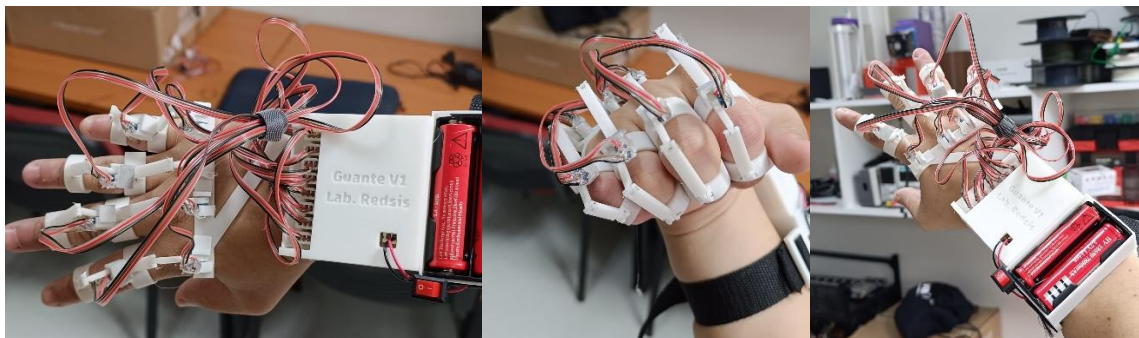


Fig. 5. 1 Guante Háptico siendo utilizado

En la figura 5. 2 puede ver el entorno simulado con todos sus componentes funcionando a la vez, en donde se pudo ver que la cortina se abrió de manera óptima, el ventilador se encuentra funcionando, la cerradura se retrajo para simular la apertura de la puerta, los leds RGB, al estar todos encendidos muestran una luz blanca, y el buzzer actuando como un sistema de alerta o notificaciones sonó en el momento indicado. Estas acciones en donde todo se encuentra en un estado de encendido se dan cuando la mano está en un estado abierta, o todas sus articulaciones extendidas.



Fig. 5. 2 Entorno Simulado funcionando

5.2. Prueba del sistema

Se realizaron pruebas con distintas posiciones de los dedos de la mano en la cual estaban montado el guante, en donde se decidió no colocar una imagen con cada combinación de acciones posibles por ser una cantidad demasiado grande, pero al ser 8 combinaciones binarias se tienen 256 combinaciones en los estados del entorno simulado. En cambio, se evaluaron solo algunas de estas combinaciones.

Se probaron varias combinaciones de movimiento de la mano, como lo son activar y desactivar cada actuador de manera individual, varios en conjuntos y movimientos que se pueden considerar aleatorios de la mano para probar el funcionamiento del entorno en distintas situaciones. Estas combinación y control de los actuadores a distintas velocidades no causo problemas en los funcionamientos de los dispositivos ni de manera individual ni global como sistema.

El tiempo de respuesta de la activación y desactivación de los actuadores respecto a los movimientos del guante son casi imperceptibles independientemente de la velocidad a la que este se mueva, por lo que se puede concluir que no hay problemas en la velocidad de comunicación entre el guante y el entorno simulado y es bastante natural su control.

5.2.1 Control cortina

En la figura 5.3 se observa el gesto realizado con la mano para activar la apertura de la cortina. En esta acción, únicamente el dedo índice, asociado al potenciómetro 0, se encuentra completamente extendido, mientras que los demás dedos permanecen flexionados. Esta configuración provoca que solo el actuador correspondiente al dedo extendido se active, mientras que los demás permanecen apagados o en estado de reposo.

