



UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA
CARRERA DE MECATRÓNICA

DISEÑO E IMPLEMENTACIÓN DEL SISTEMA DE MONITOREO DE
ALARMA CONTRA INCENDIOS PARA LA UNIVERSIDAD
POLITÉCNICA SALESIANA SEDE CUENCA

Trabajo de titulación previo a la obtención
del título de Ingeniera en Mecatrónica/
Ingeniero en Mecatrónica

AUTORES: KULLKI KINARA QUINDI PICHISACA
SANTIAGO DAVID RUIZ FLORES

TUTOR: ING. DIEGO PAUL CHACÓN TROYA, MDHD.

Cuenca – Ecuador

2023

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, Kullki Kinara Quindi Pichisaca con documento de identificación N° 0302655493 y Santiago David Ruiz Flores con documento de identificación N° 1105802274; manifestamos que:

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

Cuenca, 31 de agosto del 2023

Atentamente,



Kullki Kinara Quindi Pichisaca
0302655493



Santiago David Ruiz Flores
1105802274

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

Nosotros, Kullki Kinara Quindi Pichisaca con documento de identificación N° 0302655493 y Santiago David Ruiz Flores con documento de identificación N° 1105802274, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del Proyecto Técnico: "Diseño e implementación del sistema de monitoreo de alarma contra incendios para la Universidad Politécnica Salesiana sede Cuenca", el cual ha sido desarrollado para optar por el título de: Ingeniera en Mecatrónica / Ingeniero en Mecatrónica, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

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

Cuenca, 31 de agosto del 2023

Atentamente,



Kullki Kinara Quindi Pichisaca
0302655493



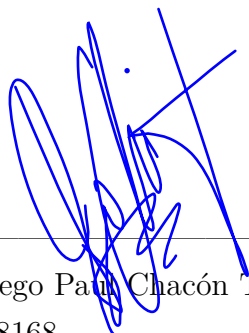
Santiago David Ruiz Flores
1105802274

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Diego Paul Chacón Troya con documento de identificación N° 1900268168, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DISEÑO E IMPLEMENTACIÓN DEL SISTEMA DE MONITOREO DE ALARMA CONTRA INCENDIOS PARA LA UNIVERSIDAD POLITÉCNICA SALESIANA SEDE CUENCA, realizado por Kullki Kinara Quindi Pichisaca con documento de identificación N° 0302655493 y Santiago David Ruiz Flores con documento de identificación N° 1105802274, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Cuenca, 31 de agosto del 2023

Atentamente,

A handwritten signature in blue ink, consisting of several loops and a long horizontal stroke, positioned above a horizontal line.

Ing. Diego Paul Chacón Troya, Mdhd.
1900268168

Dedicatoria

Kullki Kinara

A mis padres Andrés y Magdalena quienes han sido un pilar fundamental durante todo el transcurso de los años de estudio de mi carrera, apoyándome incondicionalmente en cada momento; sin su apoyo no lo habría logrado.

A mi hermana Cristina, quien siempre esta presente y ha sido un ejemplo a seguir.

De igual manera, dedico este trabajo a mis abuelos, Andrés, Rosa, José y Ana; quienes con sus palabras e historias se convirtieron en una gran inspiración para salir adelante.

Santiago David

Quiero dedicar este trabajo especialmente a mi madre Rosalba y a mi padre German por su paciencia y apoyo que me ha brindado a lo largo de estos años, a mis hermanos Anthony y Samantha que me han acompañado en diversas situaciones y me han ayudado a seguir adelante.

Agradecimientos

Kullki Kinara

Agradezco a mi padre Andrés quien me ha apoyado en este proceso y ha confiado en mí en todo momento a pesar de todas las dificultades. A mi madre que siempre me demostró que nunca es tarde para mejorar. Su compromiso y dedicación para ayudarme a alcanzar mis metas ha sido una verdadera bendición en mi vida y una fuente constante de inspiración.

A mi tutor de trabajo de titulación, Ing. Diego Chacón, por su apoyo constante durante el desarrollo del proyecto. Gracias a su orientación, he logrado alcanzar los objetivos académicos. Por lo cual le estoy eternamente agradecida.

