



**Automatización de ambientes hospitalarios para la reducción del gasto energético y
conservación del medio ambiente.**

Daniel Esteban Parra Gallego

Bioingeniero

Asesor

Jonathan Gallego Londoño, M.Sc Ingeniería Biomédica

Universidad de Antioquia

Facultad de ingeniería

Bioingeniería

Medellín

2023

Referencia

- [1] D. Parra Gallego, "Automatización de ambientes hospitalarios para la reducción del gasto energético y conservación del medio ambiente", Semestre de industria, Bioingeniería, Universidad de Antioquia, Medellín, 2023.

Estilo IEEE (2020)



Centro de Documentación de Ingeniería

Repositorio Institucional: <http://bibliotecadigital.udea.edu.co>

Universidad de Antioquia - www.udea.edu.co

Rector: John Jairo Arboleda Céspedes.

Decano/Director: Julio César Saldarriaga Molina.

Jefe departamento: John Fredy Ochoa Gómez.

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

Dedicatoria

Queridos padres, familiares y amigos:

Dedico este trabajo a ustedes, quienes han sido mi mayor apoyo y motivación en esta importante etapa de mi formación académica, especialmente a mi madre Luz Dora Gallego, a mi hermana Andrea Parra y a Claudia Arango, gracias por creer en mí y por brindarme su constante apoyo, alentándome en cada paso y animándome a seguir adelante incluso en los momentos más difíciles.

Este logro no habría sido posible sin su amor, paciencia y guía incondicional. Cada una de sus enseñanzas y valores me han ayudado a forjar mi carácter y me han hecho crecer como persona y como estudiante.

Espero que este trabajo les llene de orgullo tanto como a mí, y que sea una muestra del compromiso y dedicación que he puesto en mi formación. Gracias por ser mi motor y por estar siempre a mi lado.

Con todo mi amor y gratitud.

Agradecimientos

Quiero expresar mi más sincero agradecimiento a mi familia, quienes siempre han estado a mi lado brindándome su amor, paciencia y apoyo incondicional en cada etapa de mi formación académica.

Asimismo, quiero agradecer a mis compañeros de clase, quienes compartieron conmigo su conocimiento, tiempo y experiencias durante estos años. Gracias por ser un gran equipo de estudio y de trabajo, y por hacer de esta etapa una experiencia enriquecedora y divertida.

Por otro lado, quisiera agradecer a los profesores que me guiaron en este proceso, quienes, con su sabiduría, dedicación y apoyo, me enseñaron a pensar críticamente y a profundizar en mi conocimiento. Gracias por sus enseñanzas y por su compromiso con mi formación.

Por último, quiero expresar mi gratitud a todas las personas que de una u otra manera contribuyeron al éxito de este trabajo. Sin su ayuda, este logro no habría sido posible.

¡Gracias a todos por su apoyo y motivación!

TABLA DE CONTENIDO

RESUMEN	9
ABSTRACT	10
I. INTRODUCCIÓN	11
II. OBJETIVOS	13
Objetivo general	13
Objetivos específicos	13
III. MARCO TEÓRICO	14
IV. METODOLOGÍA	17
Análisis de la infraestructura y la tecnología de la clínica.	17
Análisis de los circuitos de la tecnología existente.	17
Selección de los microcontroladores para la construcción de los módulos.	18
Generalidades del funcionamiento del sistema modular.	18
Diseño del sistema modular.	20
Construcción del sistema modular.	20
Programación de los módulos del sistema y otras características.	20
Diseño de la carcasa o presentación visual del sistema.	21
Prueba de usabilidad y de funcionamiento.	21
V. RESULTADOS	23
Identificación de requerimientos tecnológicos y de infraestructura.	23
Análisis de la infraestructura, distribución de habitaciones y la alimentación energética.	23
Identificar los puntos de conexión energética de las habitaciones.	24
Análisis del circuito de control Honeywell y circuito de control de luces.	25
Selección de los dispositivos a usar que permita la interconexión inalámbrica.	27
Diseñar del sistema de automatización de luces y del aire acondicionado.	27

Diseño del módulo sensorial.	29
Diseño del módulo de control de luces.	30
Diseño e integración del módulo de control del aire acondicionado.	31
Evaluar la funcionalidad y usabilidad del sistema.	34
Implementación del sistema modular.	34
Realización de ajuste de parámetros de funcionamiento del sistema.	36
Aplicación de pruebas de usabilidad del sistema.	36
Realización de mejoras y entregar del sistema funcional.	37
VI. ANÁLISIS	38
Sobre el primer objetivo específico	38
Sobre el segundo objetivo específico	39
Sobre el tercer objetivo específico	40
VII. CONCLUSIONES	42
REFERENCIAS	44
ANEXOS	45

LISTA DE TABLAS

TABLA I: RESULTADOS DEL TEST DE FUNCIONAMIENTO DEL MÓDULO DE CONTROL DE LUCES	35
---	----

LISTA DE FIGURAS

Fig. 1. Fases empleadas para el desarrollo del proyecto, enfocadas al cumplimiento de los objetivos específicos.	22
Fig. 2. Imagen tomada del panel de las rutas de evacuación del piso 6.	23
Fig. 3. Distribución eléctrica de las habitaciones. Rojo: suministro de tomas e iluminación. Azul: suministro del aire acondicionado.	24
Fig. 4. Distribución y ubicación del aire y las luces dentro de la habitación. Vista esquemática.	25
Fig. 5. Distribución de los elementos de control del aire (azul) y luces (amarillo) dentro de la habitación.	25
Fig. 6. Pines de conexión de la tecnología Honeywell modelo T6861V2WG para integración del módulo de control inalámbrico del aire acondicionado.	26
Fig. 7. Comportamiento digital de los pines de integración para la lectura y escritura de señales de control.	26
Fig. 8. Puntos de conexión para el control de la iluminación.	27
Fig. 9. Diseño de las tarjetas PCB del sistema modular. De izquierda a derecha módulo sensorial, módulo control de luces y módulo control del aire acondicionado.	28
Fig. 10. Kits de desarrollo basados en ESP32 y ESP8266.	29
Fig. 11. Módulo sensorial.	29
Fig. 12. Módulo sensorial con cubierta final impresa en 3D.	30
Fig. 13. Módulo de control de luces con cubierta final impresa en 3D.	31
Fig. 14. Circuito de conversión de niveles lógicos 3.3V a 5V.	32
Fig. 15. Circuito de envío de señal de apagado/encendido del aire acondicionado.	32
Fig. 16. Módulo de control del aire sin la etapa de transformación de la energía.	33
Fig. 17. Módulo de control del aire con cubierta final impresa en 3D.	34
Fig. 18. Prueba parcial semiautomatizada del módulo de control del aire.	35

SIGLAS, ACRÓNIMOS Y ABREVIATURAS

CO₂	Dióxido de carbono
LED	Light Emitting Diode
IoT	Internet of Things
PCB	Printed Circuit Board
GEI	Gases de Efecto Invernadero
ADC	Analog to Digital Converter
DAC	Digital to Analog Converter
PWM	Pulse Width Modulation
I²C	Inter Integrated Circuit
I²S	Integrated Interchip Sound
SPI	Serial Peripheral Interface
UART	Universal Asynchronous Receiver Transmitter
WIFI	Wireless Fidelity
BLE	Bluetooth Low Energy
RAM	Random Access Memory
AC	Alternating Current
GND	Ground
I+D+i	Investigación, Desarrollo e innovación

RESUMEN

Una oportunidad de mejora que requiere solucionar la Clínica El Rosario sede El Tesoro Medellín, es el alto consumo energético relacionado con el control de la temperatura y la iluminación, las cuales son controladas manualmente por pacientes, acompañantes y personal asistencial dentro del servicio de hospitalización. Cuando los pacientes son dados de alta se frecuenta dejar activo el aire acondicionado y las luces, por lo que se propone automatizar el control de las variables involucradas en ausencia de personal en las habitaciones del servicio para reducir el consumo energético y disminuir el impacto ambiental. Por tal motivo se diseñó un sistema modular basado en microcontroladores interconectados inalámbricamente. Cada uno de los módulos cumplen con funciones específicas como: el sensado de presencia, el control de las luces y el control del aire acondicionado. Estos módulos fueron adaptados a la tecnología actual de la clínica permitiendo el control de dichas variables. Su función se fundamenta en procesar los datos de un sensor de presencia RCWL-0516 cada 330 milisegundos a través de un microcontrolador ESP32 que envía una señal de apagado a los módulos de control del aire/luces si no detecta actividad por más de 3.5 minutos. Los módulos que reciben la señal están basados en microcontroladores ESP8266 los cuales evalúan el estado de su variable, si está activa, deshabilita la variable, de lo contrario, no toman ninguna acción. Se adaptó cada módulo del sistema y se probó su funcionamiento de manera individual por cada variable a controlar, se observó un tiempo de respuesta de apagado en promedio de 3,62 minutos (3:37 mm:ss) de 8 muestras tomadas. Se decidió no realizar ningún tipo de ajuste. Posteriormente se hace una prueba conjunta de configuración del sistema y se concluyó que el sistema funciona perfectamente para lo que fue diseñado y que puede ser mejorado en próximas versiones.

***Palabras clave* — Automatización de ambientes hospitalarios, sistema modular inalámbrico, sensor de presencia, RCWL-0516, diseño de tarjetas PCB, integración de tecnologías, control de temperatura, control de iluminación, reducción del consumo energético, impacto ambiental, ESP32, ESP8266.**

ABSTRACT

An improvement opportunity that requires a solution at the Clínica El Rosario, El Tesoro Medellin branch, is the high energy consumption related to temperature and lighting control, which is manually controlled by patients, companions, and healthcare staff within the hospitalization service. When patients are discharged, it is common to leave the air conditioning and lights on, so it is proposed to automate the control of the variables involved in the absence of personnel in the service rooms to reduce energy consumption and decrease environmental impact. For this reason, a modular system based on wirelessly interconnected microcontrollers was designed. Each of the modules performs specific functions such as sensing presence, controlling lights, and controlling air conditioning. These modules were adapted to the clinic's current technology, allowing control of these variables. Their function is based on processing data from an RCWL-0516 presence sensor every 330 milliseconds through an ESP32 microcontroller that sends a shutdown signal to the air/light control modules if it does not detect activity for more than 3.5 minutes. The modules that receive the signal are based on ESP8266 microcontrollers, which evaluate the state of their variable, if it is active, it disables the variable; otherwise, it takes no action. Each module of the system was adapted and its operation was individually tested for each variable to be controlled. An average shutdown response time of 3.62 minutes was observed from 8 samples taken. It was decided not to make any adjustments. Subsequently, a joint system configuration test was conducted and it was concluded that the system works perfectly for what it was designed for and can be improved in future versions.

***Keywords* — Automation of hospital environments, wireless modular system, presence sensor, RCWL-0516, PCB design, integration of technologies, temperature control, lighting control, reduction of energy consumption, environmental impact, ESP32, ESP8266.**

I. INTRODUCCIÓN

La Clínica El Rosario, en su plan de mejoramiento continuo y de gestión ambiental (código EST-SGA-MI-01), define las “estrategias y operaciones para la protección del medio ambiente”, con la visión de lograr “la sustentabilidad a través de la mitigación de efectos potencialmente adversos de las condiciones ambientales de la institución, [...] y la mejora de su desempeño ambiental”, lo que se encuentra alineado con los requerimientos de la Norma ISO 14001:2015 [1]. Además, el Plan De Mitigación De Huella De Carbono (código EST-SGA-PL-04) para 2025, tiene como objetivos específicos: “Implementar estrategias que contribuyan a la reducción de las toneladas de CO₂ en aproximadamente el 17.19% durante los primeros tres (3) años”, “Optimizar el sistema de aire acondicionado para ahorrar en un 12% el consumo energético” y “Renovar tecnología de iluminación por LED para disminuir en 1.4% el consumo energético durante el primer año”; Sin embargo, dentro de este plan, las estrategias se centran en la automatización de las unidades manejadoras de aire (UMAS) de la mayoría de los servicios, pero no se encuentran incluidas el servicio de hospitalización ya que, cada una de las habitaciones de este servicio cuentan con un aire acondicionado (fan coil), el cual tiene un funcionamiento independiente a las UMAS, por tal motivo, la automatización de estos aires acondicionados puede tener un aporte significativo en la reducción del consumo energético; en cuanto a la iluminación, si bien, en esta propuesta de proyecto de prácticas, no se propuso realizar la renovación de la tecnología sino más bien, tener un control automatizado de las luces, independientemente del tipo de tecnología en uso, puede tener un impacto positivo para ayudar alcanzar estos objetivos, lo que se convierte en una oportunidad de mejora para la clínica.