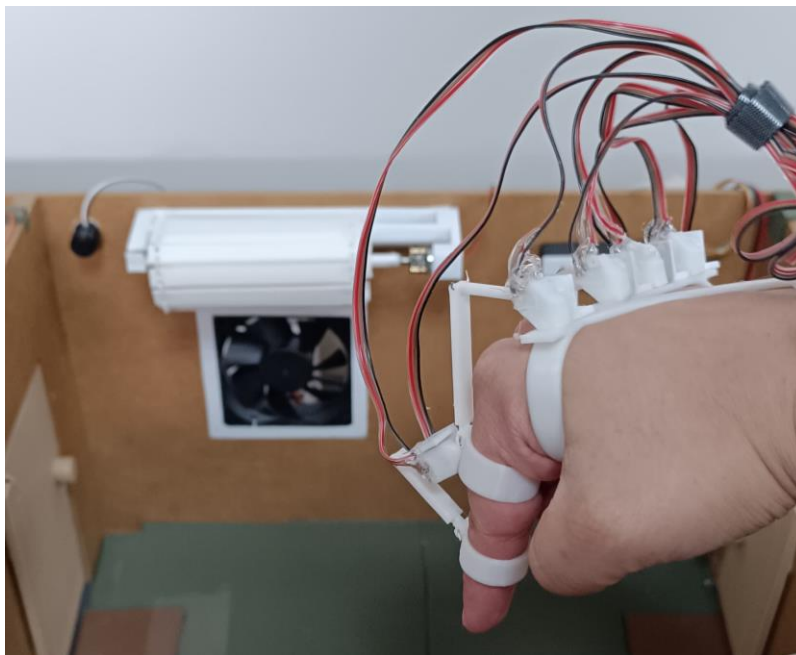


Fig. 5. 3 Abrir Cortina

5.2.2 Control ventilación

En la Figura 5.4 se muestra la ejecución del gesto necesario para activar el sistema de ventilación del entorno simulado. Las articulaciones vinculadas a los potenciómetros 0 y 1, correspondientes al dedo índice se encuentran extendidas. Esta configuración permite mantener la cortina abierta, condición indispensable para la operación visual del ventilador, y activar simultáneamente el actuador del sistema de ventilación. El resto de los actuadores permanece desactivado, ya que los valores de sus potenciómetros asociados no superan los umbrales definidos para su activación.

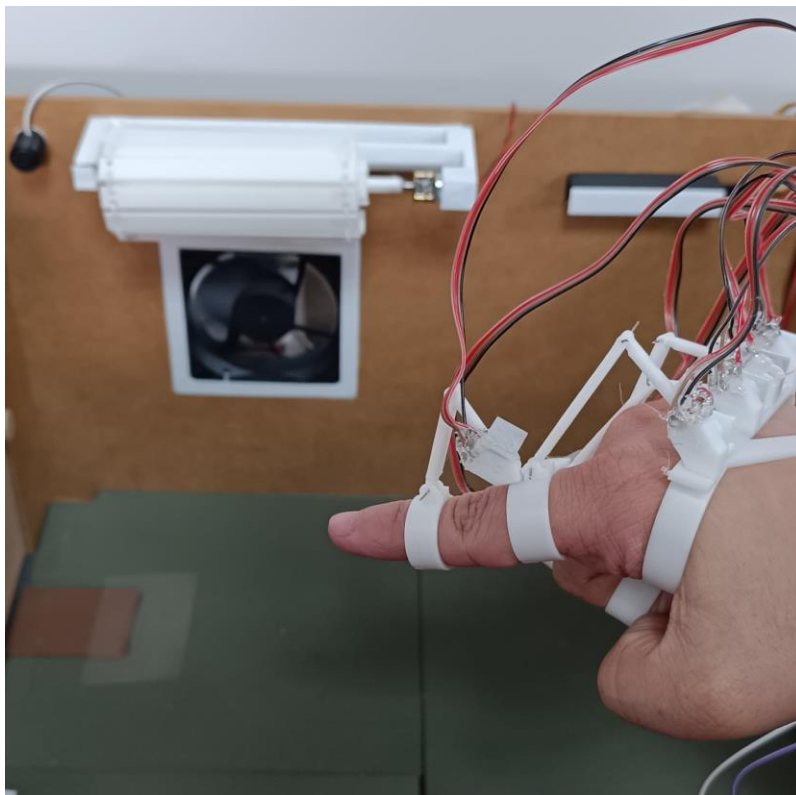


Fig. 5. 4 Encender ventilación

5.2.3 Control cerradura

En la Figura 5.5 se observa la ejecución del gesto correspondiente a la apertura de la cerradura, generado por la extensión del dedo corazón, cuya articulación está asociada al potenciómetro 2. Esta posición produce una lectura alta en dicho canal, la cual es procesada por el servidor central mediante el protocolo MQTT. Como resultado, se activa el actuador conectado a la cerradura electrónica, provocando la retracción del pistón interno y permitiendo así la apertura del mecanismo tal como se esperaba.