Santiago David

Quiero agradecer a mi madre Rosalba por estar siempre presente para mí, motivándome y ser mi guía para poder seguir adelante. A mi hermana Samantha por todo el apoyo y paciencia que me ha tenido.

A mi hermano Anthony y mi primo Cristian por acompañarme en las incontables noches en vela de estudio y ocio, quiero agradecer a mis familiares que ha estado conmigo en las buenas y en las malas.

A mis amigos que me han sabido aconsejar y brindar su apoyo a lo largo de mi vida como estudiante. Gracias por compartir conmigo los buenos momentos y estar junto a mí en los malos.

Al Ing. Diego Chacón por la paciencia que nos ha tenido y su constancia y apoyo. Gracias por brindarnos de su tiempo dentro de su apretada agenda y su conocimiento que nos ayudó a terminar este trabajo.

Este documento fue realizado enteramente en L^AT_EX

Índice

Certificado de responsabilidad y autoría del trabajo de titulación	I
Certificado de cesión de derechos de autor del trabajo de titulación a la Universidad Politécnica Salesiana	II
Certificado de dirección del trabajo de titulación	III
Dedicatoria	IV
Agradecimientos	V
Resumen	XIII
Abstract	XIV
1. Introducción	1
2. Problema	2
2.1. Antecedentes	2
2.2. Descripción del problema	2
2.3. Importancia y alcances	2
2.4. Delimitación	3
2.4.1. Espacial o geográfica	3
3. Objetivos	4
3.1. Objetivo General	4
3.2. Objetivos Específicos	4
4. Marco Teórico	4
4.1. Raspberry Pi	4
4.2. Dispositivos de inicio	5
4.3. Dispositivos de notificación	6
4.4. Dispositivos de control	6
4.5. Módulo de alarmas	6
4.6. Detector de humo	6
4.6.1. Tipos de detectores de humo	7

4.7. Python	7
4.8. Node-RED	8
4.9. Código Nacional de Alarma contra Incendios (NFPA)	8
5. Marco metodológico	10
5.1. Identificación de los dispositivos tecnológicos elementales del sistema de monitoreo de alarma contra incendios	10
5.1.1. Situación actual del sistema de alarma	10
5.1.2. Componentes del sistema	10
5.1.3. Raspberry Pi 4:	11
5.1.4. System Sensor 4W-B 4 hilos, detector de humo fotoeléctrico i3:	14
5.1.5. Dispositivos de inicio	17
5.1.6. Dispositivos de notificación	18
5.1.7. Cableado	19
5.2. Diseño del sistema de monitoreo de alarma contra incendios	21
5.2.1. Diagrama de flujo de funcionamiento	21
5.2.2. Requisitos y requerimientos	23
5.2.3. Componentes del sistema	24
5.2.4. Alimentación del sistema	24
5.2.5. Análisis de consumo de energía	25
5.2.6. Elección de la batería	27
5.2.7. Funcionamiento del sistema electrónico	29
5.2.8. Simulación del sistema electrónico	31
5.2.9. Programación de la simulación	32
5.2.10. Simulación de temperatura de la unidad de procesamiento	35
5.2.11. Diseño de la placa PCB	40
5.2.12. Elección del gabinete	43
5.3. Implementación de los dispositivos de inicio del sistema de monitoreo de alarma contra incendios	44
5.3.1. Programación del botón de emergencia físico y virtual	45
5.3.2. Programación del sensor	47
5.4. Comprobación del funcionamiento del sistema de monitoreo de alarma contra incendios	49
6. Resultados	58

7. Conclusiones	64
8. Recomendaciones	65
Referencias	69
ANEXOS	70
Anexo B: Matriz de Consistencia Lógica	70
Anexo A: Código implementado en Proteus	71

Lista de Tablas

1.	Características Raspberry	13
2.	Sensor 4W-B: Especificaciones eléctricas	14
3.	Sensor 4W-B: Especificaciones físicas	15
4.	Características de la sirena estroboscópica SH-816S-SQ	20
5.	Tabla de componentes	24
6.	Alimentación del sistema de alarma	25
7.	Consumo en potencia por un día	26
8.	Características de la batería Steren	28
9.	Características del ventilador para Raspberry Pi 4 r	38
10.	Características del gabinete eléctrico Beacoup	44
11.	Matriz de consistencia	75