Para aportar al cumplimiento de lo anterior, se construyó un sistema modularizado a partir del uso de microcontroladores, sensores de presencia y actuadores que permiten el control automatizado del aire acondicionado y de las luces de las habitaciones del servicio de Hospitalización en los momentos en las que dichas habitaciones se encuentran con ausencia de pacientes.

El sistema de módulos previsto para dar solución a esta oportunidad de mejora funciona utilizando un protocolo de comunicación inalámbrica llamado ESP-NOW (protocolo de comunicación propio de la familia de microcontroladores Espressif), que es de utilidad para la

comunicación entre los módulos de control remotamente y en tiempo real, y que además puede ser combinado con protocolos de conexión a internet para monitoreo a distancia lo que le da la propiedad de ser un sistema modular con Internet de las Cosas (por sus siglas en inglés IoT).

Estos módulos de control de las variables de iluminación y temperatura cuentan con subsistemas integrados que se conectan a un módulo central (módulo sensorial), el cual permite el procesamiento de los datos generados por el sensado de presencia de pacientes a través de un sensor de presencia RCWL-0516, y que posteriormente producirá como respuesta de señales de apagado/encendido dependiendo de un tiempo prolongado de ausencia o presencia dentro del ambiente, permitiendo encender y apagar las luces y el aire acondicionado en los momentos en los que sean necesarios o no su funcionamiento.

Cada uno de los módulos que conforman el sistema cuentan con dos botones de configuración para enlazar y desenlazar (SYNC y CLEAR respectivamente) los módulos de control de aire/luces al módulo central de sensado. Cuenta con indicadores del estado de conexión o de enlace entre módulos e indicadores de conexión eléctrica o alimentación.

El diseño del circuito esquemático y el diseño de cada una de las tarjetas PCB que componen el sistema fueron realizados online con el servicio de diseño, fabricación y ensamble de tarjetas electrónicas de la empresa china JLCPCB quienes ofrecen un software de diseño de circuitos electrónicos EasyEDA.

A pesar de que el diseño de la tarjeta PCB no pudo ser materializado, los módulos fueron prototipados con kits de desarrollo basados en ESP32 y ESP8266, con ensamble manual de cada uno de los componentes requeridos para el funcionamiento de cada módulo. Los módulos resultantes fueron puestos a prueba en el servicio de hospitalización mostrando un correcto funcionamiento según el objetivo que se pretendía lograr.

Para concluir, se espera que la implementación futura de este sistema en todas las habitaciones del servicio de hospitalización pueda dar muestra de la reducción del consumo energético gracias al control del encendido y apagado de las variables relacionadas y con esto apuntando al logro de los objetivos específicos del Plan De Mitigación De Huella De Carbono de la Clínica el Rosario.

II. OBJETIVOS

Objetivo general

Desarrollar un sistema de automatización del aire acondicionado y la iluminación de las habitaciones del servicio de Hospitalización de la Clínica El Rosario a través de la implementación de un sistema de módulos basados en microcontroladores con conexión a internet (IoT) para ayudar a disminuir el consumo energético.

Objetivos específicos

- Identificar los requerimientos tecnológicos y de infraestructura del servicio de Hospitalización que permita el diseño y la implementación de un sistema modular inalámbrico para la automatización del aire acondicionado y la iluminación.
- Diseñar un sistema que mediante la detección de presencia de personas dentro de las habitaciones permita controlar las luces y el aire acondicionado de manera remota a través de internet, generando un reporte de estado que se pueda visualizar y monitorear.
- Evaluar la funcionalidad y usabilidad del sistema mediante la recopilación de los datos del comportamiento del sistema instalado en una de las habitaciones del servicio de hospitalización.

III. MARCO TEÓRICO

Uno de los avances más importantes en las telecomunicaciones que ha permitido el desarrollo acelerado de la tecnología es el IoT, la cual consiste en una red de objetos físicos ("cosas") que integran sensores y actuadores, además de software y otras tecnologías con el fin de conectarse entre sí, compartir e intercambiar datos con otros dispositivos y controlar sistemas de manera remota a través del Internet [2].

Estos sistemas con capacidad de conectarse a internet traen consigo otras tecnologías y sistemas que apoyan su funcionamiento y que se han venido desarrollando a la par para crear un conjunto de formas y herramientas para la conexión entre dispositivos IoT; y el análisis y procesamiento de los datos generados a partir de la captura de información de sensores. Algunos de los conceptos que describen las herramientas que apoyan la construcción de estos sistemas son [3]:

- **Conectividad:** La característica principal que hace que un sistema pueda compartir información en la red y permita ser controlado de manera remota, haciendo uso de su propia red local o conectando con servidores externos que almacenen la información.
- **Sensores:** Los elementos que generan los datos a partir de la captura de la información del ambiente de donde se obtienen señales eléctricas analógicas, o incluso, las transforman a señales digitales.
- **Actuadores:** Elementos de diversos tipos que producen una acción eléctrica, mecánica, electromecánica, etc.
- **Microcontroladores:** Son las unidades que coordinan y ejecutan las ordenes programadas según la lectura de los datos recibidos por los sensores y emiten señal en respuesta a esos datos.
- **Procesamiento de datos:** La capacidad de procesamiento y la velocidad de transmisión que tienen los dispositivos IoT para intercambiar datos.
- **Almacenamiento de datos:** Además del tipo de almacenamiento de datos que se hace con hardware a través del uso de memorias propias de los dispositivos, en la actualidad la mayoría de las empresas que generan datos, están haciendo uso de la nube o software de proveedores externos (SaaS) los cuales son servidores con capacidad de almacenamiento y cómputo de

grandes cantidades de datos (Big Data), que se obtienen como resultado luego de su procesamiento y análisis.

- **Monitorización:** Análisis de los datos obtenidos que facilita la toma de decisiones basada en resultados [4].

Los ámbitos en el que se ha visto una mayor aplicación de tecnología IoT para el monitoreo, análisis y control de procesos han sido campos como el industrial, el sector salud y en los hogares. principalmente utilizados para la gestión y monitoreo automatizado y remoto de equipos o máquinas que llevan a cabo algún proceso; para el rastreo o localización de máquinas o equipos médicos; control de calidad de los ambientes industriales, hospitalarios y domésticos, lo que ha servido para mejorar los procesos según las decisiones tomadas basadas en la información obtenida del análisis de los datos recopilados [5], y a su vez permitiendo el control de las variables ambientales, productivas y sanitarias.

Las variables de temperatura e iluminación son las más automatizadas en la domótica para el control racional de la climatización y la electricidad, aunque también es común controlar otros sistemas como el agua e incluso la seguridad. “Con la regulación y el control de los sistemas de calefacción y aire acondicionado, conmutando y regulando la iluminación y controlando los sistemas de alarma, puertas y ventanas, etc. conseguimos adaptar la vivienda a nuestras necesidades de bienestar y al mismo tiempo, conseguimos un importante ahorro en la factura energética” [6].

Con lo anterior, es importante reconocer la relevancia de los sistemas de automatización, principalmente de los aires acondicionados por el impacto negativo que estos generan al medio ambiente, ya que estos climatizadores consumen mucha energía eléctrica, además de que muchos utilizan gas refrigerante, lo cual hace que se emita más Gases de Efecto Invernadero (GEI) a la atmósfera. La acumulación de estos GEI en la atmósfera obstruye la reflexión de la luz solar (radiación de onda corta) aumentando así la absorción y reemisión de la energía (radiación infrarroja). Cuanto más dióxido de carbono haya en la atmósfera, menos energía solar es capaz de salir del ambiente. Un estudio de Ecologistas en Acción (2011), “La calidad del aire en el estado español”, demostró que el uso sistemático de los climatizadores puede aumentar la temperatura en

1,5°C o 2°C en las zonas de ciudad donde estos aparatos abundan, de este modo el aporte negativo sobre el ambiente es sustancial y por lo tanto aumenta el calentamiento global [7][8].

IV. METODOLOGÍA

Análisis de la infraestructura y la tecnología de la clínica.

La primera fase que se llevó a cabo fue de reconocimiento de la infraestructura del establecimiento y de la tecnología existente, con especial atención, el servicio de hospitalización del sexto piso. Se observaron aspectos como el número de habitaciones del servicio, que cuentan con un aire acondicionado de tipo Fan Coil independiente y en buen estado; su distribución en el piso y su tamaño; los elementos de las habitaciones como la posición de la cama, la cercanía del tablero de control eléctrico a la cama, la distribución de las lámparas de iluminación, la distancia entre el control del aire y el panel de control eléctrico, entre otros aspectos que den paso a identificar los mejores puntos de conexión de los módulos para un óptimo funcionamiento.

Análisis de los circuitos de la tecnología existente.

Otro paso que llevó la fase inicial fue el análisis del circuito de control del aire acondicionado, el cual, actualmente funciona con un control del termostato de la marca Honeywell modelo T6861V2WG. Con el análisis realizado se pudieron determinar los puntos de conexión para una correcta integración del módulo inalámbrico de control con la tecnología Honeywell y los estados de actividad de cada uno de esos puntos mientras el aire esta activo e inactivo, como también el tipo de señal que se debe enviar al pin de apagado y encendido para que haya un cambio de estado del aire.

También se estudió el circuito que controla la iluminación y se establecieron los puntos de conexión del módulo inalámbrico de control de la iluminación. Se estudió la posibilidad de que el control de las luces se hiciera por medio de la interrupción de la continuidad del cable de Línea (cable rojo) que es el responsable de alimentar todas las luces a la vez, siendo el neutro quien le da la independencia a cada una de ellas. Para esto se pensó el uso de un relé, el cual permanece cerrado (en continuidad) como estado natural, de este modo se le permite al usuario en la habitación tener un control mecánico de las luces, así, el relé se abrirá cortando el suministro de energía a la iluminación cuando no haya presencia de personas y así estén encendidas las luces o no.

Selección de los microcontroladores para la construcción de los módulos.

Los microcontroladores pensados para desarrollar cada uno de los módulos pertenecen a la familia de microcontroladores Espressif los cuales contienen diferentes sistemas para la medición de señales (ADC, DAC), producción de señales (PWM), diferentes tipos de conexión (I2C, I2S, SPI, UART, WIFI, BLE), arquitecturas de hasta 32 bits, velocidades de procesamiento de hasta 160 MHz, y diferentes capacidades de memorias FLASH y RAM, entre otros sistemas integrados en un solo chip (SoC). Debido a sus capacidades, popularidad, bajo consumo energético y economía. Los microcontroladores seleccionados para el desarrollo del sistema modular son el ESP32 y el ESP8266.

Generalidades del funcionamiento del sistema modular.

El sistema de módulos previstos para dar solución a esta oportunidad de mejora funciona utilizando un protocolo de comunicación inalámbrica llamada ESP-NOW, el cual es un protocolo de comunicación propio de la familia de microcontroladores Espressif. Se pensó en el uso de indicadores del estado de conexión o de enlace entre módulos e indicadores de conexión eléctrica o alimentación para la monitorización de conexión de los módulos. También se ideó el uso de dos botones de configuración para enlazar y desenlazar (SYNC y CLEAR respectivamente) los módulos de control de aire/luces al módulo central de sensado.

Para enlazar los módulos de control de aire/luces al módulo de sensado, se deben de presionar los botones de sincronización denominados como SYNC en ambos módulos a la vez durante más de 5 segundos hasta que el indicador LED de ambos se encuentren encendidos, de esta forma, se mantiene una comunicación activa entre ambos módulos. Cabe aclarar que los módulos de control del aire y el control de luces no se comunican entre sí, esto quiere decir que cada uno de estos solo pueden comunicarse con el módulo de sensado; Si lo que se quiere es desenlazar el sistema, se debe de presionar el botón CLEAR durante 5 segundos o hasta que el indicador de enlace se apague, esto puede hacerse de forma individual por cada módulo.

Cuentan con una etapa de conversión de la energía que hace posible conectar cada uno de los módulos directamente de las fuentes de energía de hasta 220V AC (transformador de energía de 240V a 5V a 700 mA máx.).

La forma en la que se divide el sistema en módulos y la función de cada uno es la siguiente:

Módulo sensorial: Está conectado a un sensor de presencia del tipo RCWL-0516 y es el encargado de procesar los datos del sensor y generar la señal de apagado si no detecta presencia dentro de la habitación por más de 3.5 minutos, de lo contrario envía la señal de encendido a los respectivos módulos que controlan el aire y las luces.