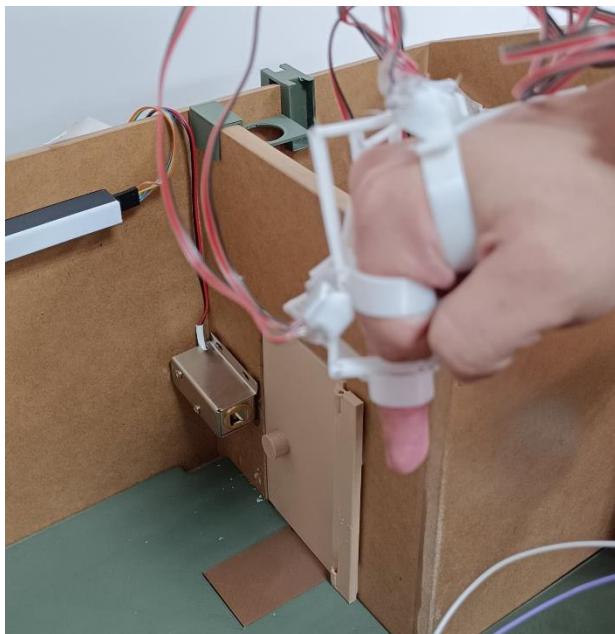


Fig. 5. 5 abrir cerradura

En la Figura 5.6 se presentan dos gestos distintos utilizados para controlar el sistema de iluminación RGB. En la imagen izquierda, solo la articulación correspondiente al potenciómetro 4 asociado al canal rojo del LED RGB se encuentra extendida, activando únicamente dicho canal y generando una luz de color rojo. En la imagen derecha, se observa la activación simultánea de todas las articulaciones relacionadas con los tres canales del LED (rojo, verde y azul), lo que produce una combinación aditiva que da como resultado una luz blanca. No se incluyeron otras combinaciones cromáticas, ya que a pesar de ser funcionales y apreciables a simple vista, la cámara utilizada no logró capturar adecuadamente la variación de colores emitidos, debido a limitaciones en fidelidad cromática y exposición.

5.2.4 Control iluminación RGB

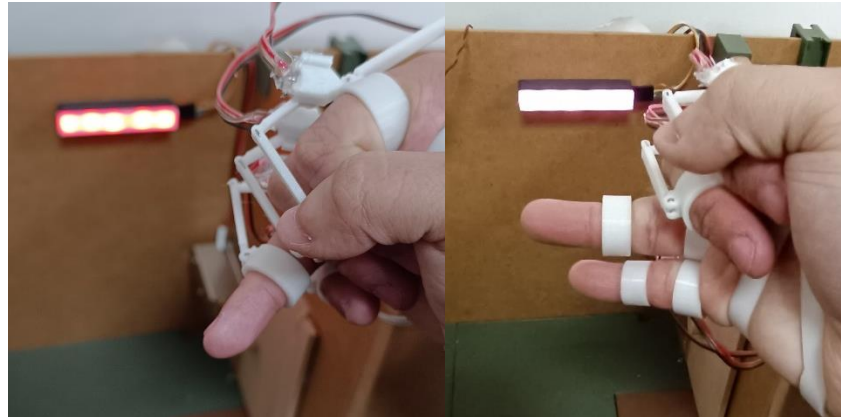


Fig. 5. 6 Combinaciones, iluminación RGB

5.2.5 Visualización dashboard Home Assistant

En la figura 5. 7 se puede ver el dashboard del servidor del guante, mientras que en la figura 5. 8 se ve la posición que tenía la mano al momento de visualizar el dashboard. Se puede ver en tiempo real la posición actual del guante y cada una de sus articulaciones, En la parte superior se logra apreciar los interruptores que representan el estado actual de los actuadores en el entorno simulado, y en la parte inferior el ángulo de las articulaciones actual con una precisión de 1° , la imagen de la figura 5. 3 se tomó en un momento en el que la mano se encontraba relajada o lo que se puede considerar en reposo, posición en la cual los dedos tienden a estar en un ángulo que se puede considerar cercano a los 45° o la “mitad” de su posición.