Lista de Figuras

1.	Delimitación espacial	3
2.	Placa Raspberry Pi	5
3.	Lenguaje de programación Python	8
4.	Conexión de dispositivos de inicio y notificación con IoT desde Node-red . .	9
5.	Esquema de minicomputador Raspberry Pi “Modelo B, rev 1”	11
6.	Diagrama del sensor 4W-B	15
7.	Esquema de conexión	16
8.	Estación manual serie G-12	18
9.	Sirena estroboscópica SH-816S-SQ	19
10.	Circuito con detectores de humo de cuatro hilos y relé de supervisión individual	20
11.	Componentes básicos de un sistema de alarma contra incendios.	21
12.	Diagrama de bloque del sistema de monitoreo de alarma contra incendio . .	22
13.	Fuente de alimentación secundaria	28
14.	Diagrama de activación automático	29
15.	Diagrama de activación manual	30
16.	Simulación del sistema electrónico	31
17.	Calor interno generado por la unidad de procesamiento	35
18.	Simulación del comportamiento térmico sin disipación	36
19.	Asignación de un ventilador para la simulación	37
20.	Ventilador para Raspberry Pi 4	38
21.	Simulación del comportamiento térmico con un disipación	39
22.	Circuito esquemático utilizado para la PCB	41
23.	PCB utilizada	42
24.	Gabinete liviano Beacoup	43
25.	Interfaz gráfica	45
26.	Nodos de programación de la estación manual y activación desde interfaz . .	45
27.	Nodos de programación para temperatura	46
28.	Dirección IP	47
29.	Nodos para la visualización de datos del sensor	49
30.	Configuración de dirección IP	50
31.	Activación del sistema de alarma por estación manual/botón de interfaz . . .	51
32.	Activación automática del módulo mediante los sensores	52
33.	Grupo de notificación de alerta en Telegram	53

34.	Mensaje recibido a Telegram	53
35.	Nodos para la conexión Node-RED y Telegram	56
36.	Nodos empleados para la notificación en caso de desconexión	57
37.	Interfaz usuario-máquina	58
38.	Notificación emitida en la interfaz	59
39.	Desactivación de alarma	60
40.	Notificación de desconexión de dispositivos	60
41.	Ajuste y desajuste de bornes.	61
42.	Comportamiento térmico	62
43.	Notificación de alerta por Telegram	63
44.	Notificación automática de desconexión por Telegram	63

Resumen

El presente trabajo de titulación fue realizado en función de la necesidad de contar con un sistema de monitoreo de alarma contra incendios, debido a que la Universidad no dispone de uno, esto surge a raíz de la falta de mantenimiento y sobrecarga del sistema de alarma actual, provocando que este falle dejando a la Universidad, inerte ante un evento de riesgo de incendio. El diseño del sistema de monitoreo de alarma se realiza mediante una investigación exhaustiva para la aplicación tecnológica de dispositivos de inicio y notificación; fundamentada en la norma de seguridad emitido en el Código nacional de alarmas de incendio y señalización por la NFPA "National Fire Prevention Association", con disposiciones para la detección de incendios, señalización y comunicación ante emergencias.

El sistema de monitoreo fue desarrollado con una Raspberry Pi como placa de procesamiento de datos, debido a la variedad de aplicaciones que permite realizar y el bajo consumo energético. La placa se programa con el lenguaje de programación Python para procesar los datos recogidos de los dispositivos de inicio como los sensores de humo. Posteriormente, mediante la herramienta de programación de Node-RED, se logra recopilar e intercambiar la información de los dispositivos de inicio y notificación en tiempo real por medio de una interfaz de usuario, en donde se podrá visualizar e interactuar directamente con el sistema de monitoreo de alarma contra incendios.

Los resultados obtenidos se desarrollan en cuatro etapas, investigación, diseño, implementación y pruebas, logrando que el sistema de monitoreo se pueda controlar por medio de un entorno virtual en tiempo real, de manera automática y por accionamiento manual. Esto posibilita la detección temprana de incendios y la transmisión de señales de alerta, tanto visual como audible dentro de la propia interfaz; adicionalmente, a través de avisos de alerta por medio de la aplicación de mensajería móvil y de escritorio, Telegram.