Módulo aire: Su tarea consiste en recibir las señales de apagado y encendido del módulo sensorial y realizar una evaluación del estado del aire a través de la lectura de los niveles lógicos del control del termostato, si la señal que recibe es de encendido y la lectura del control del termostato está en alto, el módulo inalámbrico no realizará ninguna acción; similar a si la señal recibida es de apagado y el aire se encuentra en el mismo estado; pero si la señal recibida es diferente a la lectura del estado del aire, realizará la acción que corresponde según la señal recibida mandando un bajo durante 400 milisegundos como señal de apagado al pin que corresponde al botón de apagado en el control del termostato.

Para la construcción de este módulo se emplearon dos circuitos específicos para la conexión con el control Honeywell. El primer circuito utilizado fue un conversor de nivel lógico debido a que los niveles lógicos utilizados por la tecnología Honeywell van de 0 a 5V, mientras que los niveles lógicos en alto utilizados por los microcontroladores de la familia Espressif son de 3.3V. Este circuito permite la compatibilidad en voltaje entre ambas tecnologías. El segundo circuito utilizado especialmente para el envío de la señal de encendido y apagado funciona con 3 estados lógicos: alto, bajo y alta impedancia; de estos estados lógicos se excluye el estado alto ya que el estado normal del pin del control Honeywell siempre está en alto, por lo que enviar una señal en alto no causa ningún efecto en el control del aire ni representa ninguna acción de interés, además el estado normal de la señal que proviene del módulo de control es alta impedancia lo que lo hace compatible con ambas tecnologías respecto a los niveles lógicos. Es de resaltar que ambas tecnologías están debidamente acopladas por una conexión común (GND).

Módulo luces: Tendrá conectado los actuadores necesarios para controlar el estado de las luces principales de la habitación dependiendo de la señal recibida desde el módulo sensorial. Haciendo referencia al funcionamiento del módulo de control del aire, su lógica de funcionamiento es similar: si la señal recibida es diferente al estado de las luces, el módulo toma la acción correspondiente de forma permanente según la señal recibida hasta que el estado vuelva a cambiar.

Diseño del sistema modular.

El diseño del sistema modular fue realizado en la aplicación web EasyEDA, el cual es un software de diseño y simulación de circuitos online integrado con empresas chinas que ofrecen un catálogo de componentes electrónicos y la fabricación de tarjetas PCB (LCSC y JLCPCB respectivamente) que permiten la manufactura y el ensamble de tarjetas PCB con acabados muy profesionales. Como primer paso del diseño fue la estructuración del esquemático del sistema y por último el diseño de las tarjetas PCB con las respectivas máscaras de componentes.

Construcción del sistema modular.

El primer prototipo del módulo que se construyó fue el módulo correspondiente al control del aire por su grado de complejidad mientras que a la par fue construido el módulo sensorial para probar de forma semiautomatizada la comunicación entre módulos, el accionamiento y ejecución de las señales, con la intención de observar la reacción del control del termostato Honeywell al hecho de que un dispositivo externo tuviera control sobre él.

Posteriormente se construyó el módulo de control de luces, fue integrado con un relé y la adaptación necesaria (aislamiento eléctrico y protección de corrientes de retorno producidas por la bobina del relé) para activar o desactivar los switches de las luces de la parte central de la habitación. Su función principal es interrumpir la continuidad del cable de Línea de la alimentación conectado a los switches mecánicos se encuentran en la habitación para apagar y encender las luces de la zona principal donde el paciente permanece la mayor parte de su tiempo.

Programación de los módulos del sistema y otras características.

Otra de las fases importantes de la construcción del sistema modular fue la programación de cada una de las funciones de cada módulo, la forma y las condiciones que deben de cumplir para enlazarse y mantener una comunicación estable con el módulo central o sensorial, como también la lógica con la que deben operar luego de estar conectados. También se tuvo en cuenta el comportamiento que debe de tomar cada módulo en el posible caso de que ocurran apagones eléctricos, ya que estos son bastante comunes dentro de la clínica ocurridos por condiciones climáticas y otros factores externos. El software utilizado para programar los módulos fue Arduino IDE 2.0.4. Solo se programó dos códigos, uno para el módulo sensorial y el otro para los módulos de control del Aire y las Luces. Dentro del código se puede diferenciar lo que corresponde a cada

tipo de módulo, solo se debe cambiar el tipo de módulo (“AIRE” o “LUCES”) que se va a programar, esto en una línea de código. Los códigos programados se encuentran en el anexo B y C del documento.

Diseño de la carcasa o presentación visual del sistema.

Luego de la construcción y programación de cada uno de los módulos se procedió a diseñar la envoltura o carcasa que protegerá cada módulo del ambiente exterior. Los diseños de las carcasas se realizaron con el Software SolidWorks2020 provisto por la Clínica en el que se tuvieron en cuenta las medidas de cada módulo y la distribución de los elementos como los botones, los indicadores y los puertos de conexión eléctrica. El material usado para la construcción de estas carcasas es el PLA (Ácido Poliláctico), es un polímero utilizado comúnmente para extruir modelos impresos en 3D.

Prueba de usabilidad y de funcionamiento.

La última etapa del desarrollo consistió en hacer pruebas básicas de cada una de las funcionalidades de los módulos, como el enlazamiento exitoso entre los módulos, la comunicación estable en presencia de obstáculos como los muros, el encendido de los indicadores de conexión, eléctrica y de enlace entre módulos, el uso de los botones de sincronización y desenlaces el tiempo de respuesta de la señal de apagado y encendido, la sensibilidad y el alcance del sensor para detectar la presencia de personas. Se hizo una presentación en forma de simulación del funcionamiento del sistema al personal técnico que configurarán los módulos y harán su respectiva instalación en caso de que se decida replicar el modelo en todas las habitaciones del servicio de hospitalización en un futuro.

Las fases empleadas en la construcción del este proyecto de prácticas académicas se aprecian en la imagen mostrada en la figura 1.



Fig. 1. Fases empleadas para el desarrollo del proyecto, enfocadas al cumplimiento de los objetivos específicos.

V. RESULTADOS

Alineando los objetivos específicos con cada uno de los pasos específicos propuestos para la realización del proyecto, y apuntando al cumplimiento del objetivo general, se obtuvieron los siguientes resultados divididos por dichos pasos:

Identificación de requerimientos tecnológicos y de infraestructura.

Análisis de la infraestructura, distribución de habitaciones y la alimentación energética.

Según el análisis hecho de la infraestructura del servicio de hospitalización del piso 6 de la Clínica El Rosario El Tesoro se pudo encontrar que el piso cuenta con 27 habitaciones y con 2 variaciones de tamaño. Se determinó que la distribución de las habitaciones es uniforme, dado el caso y como se puede observar en la figura 2, las habitaciones más grandes se encuentran al mismo lado del corredor que se esté cruzando desde el centro de la estructura mostrada en el plano, es decir, la más grandes se ubican del costado derecho y las pequeñas del costado izquierdo.



Fig. 2. Imagen tomada del panel de las rutas de evacuación del piso 6.

Aunque esto no es de mucha relevancia, sirvió de ayuda para emparejar las habitaciones que comparten la red eléctrica, lo que fue uno de los hallazgos al momento de buscar las redes eléctricas que suministran de energía de cada habitación, en otras palabras, hay habitaciones que comparten el mismo breaker de energía. Sin embargo, cada habitación tiene su propia red eléctrica destinada solo para el aire acondicionado. La distribución de la red eléctrica, como se observa en la figura 3, está marcada de color rojo, y el suministro eléctrico del aire acondicionado se marca de color azul.

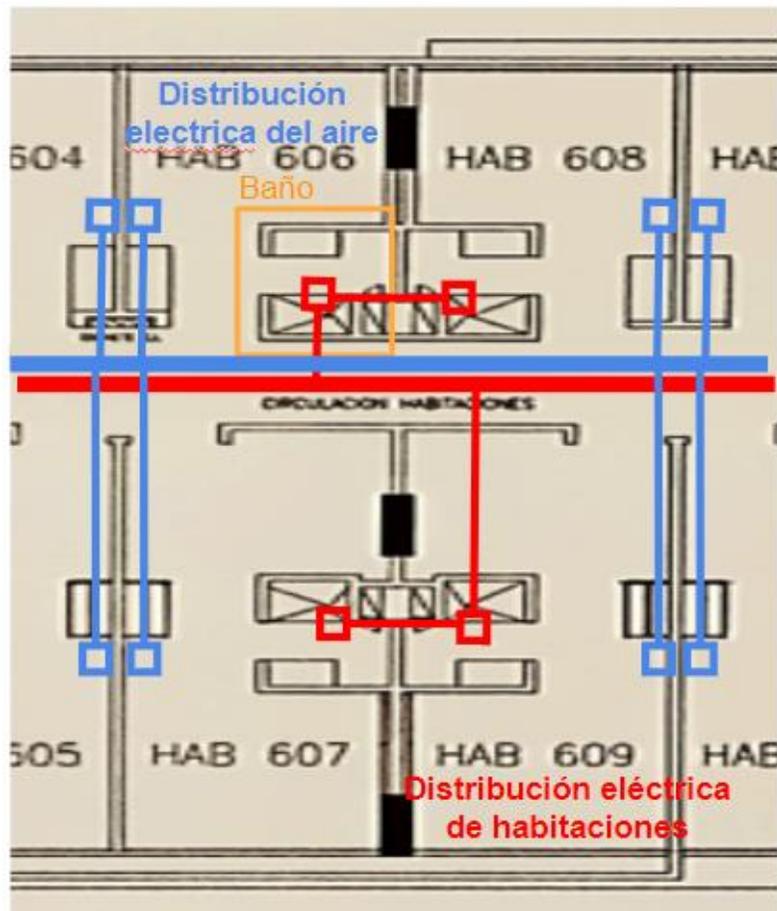


Fig. 3. Distribución eléctrica de las habitaciones. Rojo: suministro de tomas e iluminación. Azul: suministro del aire acondicionado.

Identificar los puntos de conexión energética de las habitaciones.

Luego de que se identificó la distribución general de las redes eléctricas tanto de la iluminación como del aire acondicionado, se hizo un análisis más detallado de la distribución eléctrica de las luces y del aire, a razón de identificar los puntos y nodos de conexión. En las figuras

4 y 5 se encuentran las ubicaciones de los focos lumínicos, del Fan Coil, de los puntos de control o switches del aire y las luces.

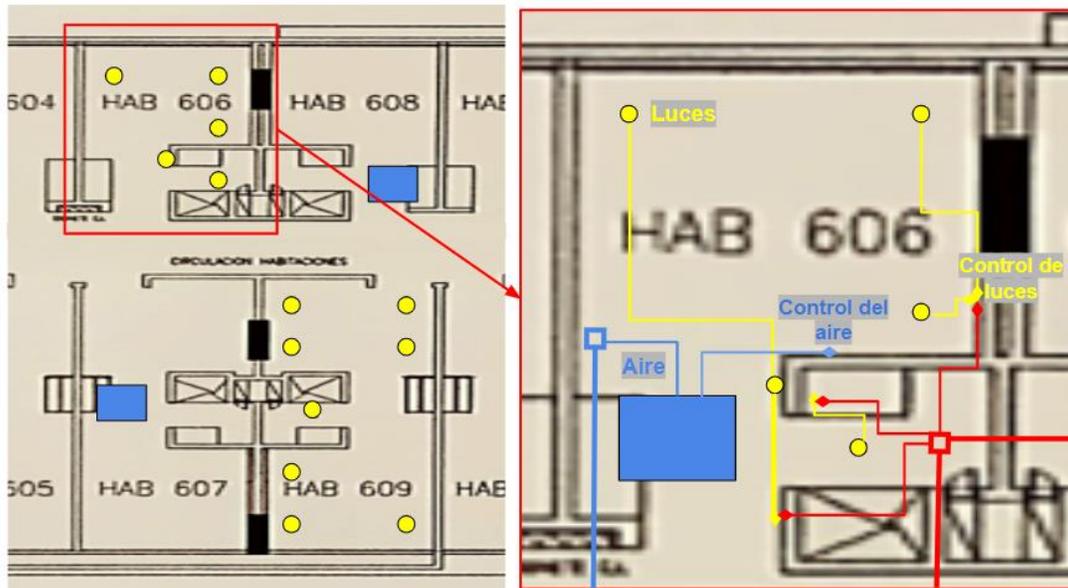


Fig. 4. Distribución y ubicación del aire y las luces dentro de la habitación. Vista esquemática.



Fig. 5. Distribución de los elementos de control del aire (azul) y luces (amarillo) dentro de la habitación.

Análisis del circuito de control Honeywell y circuito de control de luces.