Se pudo probar, como se ve en la imagen que el servidor realiza bien la conversión del Angulo actual de cada articulación a una acción binaria, en donde los valores que se ven en las sliders menores de 45° se pueden ver en una posición de apagado y los superiores a 45° como encendidos en su contraparte superior.

Mientras se realizaban las pruebas de movimiento con el guante se pudo ver que los componentes del dashboard se movían en tiempo real y de manera precisa respecto a los movimientos de la mano. Al realizar pruebas de movimiento más finas con cada articulación se pudo probar gracias a la visualización a modo de sliders que los sensores no presentaban ruido, ya que al mantener la mano estática no se apreció ninguna variación en alguna de estas medidas, así como también poder alterar estos valores hasta en precisiones de 1° de manera individual cada valor, según lo que la motricidad de la mano lo permitía.



Fig. 5. 7 DashBoard del guante en funcionamiento

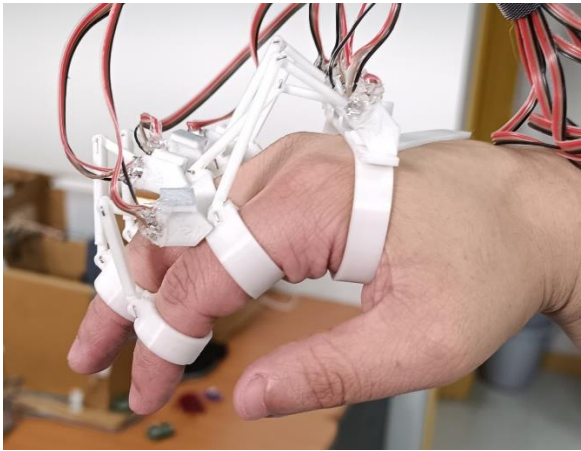


Fig. 5. 8 DashBoard del guante en funcionamiento

Capítulo 6. Conclusiones

A partir de los resultados obtenidos durante la fase de pruebas e integración del sistema, se puede concluir que el guante háptico desarrollado cumplió de manera satisfactoria con los objetivos propuestos en términos de funcionalidad, precisión y respuesta en tiempo real. La integración entre los componentes electrónicos, mecánicos y estructurales impresos en 3D logró una adaptación ergonómica y estable en la mano del usuario, permitiendo una lectura precisa de los movimientos de cada articulación con resolución de hasta los 1° grados, sin presentar interferencias ni pérdidas de señal.

Asimismo, el entorno simulado respondió correctamente a las distintas combinaciones de movimientos enviadas desde el guante, demostrando una comunicación robusta del protocolo MQTT. Se evidenció que las acciones asignadas a los actuadores del entorno fueron ejecutadas de forma sincronizada y sin retardos perceptibles, incluso al modificar la secuencia y velocidad de los movimientos.

Por su parte, el servidor central, mediante el panel de control dashboard, permitió visualizar de forma clara e intuitiva el estado actual del guante y cada uno de sus componentes, validando tanto la precisión del sistema de captura como su capacidad de actuación en tiempo real.

En conjunto el sistema integrado demostró ser funcional, estable y adaptable, sentando una base sólida para futuras iteraciones y mejoras.

6.1.1 Guante háptico

El guante háptico demostró un rendimiento preciso y estable en la lectura de los movimientos de los dedos, gracias a la correcta integración entre sensores, estructura mecánica y el sistema de adquisición. La precisión angular alcanzada de 1° por articulación permitió identificar los gestos sin interferencias, validando su utilidad como interfaz natural de control. El diseño modular y compacto facilitó tanto el montaje como el uso continuo del dispositivo.