Palabras clave: Alarma, dispositivos de inicio, dispositivos de notificación, NFPA, Node-RED, sistema de monitoreo, Raspberry.

Abstract

The present degree work was carried out based on the necessity to have a fire alarm monitoring system, because the university does not have one, this arises as a result of the lack of maintenance and overload of the current alarm system, causing it to fail, leaving the university defenseless against a fire risk event. The design of the alarm monitoring system is done through extensive research for the technological application of initiating and notification devices; based on the safety standard issued in the National Code of fire alarms and signaling by the NFPA "National Fire Prevention Association", with provisions for fire detection, signaling and communication in emergencies.

The monitoring system was developed with a Raspberry Pi as data processing board, due to the variety of applications that it allows to carry out and the low power usage. The board is programmed with the Python programming language to process the data collected from the initiating devices such as smoke sensors. Then, through the Node-RED programming tool, collect and exchange the information of the initiating and notification devices in real time through a user interface, where it can be viewed and interact directly with the fire alarm monitoring system.

The results achieved are developed in four stages, research, design, implementation and testing, ensuring that the monitoring system can be controlled through a virtual environment in real time, automatically and manually. This enables early detection of fires and the transmission of alert signals, both visual and audible on the interface; additionally, by alert notifications through the mobile and desktop messaging application, Telegram.

Keywords: Alarm, initiating devices, notification devices, NFPA, Node-RED, monitoring system, Raspberry.

1. Introducción

La implementación del sistema de alarma “lleva más de un siglo desarrollándose, tiene el principal propósito de detectar a tiempo condiciones de peligro que se susciten, con notificación inmediata ante posibles y potenciales sucesos o situaciones de incendio.” (Chugá Meneses, 2019). “Los sistemas de alarmas contra incendio han evolucionado juntamente con el campo de la electrónica en lo referente a control y comunicación logrando sistemas más eficientes y mejorando la seguridad.” (Velasco Núñez, 2015)

Los sistemas de alarmas contra incendios forman parte esencial dentro de edificaciones debido a que permiten una detección temprana de diversos desastres y de ese modo garantizar la seguridad de los ocupantes, salvaguardar la infraestructura y reducir las pérdidas económicas que esto pueda ocasionar. En este contexto se define al sistema de alarma contra incendios también denominado FAS, como:

“un sistema combinado que detecta incendios no deseados mediante el monitoreo de cambios ambientales asociados con la combustión, como humo o calor. El FAS consiste en un panel de control de alarma contra incendios (FACP), detectores, una caja de alarma manual contra incendios, un dispositivo de alarma de flujo de agua, una fuente de alimentación, circuitos y otros componentes.” (Jee, Lee, Kim, Lee, y Kim, 2014)

En este trabajo se plantea la implementación de un sistema de monitoreo de alarma contra incendio dentro de la Universidad Politécnica Salesiana sede Cuenca, debido a que, al ser un establecimiento de educación superior, debe solventar la seguridad del personal y protección de la infraestructura. Para el desarrollo se consideró la normativa de diseño e implementación del código 72 de la NFPA: "National Fire Prevention Association". Siendo esta norma la que "proporciona las últimas disposiciones de seguridad para satisfacer las cambiantes demandas de detección de incendios, señalización y comunicaciones de emergencia de la sociedad." (“NFPA JLA”, 2023)

La implementación del sistema de monitoreo de alarma contra incendios debe cumplir con requerimientos propuestos por el personal administrativo y técnico de la institución, usando nuevas técnicas y herramientas que permitan al personal de monitoreo responder de manera inmediata ante cualquier situación de riesgo. El sistema de alarma contará con diferentes fases que comprende desde la adquisición de datos que estará dado por los sensores, el procesamiento de datos realizado por una Raspberry Pi y la visualización mediante una interfaz que puede ser ejecutada desde cualquier ordenador; la misma que permitirá interactuar con los diferentes dispositivos de inicio y notificación.