Con el análisis realizado se pudieron determinar los puntos de conexión de la tecnología Honeywell para una correcta integración con el módulo inalámbrico de control. Los estados de actividad de cada uno de esos puntos mientras el aire está activo e inactivo, como también el tipo de señal que se debe enviar al pin de apagado y encendido para que haya un cambio de estado del aire son representados en la figura 6 y 7.

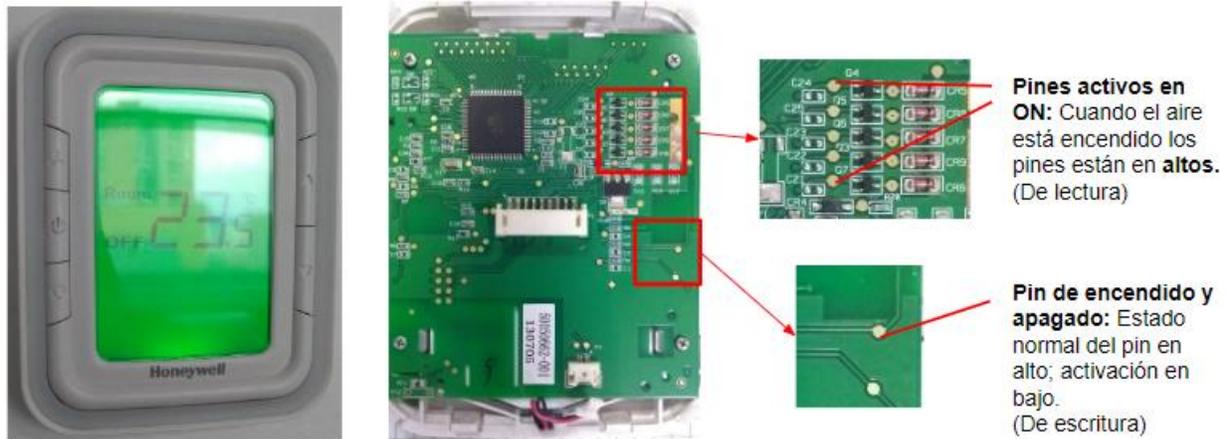


Fig. 6. Pines de conexión de la tecnología Honeywell modelo T6861V2WG para integración del módulo de control inalámbrico del aire acondicionado.

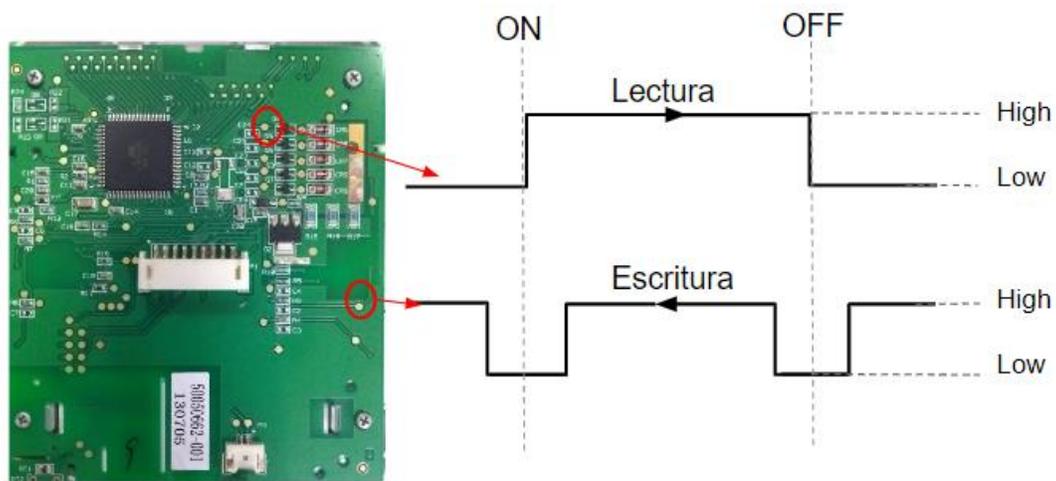


Fig. 7. Comportamiento digital de los pines de integración para la lectura y escritura de señales de control.

Además, el análisis del circuito de iluminación no fue dificultoso ya que era evidente por los colores utilizados en el cableado, de igual manera se empleó un multímetro para tener mayor certeza de cuales cables estaban activos y cuáles no. Así se pudo determinar que la mejor forma de conectar

el módulo de control de luces es por medio de la interrupción de la continuidad del cable de Línea de color rojo, el cual es el encargado de suministrar energía a todos los focos. Se muestra en la figura 8 los cables conectados al switch que controla la iluminación de la habitación.

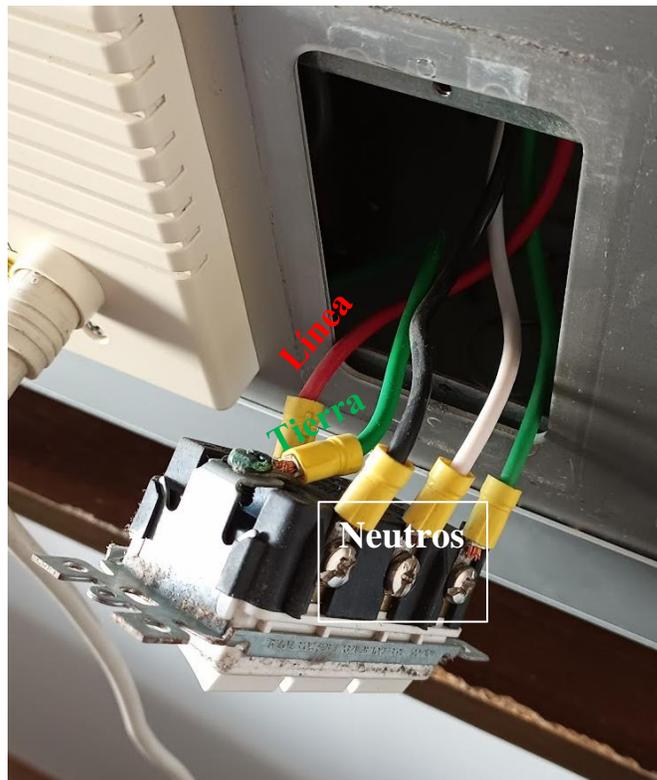


Fig. 8. Puntos de conexión para el control de la iluminación.

Selección de los dispositivos a usar que permita la interconexión inalámbrica.

Los dispositivos seleccionados para la construcción del sistema son los microcontroladores ESP32 y ESP8266, junto con otros dispositivos de uso común como resistencias, capacitores, transistores, LEDs, transformadores, relés, botones, diodos, optoacopladores y compuertas lógicas de 3 estados, etc.

Diseñar del sistema de automatización de luces y del aire acondicionado.

El diseño del sistema modular fue realizado en la aplicación web EasyEDA, el cual es un software de diseño y simulación de circuitos online integrado con empresas chinas que ofrecen un

catálogo de componentes electrónicos y la fabricación de tarjetas PCB (LCSC y JLCPCB respectivamente) que permiten la manufactura y el ensamble de tarjetas PCB con acabados muy profesionales. Como primer paso del diseño fue la estructuración del esquemático del sistema y por último el diseño de las tarjetas PCB con las respectivas máscaras de componentes. El diseño de las tarjetas PCB de cada uno de los módulos se hicieron simultáneamente. El resultado de este trabajo se aprecia en la figura 9. El esquemático de este proyecto puede encontrarse en el anexo A.

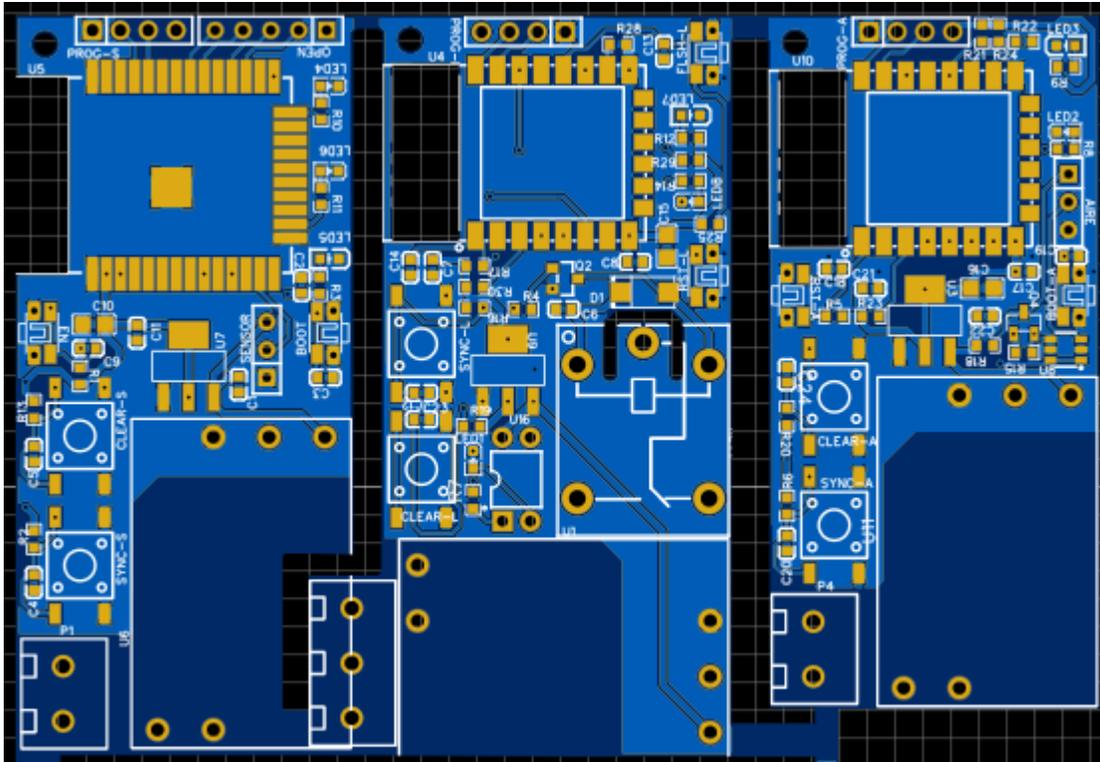


Fig. 9. Diseño de las tarjetas PCB del sistema modular. De izquierda a derecha módulo sensorial, módulo control de luces y módulo control del aire acondicionado.

Estas tarjetas no pudieron ser materializadas debido a que el tiempo de manufactura superaba el tiempo máximo a la terminación del proyecto, por lo que se optó por construir cada uno de los módulos de una forma manual utilizando tarjetas electrónicas de ensamble universal o bien conocidas como baquelitas, y utilizando los módulos de desarrollo prefabricados los cuales contienen todas las adaptaciones y el hardware necesario para ser programados y fácilmente ensamblados según las necesidades. En este caso, se utilizaron los kits de desarrollo ESP32 DevKit V1, ESP8266 NodeMCU V1 y ESP8266 D1 MINI WeMos, estos se visualizan en la figura 10 de izquierda a derecha respectivamente.



Fig. 10. Kits de desarrollo basados en ESP32 y ESP8266.

Diseño del módulo sensorial.

Se diseñó y construyó el módulo sensorial al cual se le integró un sensor de presencia RCWL-0516. Aunque en la figura 11 no se muestre el sensor utilizado ya que está cubierto por la carcasa construida, en la figura 12 se logra ver su acabado final.

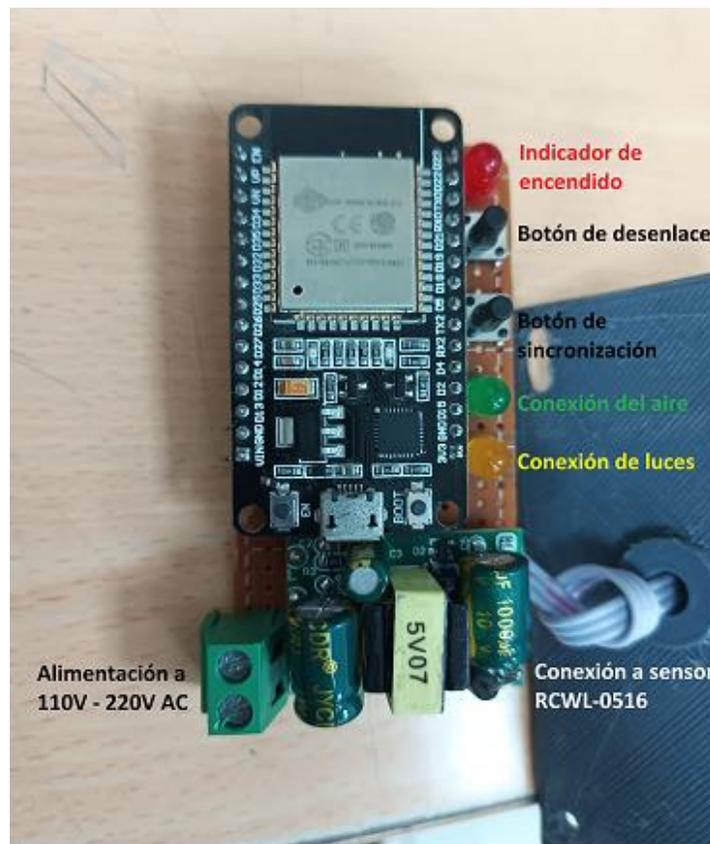


Fig. 11. Módulo sensorial.