6.1.2 Entorno Simulado

El entorno simulado respondió de manera precisa a las señales enviadas desde el guante, accionando cortinas, ventiladores, iluminación y buzzer de forma sincronizada y sin errores. Se validó

su capacidad para ejecutar múltiples combinaciones de entrada sin perder estabilidad, lo que demuestra su idoneidad como espacio inteligente controlado por gestos en tiempo real.

6.1.3 Servidor Central

Home Assistant cumplió su función como núcleo integrador del sistema, actuando como intermediario entre el guante háptico y el entorno simulado. Su arquitectura permitió recibir, procesar y visualizar en tiempo real los datos enviados por el guante, facilitando el control de los actuadores según las condiciones establecidas. El panel de control ofreció una interfaz clara e intuitiva, mientras que el uso de MQTT garantizó fluidez y fiabilidad en la gestión de eventos, haciendo del sistema una solución escalable para entornos domóticos personalizados.

6.2. Trabajo Futuro

Este trabajo abre la posibilidad de desarrollar múltiples investigaciones futuras, tanto en la recopilación de datos como incorporaciones de nuevos sensores y dispositivos. Entre ellas están:

- Agregar sensores al dedo pulgar para lograr la medición de la totalidad de las articulaciones de la mano
- Hacer una recopilación de datos con diversas acciones y movimientos para poder reconocer gestos específicos mediante reconocimiento de patrones, como por ejemplo el lenguaje de señas.
- Adición de más sensores en el guante háptico, como lo puede ser un sensor acelerómetro/giroscopio para, además de recolectar el movimiento individual de cada dedo poder saber la posición y movimiento general de la mano que porta el guante.
- Agregar un dispositivo adicional, como una pequeña pantalla LCD, que brinde retroalimentación del guante. Esta pantalla podría mostrar información útil, como el estado de la conexión a internet, la confirmación de una calibración exitosa y los datos individuales de cada sensor, en caso de ser necesarios.
- Agregar sensor de pulso cardíaco y presión sanguínea para poder realizar monitorizaciones de los usuarios en casos de personas que requieran algún tipo de atención constante.

- Cambiar la batería por alguna más compacta, y agregar un módulo que permita cargar de manera directa el guante, de forma que no sea necesario retirar las baterías.
- Implementar el sistema como un intermediario entre el usuario y asistentes de voz como Google Assistant o Alexa para la identificación de lenguaje de señas. Incluso, podría funcionar como un asistente autónomo, especialmente útil para personas con discapacidades que les impidan comunicarse mediante la voz.
- Integración en otros ámbitos además de una Smart House, como lo pueden ser laboratorios, empresas, realidad virtual, o robótica.

Bibliografía

- [1] C. Stolojescu-Crisan, C. Crisan, and B.-P. Butunoi, "An IoT-based smart home automation system," *Sensors*, vol. 21, no. 11, p. 3784, May 2021. doi: 10.3390/s21113784.
- [2] V. A. Orfanos, S. D. Kaminaris, D. Piromalis, and P. Papageorgas, "Smart Home Automation in the IoT Era: A Communication Technologies Review," *AIP Conference Proceedings*, vol. 2307, no. 1, p. 020054, Dec. 2020. doi: 10.1063/5.0032939.
- [3] C. Irugalbandara, A. S. Naseem, S. Perera, S. Kiruthikan, and V. Logeeshan, "A Secure and Smart Home Automation System with Speech Recognition and Power Measurement Capabilities," *Sensors*, vol. 23, no. 5784, pp. 1–15, Jun. 2023. doi: 10.3390/s23135784.
- [4] H. Yar, A. S. Imran, Z. A. Khan, M. Sajjad, and Z. Kastrati, "Towards Smart Home Automation Using IoT-Enabled Edge-Computing Paradigm," *Sensors*, vol. 21, no. 4932, pp. 1–23, Jul. 2021. doi: 10.3390/s21144932.
- [5] W. N. A. A. W. Husni, M. Faisal, P. J. Ker, D. N. T. How, M. A. Hannan, and M. A. Salam, "Intelligent Home Automation System for Disabled People," *International Journal of Engineering and Advanced Technology (IJEAT)*, vol. 9, no. 2, pp. 4142–4148, Dec. 2019. doi: 10.35940/ijeat.B4921.129219.
- [6] C. Pacchierotti, S. Sinclair, M. Solazzi, A. Frisoli, V. Hayward, and D. Prattichizzo, "Wearable Haptic Systems for the Fingertip and the Hand: Taxonomy, Review, and Perspectives," *IEEE Transactions on Haptics*, vol. 10, no. 4, pp. 580–600, Oct. 2017. doi: 10.1109/TOH.2017.2689006.
- [7] K. Song, S. H. Kim, S. Jin, S. Kim, S. Lee, J. Kim, J.-M. Park, and Y. Cha, "Pneumatic actuator and flexible piezoelectric sensor for soft virtual reality glove system," *Scientific Reports*, vol. 9, no. 8988, pp. 1–11, 2019. doi: 10.1038/s41598-019-45422-6.
- [8] M. F. Ahmed, N. Hasan, M. W. Ullah, M. H. Kabir y M. Rahman, "Elderly Safety and Communication: An IoT-Based Smart Glove for Fall Detection and Hand Gesture Communication", 3rd International Conference on Advancement in Electrical and Electronic Engineering (ICAEEE), pp. 1–8, Abr. 2024. doi: 10.1109/ICAEEE62219.2024.10561702.
- [9] M. F. Ahmed y N. Hasan, "IoT-Enabled Smart Glove for Enhanced Elderly Care: A Health Monitoring Solution", *Preprint*, Research Square, pp. 1–25, Oct. 2024. doi: 10.21203/rs.3.rs-