2. Problema

2.1. Antecedentes

La Universidad Politécnica Salesiana (UPS) sede Cuenca, cuenta con un sistema de alarma contra incendios que actualmente se encuentra fuera de servicio debido a la falta de mantenimiento de la tarjeta electrónica usada como base de todo el sistema. Ante esta problemática la Universidad se ha visto en la necesidad de renovarlo. Por otro lado, se debe tener en cuenta que dentro de la institución existen lugares que no cuentan con la alarma contra incendios por lo que también se debe ampliar la zona de alcance de este.

2.2. Descripción del problema

El sistema de alarmas con el que cuenta la Universidad ha presentado varias fallas y problemas de conexión, debido a la falta de mantenimiento, convirtiéndose en un sistema obsoleto. Como consecuencia, se pone en peligro la seguridad e integridad tanto del personal administrativo, comunidad educativa como también el de los bienes y áreas que conforman la institución. Siendo indispensable contar con un sistema de alarma con el fin de solventar y cumplir con la responsabilidad que como institución educativa debe proveer en caso de emergencia, se toma como base las destrezas adquiridas juntamente con las nuevas tecnologías y se plantea la implementación de un sistema de alarmas que integra una tarjeta electrónica moderna para mejorar la interfaz de monitoreo entre el usuario y el sistema.

2.3. Importancia y alcances

En la Universidad Politécnica Salesiana Sede Cuenca a fin de solventar la integridad y seguridad de personal y de los bienes de la institución se desarrollará y renovará el sistema de monitoreo de alarma contra incendios actual de manera que se optimice el funcionamiento y expansión de zona de alcance del sistema para el control y monitoreo ante posibles riesgos. Para ello se implementará un sistema alarma dentro del sector universitario; dicho sistema será programado en una placa electrónica “Raspberry Pi”, con el lenguaje de programación Python y la herramienta de programación Node-Red. El alcance del diseño e implementación del sistema de monitoreo de alarma contra incendios se pretende desarrollar e implementar en un periodo total de cuatrocientas horas que durara aproximadamente seis meses.

2.4. Delimitación

El problema de estudio se delimitará en las siguientes dimensiones:

2.4.1. Espacial o geográfica

El proyecto está enfocado a desarrollarse en la Universidad Politécnica Salesiana, específicamente la sede Cuenca la cual se encuentra al noroeste de la ciudad de Cuenca provincia del Azuay, Ecuador.

Figura 1

Delimitación espacial



Nota: La imagen muestra la ubicación de la Universidad Politécnica Salesiana sede Cuenca en la ciudad de Cuenca provincia del Azuay, Ecuador. Obtenido de (Google, 2023).

3. Objetivos

3.1. Objetivo General

- Diseñar e implementar el sistema de monitoreo de alarma contra incendios para la Universidad Politécnica Salesiana sede Cuenca

3.2. Objetivos Específicos

- Identificar los dispositivos tecnológicos elementales para la implementación del sistema de monitoreo de alarma contra incendios en la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria.
- Diseñar el sistema de monitoreo de alarma contra incendios en la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria, de tal forma que sea fácil su manipulación.
- Implementar los dispositivos de iniciación al sistema de monitoreo de alarma contra incendios de la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria, eludiendo eventos que provoquen un riesgo de incendio.
- Realizar pruebas del sistema de monitoreo de alarma contra incendios en la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria

4. Marco Teórico

4.1. Raspberry Pi

El Raspberry Pi se trata de un dispositivo que se puede asimilar como una computadora de tamaño reducido, que permite realizar actividades que se realizan comúnmente en una computadora normal, también se lo describe como “un ordenador low_cost capaz de realizar las mismas funciones que un ordenador estándar de sobremesa, desarrollado con el objetivo de hacer la informática accesible y asequible para todos.” (*¿Qué es Raspberry Pi y para qué sirve?*, 2021). En la figura 2 mostrada a continuación observamos la estructura de la placa Raspberry Pi:

Figura 2

Placa Raspberry Pi.



Nota: En la figura se puede observar la estructura de una Placa Raspberry Pi. Obtenido de (Upton, 2019).