Fig. 12. Módulo sensorial con cubierta final impresa en 3D.

Diseño del módulo de control de luces.

Se diseñó y construyó el módulo de control de luces al cual se le integró un relé normalmente abierto. Las adaptaciones del control del relé como el aislamiento eléctrico y la protección de corrientes de retorno causadas por la bobina del relé se encuentran debajo del kit de desarrollo. En la figura 13 se muestra el módulo construido. A pesar de que en la cubierta del acabado final se perciba un error de ortografía en el idioma inglés, la instalación del módulo quedará oculta a la vista del usuario de la habitación, por lo que el error también.

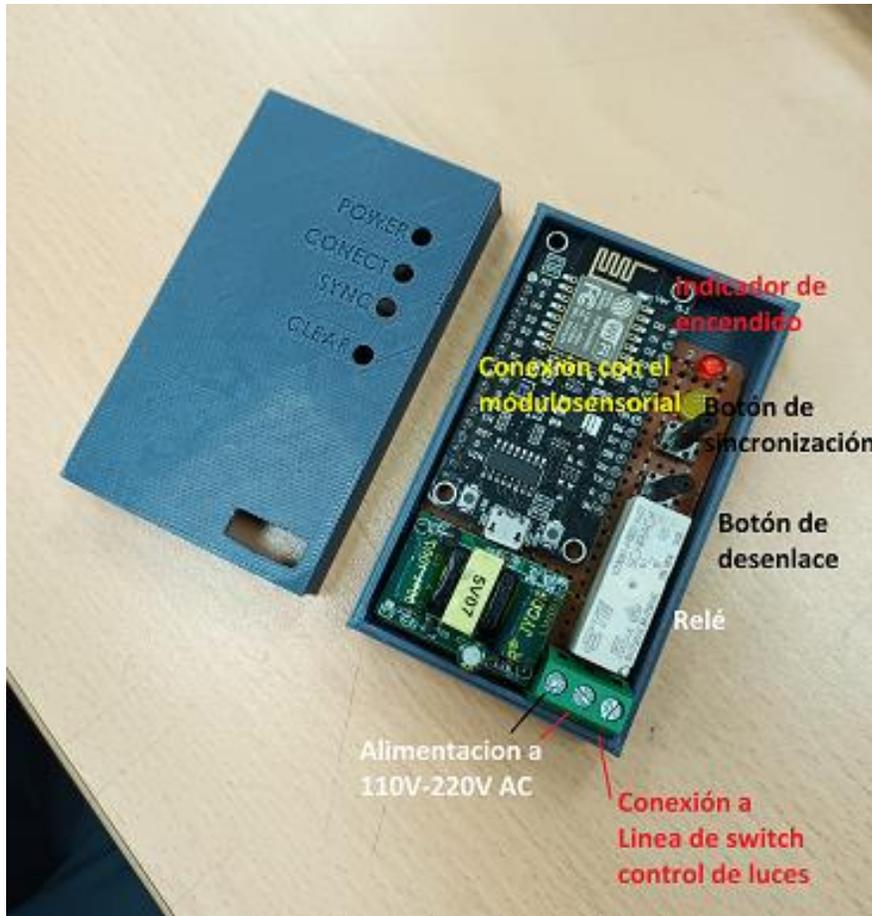


Fig. 13. Módulo de control de luces con cubierta final impresa en 3D.

Diseño e integración del módulo de control del aire acondicionado.

Para la construcción del módulo de control del aire acondicionado se emplearon circuitos muy específicos para la lectura del estado del control Honeywell y para el envío de la señal de apagado. El primer circuito utilizado fue un conversor de nivel lógico debido a que los niveles lógicos utilizados por la tecnología Honeywell van de 0 a 5V, mientras que los niveles lógicos en alto utilizados por los microcontroladores de la familia Espressif son de 3.3V. Este circuito permite la compatibilidad en voltaje entre ambas tecnologías, de este modo resulta seguro evaluar el estado de actividad del aire acondicionado. El circuito descrito se ensambló con el transistor 2N7000, tal y cómo se muestra en la figura 14.

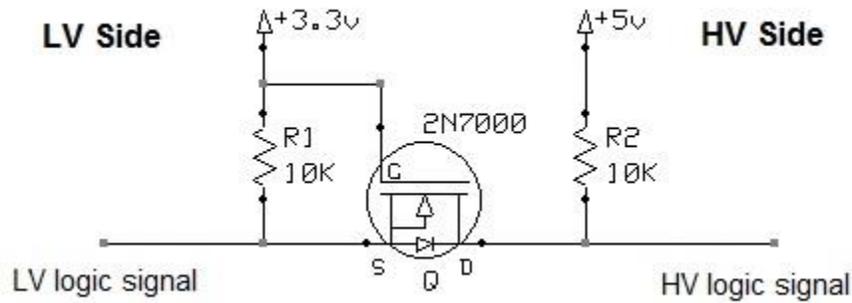


Fig. 14. Circuito de conversión de niveles lógicos 3.3V a 5V.

Nota: fuente <https://www.electronicoscaldas.com/es/content/9-convertidor-activo-bidireccional-de-niveles-de-voltaje-logicos>

El segundo circuito fue utilizado para el envío de la señal de encendido y apagado funciona con 3 estados lógicos: alto, bajo y alta impedancia; de estos estados lógicos se excluye el estado alto ya que el estado normal del pin del control Honeywell siempre está en alto, por lo que enviar una señal en alto no causa ningún efecto en el control del aire y tampoco es de interés, además el estado normal de la señal que proviene del módulo de control es Alta impedancia debido a la configuración utilizada lo que no causa efecto alguno sobre el control Honeywell de esta manera, existe compatibilidad entre ambas tecnologías con respecto a los niveles lógicos. Para haber logrado esto, se utilizó el circuito integrado SN74LVC125 con la configuración que se muestra en la figura 15. Es de resaltar que ambas tecnologías están debidamente acopladas por una conexión común (GND).

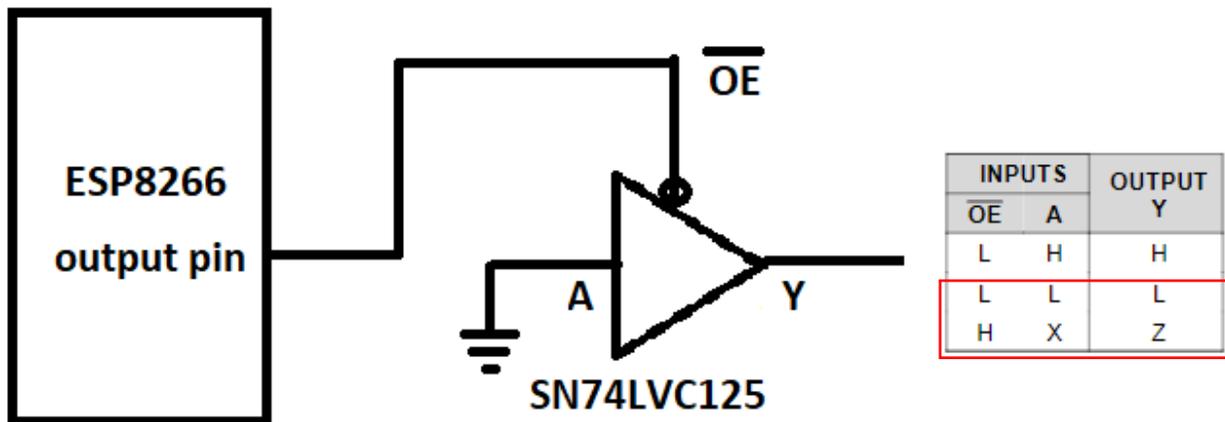


Fig. 15. Circuito de envío de señal de apagado/encendido del aire acondicionado.

Nota: fuente <https://electronics.stackexchange.com/questions/486903/esp32-output-pin-internal-pull-up-pull-down-resistor/> La imagen fue editada y adaptada para la ilustración.

De este modo, el módulo construido sin la etapa de transformación de energía se muestra en la figura 16. En la figura 17 también se puede apreciar el módulo con la terminación final, la cual incluye la etapa de transformación de energía, la cubierta impresa en 3D y un cambio de colores en los cables de conexión a la tecnología Honeywell.

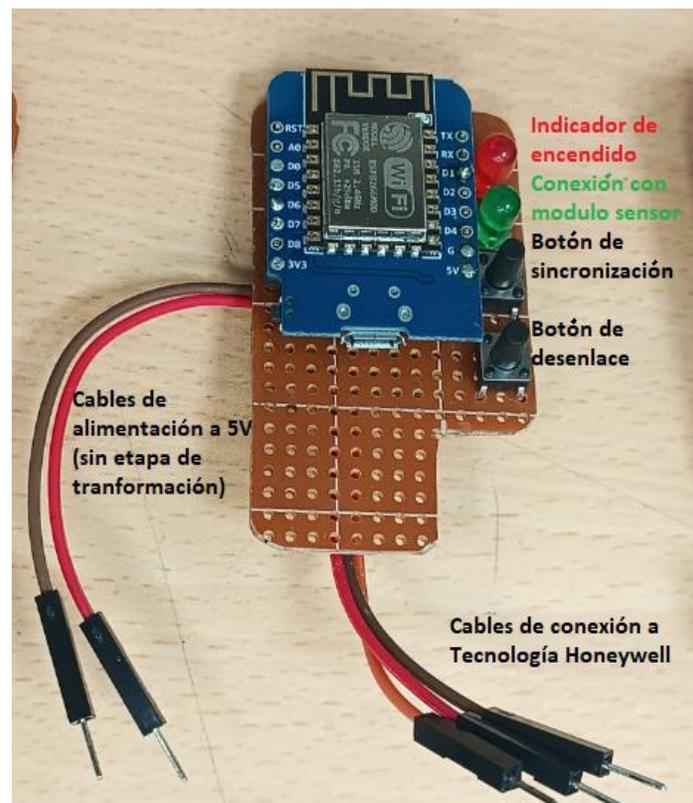


Fig. 16. Módulo de control del aire sin la etapa de transformación de la energía.

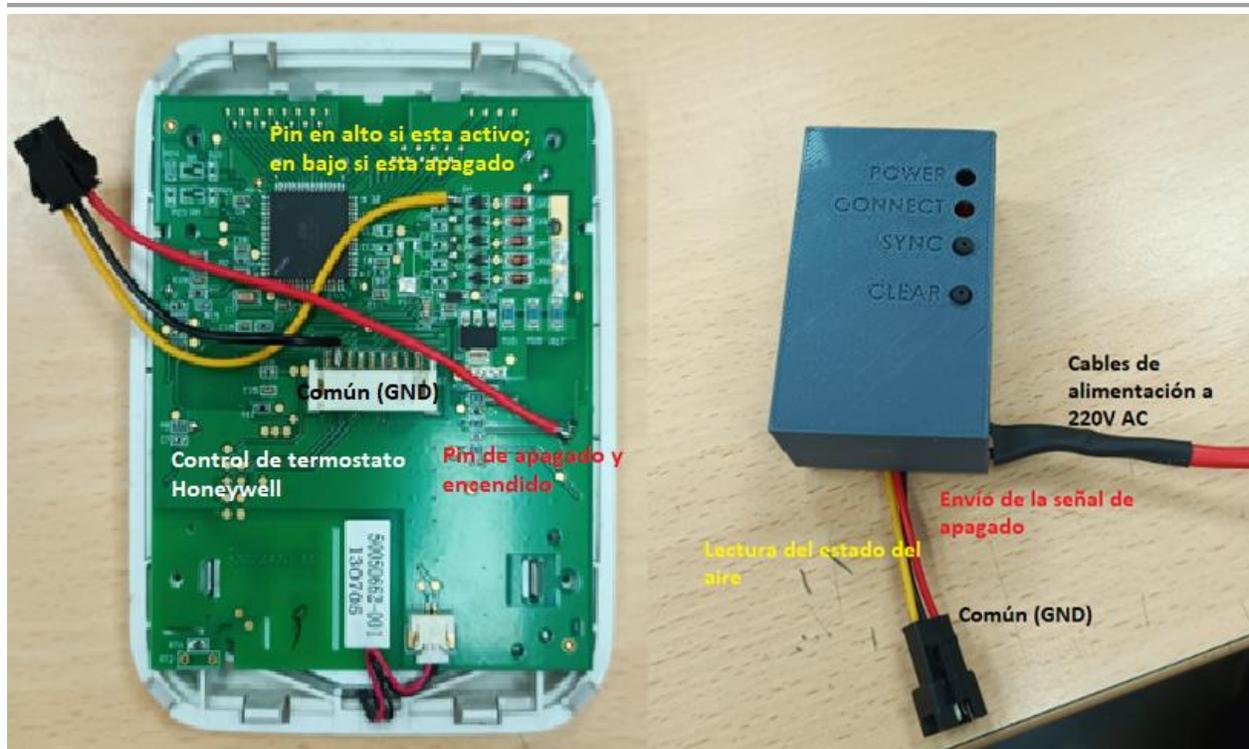


Fig. 17. Módulo de control del aire con cubierta final impresa en 3D.