4956656/v1.

- [10] S. B, P. P, P. Gupta, S. Aluvala and T. Stephan, "Enhanced Health Monitoring Using IoT-Embedded Smart Glove and Machine Learning," *2023 3rd International Conference on Innovative Sustainable Computational Technologies (CISCT)*, Dehradun, India, 2023, pp. 1-5, doi: 10.1109/CISCT57197.2023.10351358.
- [11] M. Oudah, A. Al-Naji, and J. Chahl, "Hand Gesture Recognition Based on Computer Vision: A Review of Techniques," *Journal of Imaging*, vol. 6, no. 8, pp. 73–89, Jul. 2020. doi: 10.3390/jimaging6080073.
- [12] H. Luo, Y. -B. Lin, C. -C. Liao and Y. -H. Huang, "An IoT-Based Microservice Platform for Virtual Puppetry Performance," in *IEEE Access*, vol. 11, pp. 103014-103032, 2023, doi: 10.1109/ACCESS.2023.3315656.
- [13] A. Atzori, A. Gijsberts, C. Castellini, B. Caputo, A. Hager, S. Elsig, G. Giatsidis, F. Bassetto y H. Müller, "EMG-based hand gesture recognition: comparison of features and classifiers," *Biomedical Signal Processing and Control*, vol. 14, pp. 246–255, Oct. 2014. doi: 10.1016/j.bspc.2014.08.009.
- [14] K. Li, Z. Li, H. Zeng, and N. Wei, "Control of Newly-Designed Wearable Robotic Hand Exoskeleton Based on Surface Electromyographic Signals," *Frontiers in Neurorobotics*, vol. 15, no. 711047, Sep. 2021. doi: 10.3389/fnbot.2021.711047.
- [15] J. Gudiño-Lau, D. Vélez-Díaz, F. Chavez-Montejano, M. Durán-Fonseca, S. Charre-Ibarra, y O. Díaz-Parra, "Las TIC aplicadas a un dispositivo háptico Novint Falcon," *Revista Xikua* <https://repository.uaeh.edu.mx/revistas/index.php/xikua/article/view/329/4431> [Accedido: 30-abr-2025].
- [16] S. Khandelwal, "What Are Haptics?", Spiceworks, Aug. 2, 2023. [Online]. Available: <https://www.spiceworks.com/tech/tech-general/articles/what-are-haptics/> [Accedido: 30-abr-2025].
- [17] A. I. Team, "La domótica y el Internet de las cosas," *Alfa IoT*, 2021. [Online]. Available: <https://alfaiot.com/actualidad-iot/la-domotica-y-el-internet-de-las-cosas/> [Accessed: 30-Apr-2025].