4.2. Dispositivos de inicio

Los dispositivos de inicio “son aquellos que permiten alertar sobre alguna emergencia o comportamiento anormal del sistema, estos pueden ser accionados de manera manual o automática la cual envía una señal hacia el panel de control.” (firesystem.ec, 2021). Los dispositivos de inicio pueden ser:

- Detector de humo:
- Estación manual:
- Detector de calor

4.3. Dispositivos de notificación

Los dispositivos de notificación “son muy necesarios para detectar oportunamente el fuego temprano y luego enviar las señales de alarma antes de que el fuego se propague rápidamente. Los sensores de advertencia de incendios son componentes clave.” (He, Feng, Xu, Chen, y Yu, 2022).

“Los dispositivos acústicos fueron los primeros en usarse; la mayoría generan una presión acústica entre 45 y 120 decibeles a un metro. Por otro lado, la sirena de alarma luminosa es para ahuyentar, pues la luz suele ser de tonos llamativos como naranjas o rojos, además que puede girarse para que se proyecte en diferentes direcciones.”(Mendieta, 2022).

4.4. Dispositivos de control

Los dispositivos de control son los que “permiten mantener o lograr un valor deseado mediante el uso o medición de distintas variables de acuerdo con lo que se requiera implementar. Es un tipo de sistema de regulación activa diseñado para mejorar u optimizar el rendimiento específico o los indicadores para sistemas.” (Song, 2022).

4.5. Módulo de alarmas

El monitoreo de alarmas es “el cerebro o administrador de un sistema de alarma, el cual cumplen la función de recibir las señales de los sensores periféricos, para proporcionar una salida de alerta (que puede ser o no audible) y una comunicación a los sistemas de vigilancia, además puede activar salida a otros dispositivos auxiliares.” (Juarez, 2023)

4.6. Detector de humo

Según (Turmo, 2022) menciona que los detectores de humo también se les denomina detectores de productos de combustión. Explica que se trata de:

Detectores agrupados en zonas (planta de un edificio, sección, sector, etc.) y conectados a la central de control y señalización por unos bucles (línea o circuito eléctrico que une los detectores a la central). Los detectores son unos dispositivos que captan un determinado fenómeno (en nuestro caso humo) y cuando el valor de ese fenómeno sobrepasa un umbral prefijado se genera una señal de alarma que es transmitida a la central de control y señalización de una forma muy simple, generalmente como cambio de consumo o tensión en la línea de detección. (Turmo, 2022)

4.6.1. Tipos de detectores de humo

Existen diversos tipos de detectores de humo de entre los cuales los más destacados son:

- Detectores de humo iónicos
- Detectores ópticos
- Detector térmico
- Detector de radiación

4.7. Python

Se trata de uno de los lenguajes de programación, considerado lenguaje de alto nivel, con mayores aplicaciones en el desarrollo web, de aplicaciones y en manejo de la ciencia de los datos. Se puede usar libremente debido a que es de código abierto, haciendo referencia a que cualquier usuario puede hacer uso, modificación o creación de programas con este lenguaje. Una explicación más concreta se la puede encontrar en su página oficial Python, donde define como: un lenguaje de código abierto, portátil y de propósito general de alto nivel. A continuación, en la figura 3 se muestra una ventana con del software Python.

Cuenta con un intérprete que proporciona un entorno interactivo, un sistema de tipo dinámico, así como la gestión automática de la memoria. Al estar orientado a objetos por naturaleza, es ampliamente utilizado y proporciona una biblioteca grande y completa para aplicaciones del mundo real. Los programas Python se pueden incrustar fácilmente en programas escritos en lenguajes de programación como Julia, C, C ++, R, etc. (Nagar, 2018).

Figura 3

Lenguaje de programación Python.



Nota: Se visualiza el logo del lenguaje de programación Python. Obtenido de: (Python, 2021).