Evaluar la funcionalidad y usabilidad del sistema.

Implementación del sistema modular.

Inicialmente el sistema fue probado de forma parcial conforme se iba desarrollando cada módulo. La primera prueba se hizo semiautomatizada para evaluar algunas funcionalidades como la sincronización a través del uso de los botones pulsadores, la estabilidad de la conexión, la lectura y ejecución de las señales, principalmente del módulo del control del aire acondicionado, aquí se pudo observar que los circuitos de lectura y de envío de la señal de apagado funcionaban correctamente por lo que se decidió continuar con la construcción de los módulos (ver figura 18).



Fig. 18. Prueba parcial semiautomatizada del módulo de control del aire.

Posterior a eso se procede con la construcción del módulo de control de luces y se hacen a respectivas pruebas, sin embargo, estas no se pudieron hacer en las habitaciones del servicio de hospitalización de la clínica, sino que se hicieron en una habitación residencial. De estas pruebas, se tomaron 8 muestras y se contabilizó el tiempo desde el último movimiento consciente hasta el apagado del sistema. Salieron los siguientes resultados (TABLA I) para un tiempo de espera programado de 3.5 minutos.

TABLA I: RESULTADOS DEL TEST DE FUNCIONAMIENTO DEL MÓDULO DE CONTROL DE LUCES

Intento	Tiempo de apagado (mm:ss)	Desviación en segundo (s)
1	03:40	10
2	03:35	5
3	03:38	8
4	03:38	8
5	03:37	7
6	03:38	8
7	03:31	1
8	03:39	9
Desviación media		7

Realización de ajuste de parámetros de funcionamiento del sistema.

Según el análisis realizado de los datos, se determinó que no es necesario ajustar los parámetros de funcionamiento del sistema.

Aplicación de pruebas de usabilidad del sistema.

La prueba de usabilidad fue realizada a modo de simulación con el personal técnico quienes operarán y configurarán el sistema en el servicio de hospitalización si se decide replicar e instalar el modelo en todas las habitaciones. Inicialmente se conectaron cada uno de los módulos tanto a la corriente eléctrica como al elemento de la variable a controlar, es decir, el módulo de control del aire se conectó al control del termostato Honeywell y el módulo de control de las luces se conectó a una lámpara. El estado natural del sistema es mantener el estado de las variables encendidas si estas lo están.

En la prueba, se demostró la forma de configuración del sistema de la siguiente manera:

- Se verificó visualmente a través del encendido del LED indicador color rojo (POWER) que cada módulo estuviera encendido o alimentado por la corriente eléctrica.
- Se verificó que el estado de las variables a controlar estuviera ON, es decir, aire y luces encendidas.
- Primero se enlazó el módulo de control del aire al módulo sensor de presencia presionando los botones correspondientes de sincronización (SYNC) durante más de 5 segundos.
- Se verificó que el enlace fue exitoso a través del encendido del LED indicador color verde en cada uno de los módulos relacionados.
- Se procedió a enlazar el módulo de control de luces al módulo sensor de presencia de la misma manera, presionando durante más de 5 segundos el botón SYNC en cada módulo.
- Se verificó el enlazamiento con el encendido del LED indicador color amarillo presente en cada módulo relacionado.

-
- Se interrumpió la alimentación en todo el sistema para verificar que los módulos se enlazan automáticamente sin necesidad de presionar nuevamente el botón SYNC al reconectar el sistema nuevamente a la energía
 - Por último, se hizo el proceso de desenlace de cada uno de los módulos presionando el botón CLEAR durante más de 5 segundos. No fue necesario hacerlo sincrónicamente entre los módulos dado que cada módulo se desenlaza individualmente borrando los registros de los módulos enlazados previamente.
 - Se verificó que todos los LED indicadores de conexión de enlace están apagados.

Mientras se hacía esta prueba, se les fue pidiendo a algunos de los asistentes que participaran en la verificación de los estados de las variables, en la sincronización entre módulos, la verificación visual del estado de enlace y en el proceso de desenlace de todo el sistema.

Dado que la prueba de usabilidad se hizo en un lugar de espacio reducido donde el número de asistentes fueron 7, más quien presentaba el proyecto (8 en total), no fue posible alcanzar el estado de quietud para que el sistema detectara ausencia en el espacio de prueba y pudiera enviar la señal de apagado a los módulos de control de las variables para verificar con los asistentes en funcionamiento del sistema. Sin embargo, la prueba de funcionamiento se realizó individualmente previo a la prueba de usabilidad tal y como se explicó anteriormente, donde se verificó su funcionamiento.

Realización de mejoras y entregar del sistema funcional.

Las mejoras que se deben hacer al sistema son propias de una actualización de versión, ya que este caso al ser un prototipo del sistema, y al estar construido manualmente, está abierto a mejoras de funcionamiento y de usabilidad.

VI. ANÁLISIS

Sobre el primer objetivo específico

Uno de los planteamientos realizados para darle cumplimiento al primer objetivo específico fue determinar la distribución de las habitaciones, planteamiento que hubiera tenido mayor relevancia si al sistema se le hubiera dado la funcionalidad de conectarse a internet y de permitir el acceso a los datos de los estados de las variables de control a través de un servidor web almacenado en el módulo central. Con el análisis de la distribución de habitaciones viene consigo el análisis del alcance que tienen los nodos de conexión a internet distribuidos por el piso del servicio de hospitalización, y a su vez, la cobertura que los usuarios para conectarse fácilmente a la red wifi. Esto se refiere a que esta funcionalidad es una gran oportunidad de mejora del sistema en una posible versión más actualizada, debido a que no fue posible desarrollar por las limitaciones de tiempo en el desarrollo del proyecto.

La identificación de los requerimientos tecnológicos y de infraestructura de una con el nivel de complejidad de la Clínica el Rosario siempre será un desafío si no se cuenta con la información necesaria, aún más si no se cuenta con el tiempo adecuado para realizar estos análisis desde cero debido a que el tiempo empleado en entender, por ejemplo, la distribución de las redes eléctricas y la infraestructura en general fueron largos, debido a que no se contó con los planos de la red eléctrica desde el principio, aunque fueron solicitados en varias ocasiones, lo que sumado a otras cuestiones, retraso en parte el desarrollo del proyecto.

El espacio para el módulo de control del aire, el cual tiene que coexistir con el mismo espacio que ocupa el control de termostato Honeywell es escaso, y varía de habitación en habitación. Por lo que es necesario que las próximas versiones del módulo de control inalámbrico se utilicen ambas caras de la tarjeta para ensamblar los componentes relacionados con la intención de que pueda ser mucho más compacta para que se pueda adaptarse fácilmente al espacio compartido con la tecnología del control Honeywell.

Dado que se pudo integrar con éxito el módulo de control de aire con la tecnología Honeywell, uno de los posibles errores que se pudo percibir es que cuando el módulo de control inalámbrico es desconectado de la energía y el control Honeywell aún sigue alimentado y conectados entre sí, el control Honeywell entra sin cesar al modo de configuración debido a que el acople de tierras hace que el control se envíe una señal de un cero o bajo prolongado haciendo que entre en esta configuración. Teniendo presente que el botón de encendido del control Honeywell funciona de dos maneras: la primera es cambiar el estado de encendido o apagado con un pulso en bajo por menos de 8 segundos; el otro modo de funcionamiento es para acceder al modo de configuraciones y se logra manteniendo en botón de encendido presionado por más de 8 segundos. Aunque esto no trae consecuencias negativas al sistema de control a largo plazo, sirve de indicador de que el módulo de control de aire está desconectado del suministro eléctrico. Una de las propuestas para corregir el error es switchear el acople de tierra con un transistor que se active si el módulo control del aire está activo, es decir, que el emisor y el colector del transistor estén conectados respectivamente a cada una de las tierras y que la base del transistor se sature mientras el módulo de control de aire esté activo.

Inicialmente, para el control de la iluminación se había planteado el uso de varios relés (tantos relés como focos hubiese) pero teniendo en cuenta que la intención principal es apagar todas las luces, se decidió que lo óptimo era utilizar un solo relé encargado de la tarea conjunta.

Sobre el segundo objetivo específico

El diseño realizado con la aplicación web EasyEDA, además de ser un diseño muy profesional y funcional (dadas las pruebas hechas), puede ser aún mejor utilizando otro tipo de componentes más compactos y que haya una mejor distribución de los mismos. En el caso del módulo de control de luces, utilizar un relé de estado sólido que además de ser más compacto que un relé mecánico, es silencioso, de esta manera se evita el sonido del “click” del relé tradicional cuando se activa. Redistribuir los componentes, principalmente en el módulo de control del aire y utilizando la capa inferior puede ayudar a reducir el tamaño del módulo para que se adapte mejor a los espacios reducidos y compartidos donde debe existir.

La posible manufactura de las placas PCB diseñadas hubiera agilizado el tiempo que tarda el ensamble manual de cada uno de los componentes de cada módulo.

La utilización de kits de desarrollo, aunque son muy flexibles para el proceso de prototipado con ayuda de una protoboard para probar funcionalidades individuales, ocupan mucho espacio y dificulta el ensamble del resto de componentes cuando se va a utilizar como componente principal en la baquelita (tarjeta universal). Otra de las posibles dificultades es que, si se requiere hacer una actualización del código para mejorar las funcionalidades desde el software, se vuelve complejo desensamblar la tarjeta para liberar el espacio de la conexión del puerto USB, además de que puede causar daños irreparables en el hardware durante el proceso.

Los componentes utilizados para el ensamble del módulo de control del aire fueron adaptados varias veces según la disponibilidad de los componentes en la región, es decir. Hay ocasiones en que los mejores componentes no se encuentran disponibles en el área de ubicación, y se debe de utilizar componentes homólogos que se encuentren en inventario en alguna de las tiendas de electrónica más cercanas. Con esto hago referencia al MOSFET BSS138 que fue el primero en ser propuesto para el convertidor de nivel lógico y tuvo que ser cambiado primero por el 2N7002 y luego por el 2N7000 como último recurso sin olvidar que el primero era el que presentaba mejores propiedades eléctricas para la función requerida; ocurrió similar a la compuerta de 3 estados SN74LVC125, que en este caso se tuvo que utilizar solo una de las compuerta de un circuito integrado que contenía 4 compuertas, lo que hace que el tamaño y el gasto del recurso sea innecesario pero obligatorio ya que era el único componente de ese tipo en existencia en la región.

Algo que no se tenía contemplado en la línea de tiempo del proyecto fue el diseño en 3D de las carcasas de cada uno de los módulos, lo que fue algo de última hora y que al final le dio una mejor presentación al desarrollo del proyecto pero que también empleó una gran cantidad de tiempo.

Sobre el tercer objetivo específico

Al observar la TABLA I obtenida de las pruebas del funcionamiento del módulo de control de luces, se observa que la desviación promedio fue de 7 segundos por encima del tiempo que debe durar el módulo en enviar la señal de apagado luego de que el módulo sensorial deja de sentir presencia dentro de la habitación. Esta fue una prueba donde los datos obtenidos fueron son una

aproximación de los datos reales ya que no es una medida exacta que se pueda calcular debido a que no se sabe exactamente en qué momento el sensor deja de detectar presencia luego de que se empieza a medir el tiempo. Los datos obtenidos sirvieron para determinar que el tiempo aproximado es correcto y que no necesita un ajuste para aumentar y disminuir el tiempo de tolerancia de la señal de apagado. Es de recordar que la ventana de tiempo se realiza para disminuir los falsos positivos en caso de que el personal dentro de la habitación deje de moverse durante este tiempo.

Aunque las pruebas de usabilidad realmente fueron pocas debido a que inicialmente se tenía pensado realizar un monitoreo remoto a través de la conexión a internet y de un servidor almacenado en el módulo central para el monitoreo remoto constante, solo se pudo hacer una demostración pública del funcionamiento del sistema una sola vez, en la que se mostró cómo funciona el sistema y cómo enlazar cada uno de los módulos del sistema al módulo central o de sentido. No obstante, la prueba o la demostración pública fue exitosa, no se encontraron fallas que requieran de un ajuste, pero sí de oportunidades de mejora como la adición de nuevas funcionalidades como la del servidor web y la reducción del tamaño del módulo de control del aire y el cambio de relé del módulo de control de luces.

VII. CONCLUSIONES

El sistema desarrollado en este proyecto cumple con su función principal que es regular el uso de las luces y el aire acondicionado en el momento en el que no se encuentra presencia de personas en las habitaciones, esto para reducir el gasto energético y disminuir el impacto medio ambiental. Está alineado con las políticas de mejoramiento continuo y de gestión ambiental que definen las operaciones y estrategias que ayudan a proteger el medio ambiente, además de que ayuda a la mejora del desempeño ambiental de la clínica, lo que se encuentra alineado con los requerimientos de la Norma ISO 14001:2015.

Ya que dentro del Plan De Mitigación De Huella De Carbono (código EST-SGA-PL-04) establecido por la clínica, las estrategias planeadas se centran en la automatización de las Unidades Manejadoras de Aire (UMAS) de la mayoría de los servicios, pero no se encuentran incluidas el servicio de hospitalización, y además de que el hecho de que sea posible la implementación en masa del sistema desarrollado en este proyecto en todas las habitaciones, para la automatización de los aires acondicionados, puede representar un aporte significativo en la reducción del consumo energético, ya que el sistema es de bajo consumo energético y previene de que hayan gastos innecesarios del consumo producido, no solo de los aires acondicionados, sino también de las luces.

La integración de este tipo de desarrollos a tecnologías existentes que no tiene la capacidad de ser controlados ni monitorizados remotamente, son de bastante utilidad para extender la vida útil y funcional de esa tecnología ya existente, ya que se aplaza la necesidad de cambiar sistemas totalmente funcionales por tecnologías de vanguardia con las misma capacidad de conexión a internet y de operación remota, lo que significa que es posible darle ese tipo de propiedades con una corta inversión protegiendo el medio ambiente de los desechos que se producen con los cambios de tecnología por la obsolescencia.

Este tipo de sistemas adaptativos e integrativos a cualquier otro sistema son bastante flexibles, debido a que pueden tener las características y propiedades que se requieran según la demanda requerida y los sistemas que se quieran controlar remota y automáticamente.

El sistema de automatización desarrollado puede ayudar al personal asistencial de la clínica a enfocarse en tareas importantes, ya que el control manual de la iluminación y el aire acondicionado en las habitaciones pueden ser distractores y generar preocupaciones. Además, el sistema evita el desgaste que se produce al tener que verificar constantemente si las variables de control están inactivas en las habitaciones que no están en uso.

Este proyecto puede abrir paso a la investigación, el desarrollo y la innovación (I+D+i) de la tecnología integrada en el ámbito clínico, reforzando la idea de competitividad de la institución a nivel nacional e internacional [9], transformando el conocimiento en desarrollo tecnológico, lo que se traduce en generación de riqueza a través de la productividad y la mejora los servicios de la clínica, que al poner a disposición de la sociedad, diferencia y agrega valor a la Clínica El Rosario.

REFERENCIAS

- [1] R. Howe, ISO 14001: the green standard; in Computer, vol. 30, no. 11, pp. 133-134, Nov. 1997, doi: 10.1109/2.634868.
- [2] ¿Qué es el internet de las cosas (IOT)? (no date) ¿Qué es el Internet de las cosas (IoT)? | Oracle Colombia. [En Línea]. Disponible en: <https://bit.ly/3GzJuBX>
- [3] ¿Qué es iot y cómo funciona?: SAP insights (no date) SAP. [En Línea]. Disponible en: <https://bit.ly/3ZSv72l>.
- [4] ¿Qué es SAAS? software como servicio: Microsoft azure (no date) Software como servicio | Microsoft Azure. [En línea]. Disponible en: <https://bit.ly/3UoRxai>
- [5] NexusAdmintraIntegra (2022) 7 aplicaciones de Iot Industrial, Nexus Integra. [En línea]. Disponible en: <https://nexusintegra.io/es/7-aplicaciones-del-industrial-iot/>
- [6] Sarachu, E. (2022) ▷domótica ¿Qué es la domótica? ¿Cómo funciona?, Eficiencia Energetica. [En línea]. Disponible en: <https://e-ficiencia.com/domotica-que-es-y-como-funciona/>
- [7] Argemi, A. (2013) 7 Razones Para no Usar El Aire Acondicionado, El País. [En línea]. Disponible en: <https://bit.ly/40YaDXl>
- [8] El Impacto Aire Acondicionado en el medio ambiente- Razones (2019) Suelo Radiante Alicante. [En línea]. Disponible en: <https://www.sueloradiantealicante.es/impacto-aire-acondicionado/>
- [9] Clínica El Rosario. Centro de Innovación e Investigación en salud, Clínica el Rosario. [En línea]. Disponible en: <https://bit.ly/41lscQY>

ANEXOS

Anexo A. Proyecto desarrollado en EasyEDA.

El desarrollo del proyecto puede encontrarse en:

<https://oshwlab.com/clinicaelrosario8/automatizacion-de-habitaciones-6to-piso>

Anexo B. Código de programación para el módulo de sensado

```
/*  
Módulo Sensor  
*/  
  
// Librerías  
#include <esp_now.h>  
#include <WiFi.h>  
#include <EEPROM.h>  
  
// Variables y definiciones  
  
#define syncBUTTON 26 // Pin del botón de sincronización  
#define clearBUTTON 33 // Pin del botón de borrado  
#define syncAireLED 2 // Pin del led Verde de conexión  
#define syncLucesLED 13 // Pin del led amarillo de las luces  
#define sensor 14 // Pin del sensor  
  
bool aireConectado = 0; // Estado de conexión del aire  
bool lucesConectado = 0; // Estado de conexión de luces  
bool currentStatus = 1; // Estado de activación  
bool OnOff; /// Estado del sensor  
  
int timeWaitSec = 600; // Tiempo de espera para mandar la señal de apagado  
  
unsigned int countSync = 0; // Contador para sincronizar los módulos (5 sec)  
unsigned int countClear = 0; // Contador para borrar los módulos (5 sec)  
unsigned int countSense = 0; // Cuenta el tiempo de espera mientras se manda la  
señal de apagado y el aire enciende (pulso LOW 400ms)  
unsigned int countSense2 = 0; //Cuenta la frecuencia de lectura del sensor y  
envío de los datos (330ms)  
  
int selfChannel = 1; // Canal de transmisión
```

```
// Dirección MAC de comunicación abierta
uint8_t broadcastAddressX[] = {0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF}; // Dirección
MAC de transmisión abierta (todos)

enum MessageType {SYNC, DATA,}; // Tipo de mensajes SYNC mientras se presiona el
botón de sincronización; DATA para los mensajes de transmisión de estado del
aire/luces
MessageType messageType;

// Estructura de datos para la sincronización
typedef struct struct_sync {
    uint8_t messageType; // SYNC, DATA
    char moduleType[7]; //AIRE, LUCES, MASTER
    uint8_t moduleChannel; // 1
    bool conected; //0:Disconected, 1:Connected
    uint8_t statusOnOff; //0:OFF-DISABLE, 1:ON-ENABLE
} struct_sync;

// Estructura de datos para envío de datos
typedef struct struct_data {
    uint8_t messageType; // SYNC, DATA
    char moduleType[7]; //AIRE, LUCES, MASTER
    bool conected; //0:Disconected, 1:Connected
    uint8_t statusOnOff; //0:OFF-DISABLE, 1:ON-ENABLE
} struct_data;

// Estructura de datos para guardar el tipo de módulo
struct eeprom_moduloInfo {
    uint8_t macAddress[7]; // SYNC, DATA
    char moduleType[7]; //AIRE, LUCES, MASTER
    uint8_t moduleChannel; // 1
    bool conected; //0:Disconected, 1:Connected
};

// Se crean objetos estructurados
struct_sync syncData;
struct_data myData;
eeprom_moduloInfo modInfo;

// Información del par de emparejamiento
esp_now_peer_info_t peerInfo; // Estructura para registro del módulo en espnow
esp_now_peer_info_t peerToClean; // Estructura para eliminar TODOS los módulos de
espnow

//-----
```

```
// Temporizador para ejecución del código loop cada 1 ms
hw_timer_t *Timer0_Cfg = NULL;
bool bandTime = 0;

void IRAM_ATTR Timer0_ISR(){
    bandTime = 1;
}

void init_Timer(){
    // Función del temporizador
    Timer0_Cfg = timerBegin(0, 80, true);
    timerAttachInterrupt(Timer0_Cfg, &Timer0_ISR, true);
    timerAlarmWrite(Timer0_Cfg, 1000, true); // 1000: 1ms (1000000us -> 1000ms ->
1s)
    timerAlarmEnable(Timer0_Cfg);
}
//-----

// Procesa los mensajes recibidos de los módulos aire/luces
void onDataRecv(const uint8_t *mac_addr, const uint8_t *incomingData, int len){
    uint8_t type = incomingData[0];
    switch (type) {
        case SYNC :
            memcpy(&syncData, incomingData, sizeof(syncData));
            if(!aireConectado || !lucesConectado){
                if (!esp_now_is_peer_exist(mac_addr)){
                    registerPeer(mac_addr, syncData.moduleType, syncData.moduleChannel);
                    saveInEEPROM(mac_addr);
                }
            }
            break;
    }
}

// Registra el módulos sincronizado o guardado en la memoria EEPROM
void registerPeer(const uint8_t *mac_addr, char tipoModulo[], uint8_t channel){
    memcpy(peerInfo.peer_addr, mac_addr, 6);
    peerInfo.channel = channel;
    peerInfo.encrypt = false;
    if (esp_now_add_peer(&peerInfo) != ESP_OK){
        Serial.println("Failed to add peer");
        return;
    } else {
        Serial.print("Peer module ");
        Serial.print(tipoModulo);
    }
}
```

```
    Serial.print(" added");
  }
}

// Guarda el módulo sincronizado (aire/luces) en la memoria EEPROM
void saveInEEPROM(const uint8_t *mac_addr){
  Serial.println("");
  memcpy(modInfo.macAddress, mac_addr, 6);
  strcpy(modInfo.moduleType, syncData.moduleType);
  modInfo.moduleChannel = syncData.moduleChannel;
  modInfo.conected = syncData.conected;
  if(strcmp(modInfo.moduleType, "AIRE") == 0) {
    aireConectado = 1;
    EEPROM.put(0, modInfo);
    if(EEPROM.commit()) Serial.println("Module Aire Saved in Memory!");
  } else if(strcmp(modInfo.moduleType, "LUCES") == 0){
    lucesConectado = 1;
    EEPROM.put(1, modInfo);
    if(EEPROM.commit()) Serial.println("Module Luces Saved in Memory!");
  } else Serial.println("MODULE NO SAVED!!");
}

// Revisa si hay módulos registrados en la memoria EEPROM para registrarlos en
espnw
void checkModuleEEPROM(){
  for (int i=0; i<2; i++){
    Serial.println("");
    EEPROM.get(i, modInfo);
    if(strcmp(modInfo.moduleType, "AIRE") == 0 || strcmp(modInfo.moduleType,
"LUCES") == 0){
      Serial.println("Adding Module...");
      registerPeer(modInfo.macAddress, modInfo.moduleType,
modInfo.moduleChannel);
      Serial.println(" from EEPROM");
      Serial.print("Module "); Serial.print(modInfo.moduleType); Serial.println("
conected.");
      if(strcmp(modInfo.moduleType, "AIRE") == 0) aireConectado = 1;
      if(strcmp(modInfo.moduleType, "LUCES") == 0) lucesConectado = 1;
    } else Serial.println("No module in memory");
  }
}

// Inicialización de configuraciones y definición de pines
void setup() {
  //Serial.begin(115200);
```

```
EEPROM.begin(36); // tamaño de la memoria para guardar los datos de los módulos
sincronizados
Serial.println("Ready for pairing...");

// Definición de los pines
pinMode(syncBUTTON, INPUT);
pinMode(clearBUTTON, INPUT);
pinMode(sensor, INPUT);
pinMode(syncAireLED, OUTPUT); digitalWrite(syncAireLED, HIGH);
pinMode(syncLucesLED, OUTPUT); digitalWrite(syncLucesLED, HIGH);

// Temporizador inicializado
init_Timer();

// Configura el modo wifi del ESP32 como STATION
WiFi.mode(WIFI_STA);

// Inicializa el protocolo ESP-NOW
if (esp_now_init() != ESP_OK) {
  Serial.println("Error initializing ESP-NOW");
  return;
} else {
  Serial.println("ESP-NOW Ready");
}
// Registra la llamada a los mensajes recibidos desde los módulos con espnow
esp_now_register_recv_cb(OnDataRecv);
// Registra la dirección mac abierta para que todos los espnow escuchen
registerPeer(broadcastAddressX, "NONE", 0);
// Revisa si hay módulos guardados y los registra
checkModuleEEPROM();

// Se definen el valor de los datos iniciales
syncData.messageType = SYNC;
strcpy(syncData.moduleType, "MASTER");
syncData.moduleChannel = selfChannel;
syncData.conected = false;
syncData.statusOnOff = 0;
}

void loop() {
  if(bandTime){ // Ejecución cada 1ms
    bandTime=0;
    // Cuando el botón de sincronización se presiona por 5 segundos
    // envía mensaje de sincronización a todos los espnow cerca
    if(!digitalRead(syncBUTTON) && !(lucesConectado && aireConectado)){
```

```

    Serial.println("Sending request");
    countSync++;
    if(countSync >= 2500){ // 2500 -> 5 sec
        syncData.messageType = SYNC;
        esp_err_t result = esp_now_send(broadcastAddressX, (uint8_t *) &syncData,
sizeof(syncData));
    }
} else countSync = 0;

// Cuando el botón de borrado se presiona por 5 segundos se borran y
desasocian los módulos registrados y guardados
if(!digitalRead(clearBUTTON) && (aireConectado || lucesConectado)){
    countClear++;
    if(countClear >= 2500){ // 2500 -> 5s
        for (int i = 0 ; i < EEPROM.length() ; i++) EEPROM.write(i, 0);
        EEPROM.commit();
        esp_now_fetch_peer(1, &peerToClean);
        esp_now_del_peer(peerToClean.peer_addr);
        aireConectado = false;
        lucesConectado = false;
        Serial.println("Module(s) Disconnected.");
    }
} else countClear = 0;

// Si no está presionado el botón de sincronización y alguno de los módulos
está conectado, lee el sensor y manda los datos cada 330 ms
if(digitalRead(syncBUTTON) && (aireConectado || lucesConectado)){
    countSense2++;
    if(countSense2==330){
        countSense2 = 0;
        OnOff = digitalRead(sensor);
        if(OnOff){
            currentStatus = 1; //Sistema habilitado
            countSense = 0;
        } else {
            countSense++;
            Serial.println(countSense);
            if(countSense >= timeWaitSec){ //
                currentStatus = 0; //Sistema apagado
                countSense = 0;
            }
        }
    }
}
myData.messageType = DATA;
strcpy(myData.moduleType, "MASTER");
myData.conected = true;

```

```

        myData.statusOnOff = currentStatus;

        Serial.print(myData.messageType);
        Serial.print(myData.moduleType);
        Serial.print(myData.conected);
        Serial.print(myData.statusOnOff);
        Serial.println(currentStatus);

        esp_now_send(NULL, (uint8_t *) &myData, sizeof(myData));
    }
}
// Indica si los módulos están conectados
digitalWrite(syncAireLED, aireConectado); // Verde
digitalWrite(syncLucesLED, lucesConectado); // Amarillo
}
}

```

Anexo C. Código de programación para el módulo de aire y luces

```

/*
Modulo Aire/Luces
*/

// Librerías incluidas
#include <ESP8266WiFi.h>
#include <espnow.h>
#include <EEPROM.h>

#define syncBUTTON 14 // Pin del botón de sincronización
#define clearBUTTON 16 // Pin del botón de borrado
#define syncAireLED 5 // Pin del led Verde de conexión
#define signalOnOff 12 // Pin de salida de la señal de apagado

char modType[7] = "LUCES"; // Se debe de cambiar según el tipo de módulo que se
vaya a programar "Aire" / "LUCES"
int selfChannel = 1;

int checkStatus = 13; // Estado actual del Aire/Luces

unsigned int countSync = 0; // Contador para sincronizar los módulos (5 sec)
unsigned int countClear = 0; // Contador para borrar los módulos (5 sec)
unsigned int countSignal = 0; // Contador de la duración de la señal de
apagado/encendido enviada

bool aireStatus = 1; // Estado inicial al conectar el módulo aire

```

```
bool lucesStatus = 1; // Estado inicial al conectar el módulo luces
bool syncButtonPressed = 0; // Indica si el botón de sincronización está
presionado
bool masterConectado = false; // Indica si el módulo de sensado está conectado
uint8_t *MACpar; // Almacena la dirección MAC del módulo sensor para su borrado

// Tipo de variable numerada: {0, 1,}
enum MessageType {SYNC, DATA,};
MessageType messageType;

// Estructura de datos para la sincronización
typedef struct struct_sync {
    uint8_t messageType; // SYNC, DATA
    char moduleType[6]; //AIRE, LUCES, MASTER
    uint8_t moduleChannel; // 1
    bool conected; //0:Disconnected, 1:Connected
    uint8_t statusOnOff; //0:OFF-DISABLE, 1:ON-ENABLE
} struct_sync;

// Estructura de datos para envío de datos
typedef struct struct_data {
    uint8_t messageType; // SYNC, DATA
    char moduleType[7]; //AIRE, LUCES, MASTER
    bool conected; //0:Disconnected, 1:Connected
    uint8_t statusOnOff; //0:OFF-DISABLE, 1:ON-ENABLE
} struct_data;

// Estructura de datos para guardar el tipo de módulo
struct eeprom_moduloInfo {
    uint8_t macAddress[7]; // SYNC, DATA
    char moduleType[7]; //AIRE, LUCES, MASTER
    uint8_t moduleChannel; // 1
    bool conected; //0:Disconnected, 1:Connected
};

// Se crean objetos estructurados
struct_sync syncData;
struct_data myData;
eeprom_moduloInfo modInfo;

// Procesa los mensajes recibidos del módulo sensor
void OnDataRecv(uint8_t *mac_addr, uint8_t *incomingData, uint8_t len){
    uint8_t type = incomingData[0];
    switch (type) {
        case SYNC :
```

```

        memcpy(&syncData, incomingData, sizeof(syncData));
        if(!masterConectado && syncButtonPressed){
            if (!esp_now_is_peer_exist(mac_addr) && syncData.moduleType ==
"MASTER");{
                registerPeer(mac_addr);
                sendRequestSync(mac_addr);
                saveInEEPROM(mac_addr);
            }
        }
        break;
    case DATA:
        memcpy(&myData, incomingData, sizeof(myData));
        // Serial.print(myData.messageType);
        // Serial.print(myData.moduleType);
        // Serial.print(myData.conected);
        // Serial.println(myData.statusOnOff);
        if(myData.conected && masterConectado){
            if(myData.statusOnOff != digitalRead(checkStatus)){
                aireStatus = 0;
                lucesStatus = myData.statusOnOff;
            } else aireStatus = 1;
        }
        break;
    }
}

// Registra el módulos sincronizado o guardado en la memoria EEPROM
void registerPeer(uint8_t *mac_addr){
    if (esp_now_add_peer(mac_addr, ESP_NOW_ROLE_COMBO, selfChannel, NULL, 0) != 0){
        Serial.println("Fail to add peer");
        return;
    } else {
        Serial.println("Peer module MASTER added.");
    }
}

// Envía la solicitud de enlace cuando recibe la solicitud de enlace del módulo
sensor
void sendRequestSync(uint8_t *mac_addr){
    syncData.messageType = SYNC;
    strcpy(syncData.moduleType, modType);
    syncData.moduleChannel = selfChannel;
    syncData.conected = true;
    syncData.statusOnOff = 0; //Read status air
    if (esp_now_send(mac_addr, (uint8_t *) &syncData, sizeof(syncData)) == 0){

```

```
        Serial.println("Sending request confirmed");
        masterConectado = true;
    }
    else Serial.println("Sending request error");
}

// Guarda el módulo sensor sincronizado en la memoria EEPROM
void saveInEEPROM(uint8_t *mac_addr){
    memcpy(modInfo.macAddress, mac_addr, 6);
    strcpy(modInfo.moduleType, "MASTER");
    modInfo.moduleChannel = syncData.moduleChannel;
    modInfo.conected = true;
    EEPROM.put(0, modInfo);
    if(EEPROM.commit()) Serial.println("Module Master Saved in Memory!");
    else Serial.println("MODULE NO SAVED!!");
}

// Inicialización de configuraciones y definición de pines
void setup() {
    // Serial.begin(115200);
    EEPROM.begin(16);
    Serial.println("Ready for pairing...");

    // Definición de los pines
    pinMode(syncBUTTON, INPUT);
    pinMode(clearBUTTON, INPUT);
    pinMode(syncAireLED, OUTPUT); digitalWrite(syncAireLED, HIGH);
    pinMode(signalOnOff, OUTPUT); digitalWrite(signalOnOff, HIGH);
    if(strcmp(modType, "LUCES") == 0) {
        checkStatus = signalOnOff;
    } else pinMode(checkStatus, INPUT);

    // Configura el modo wifi del ESP32 como STATION
    WiFi.mode(WIFI_STA);

    // Inicializa el protocolo ESP-NOW
    if (esp_now_init() != 0) {
        Serial.println("Error initializing ESP-NOW");
        return;
    } else {
        Serial.println("ESP-NOW Ready");
    }
    // Define su propio rol
    esp_now_set_self_role(ESP_NOW_ROLE_COMBO);
    // Registra la llamada a los mensajes recibidos desde el módulo S. con espnow
```

```
    esp_now_register_recv_cb(OnDataRecv);
    // Evalua si hay alguna información del módulo sensor en la memoria EEPROM
    EEPROM.get(0, modInfo);
    // Si existe un módulo guardado en la EEPROM, lo agrega a la lista de módulos
    enlazados
    if(strcmp(modInfo.moduleType, "MASTER") == 0){
        Serial.println("Adding Module...");
        esp_now_add_peer(modInfo.macAddress, ESP_NOW_ROLE_COMBO,
modInfo.moduleChannel, NULL, 0);
        masterConectado = modInfo.conected;
        Serial.print("Module "); Serial.print(modInfo.moduleType); Serial.println("
conected.");
    } else Serial.println("No module in memory");
}

void loop() {
    static unsigned long lastEventTime = millis();
    static const unsigned long EVENT_INTERVAL_MS = 1;

    if ((millis() - lastEventTime) > EVENT_INTERVAL_MS) { // Ejecución cada "1ms"
        // Cuando el botón de sincronización se presiona por 5 segundos
        // cambia la forma de procesar los datos que recibe a través de espnow
        if(!digitalRead(syncBUTTON) && !masterConectado){
            countSync++;
            if(countSync >= 2500){ // 2500 -> 5sec
                syncButtonPressed = 1;
            }
        }else {
            countSync = 0;
            syncButtonPressed = 0;
        }

        // Cuando el botón de borrado se presiona por 5 segundos se borran y
        desasocian los módulos registrados y guardados
        if(!digitalRead(clearBUTTON) && masterConectado){
            countClear++;
            if(countClear >= 2500){
                for (int i = 0 ; i < EEPROM.length() ; i++) {
                    EEPROM.write(i, 0);
                }
                EEPROM.commit();
                MACpar = esp_now_fetch_peer(1);
                esp_now_del_peer(MACpar);
                masterConectado = false;
            }
        }
    }
}
```

```
        Serial.println("Module Disconnected.");
    }
}else {
    countClear = 0;
}

// Si el módulo programado es del AIRE y el módulo S. está conectado
// y recibe una señal de apagado o encendido, cambia el estado del aire
if(!strcmp(modType, "AIRE") && !aireStatus && masterConectado){
    countSignal++;
    Serial.println(countSignal);
    if (countSignal <= 400){
        digitalWrite(signalOnOff, LOW);
    }
    if(countSignal > 400 && countSignal < 8000){
        digitalWrite(signalOnOff, HIGH);
    }
    if(countSignal >= 8000) countSignal = 0;
} // Si el módulo programado es el de LUCES, cambia el estado de las luces
else if(!strcmp(modType, "LUCES") && !aireStatus && masterConectado){
    digitalWrite(signalOnOff, lucesStatus);
}
else countSignal = 0;

lastEventTime = millis();

digitalWrite(syncAireLED, masterConectado);
}
}
```