4.8. Node-RED

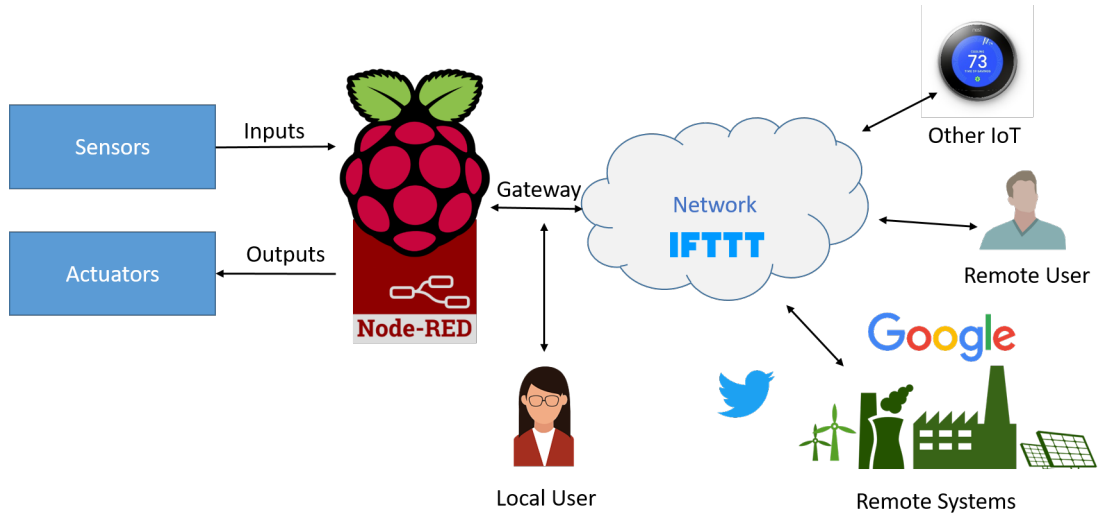
“Node-RED es una herramienta de programación para conectar dispositivos de hardware, API y servicios en línea de formas nuevas e interesantes.” de acuerdo con Node-RED Se trata de un programa o una herramienta que basada en Java Script, en general se trata de una “herramienta flexible y potente que se utiliza para crear prototipos. Este sistema permite la creación rápida de aplicaciones, especialmente aplicaciones que se activan en un evento como las aplicaciones IoT.” (Lekić y Gardašević, 2018). En la figura 4 se muestra el esquema de conexión de señales de dispositivos inicio con IoT mediante Node-Red.

4.9. Código Nacional de Alarma contra Incendios (NFPA)

La Asociación Nacional de Protección contra el Fuego o NFPA por sus siglas en ingles es una organización que se encargada de crear las normas y requisitos mínimos para la prevención de incendio, capacitación, instalación y uso de medios de protección. Para el proyecto se hará uso del código 72 de la NFPA Protective Signal System que es la norma usada para las alarmas contra incendios. (“NFPA JLA”, 2023).

Figura 4

Conexión de dispositivos de inicio y notificación con IoT desde Node-red.



Nota: Se puede observar un esquema de conexión de señales de sensores y actuadores con IoT mediante el software de programación Node-RED. Obtenido de: (Hale, 2017).

5. Marco metodológico

5.1. Identificación de los dispositivos tecnológicos elementales del sistema de monitoreo de alarma contra incendios

5.1.1. Situación actual del sistema de alarma

La unidad de control para la alarma contra incendios que dispone la Universidad está construida una tarjeta electrónica Arduino como unidad de procesamiento, este módulo ha dejado de funcionar por varios problemas de hardware y software. Principalmente no responde a señales de detección de humo y cambios de temperatura asociados con riesgos de incendio, tampoco emite ningún sonido de alarma, ni activa las notificaciones para lo cual fue desarrollada; siendo esas características cruciales para alertar a sobre el peligro de incendio y tomar medidas preventivas.

A pesar de que se ha intentado solucionar el problema verificando las conexiones, e incluso cargando nuevamente el código de programa en el Arduino, no se ha tenido éxito. En ese aspecto, el sistema de alarma contra incendios sigue sin funcionar y no cumple con su propósito principal de brindar protección contra incendios. Siendo necesario reconstruir una nueva unidad de control para abordar rápidamente el problema y restaurar el funcionamiento del sistema de alarma contra incendios para garantizar la seguridad de la comunidad universitaria y los bienes materiales.

5.1.2. Componentes del sistema

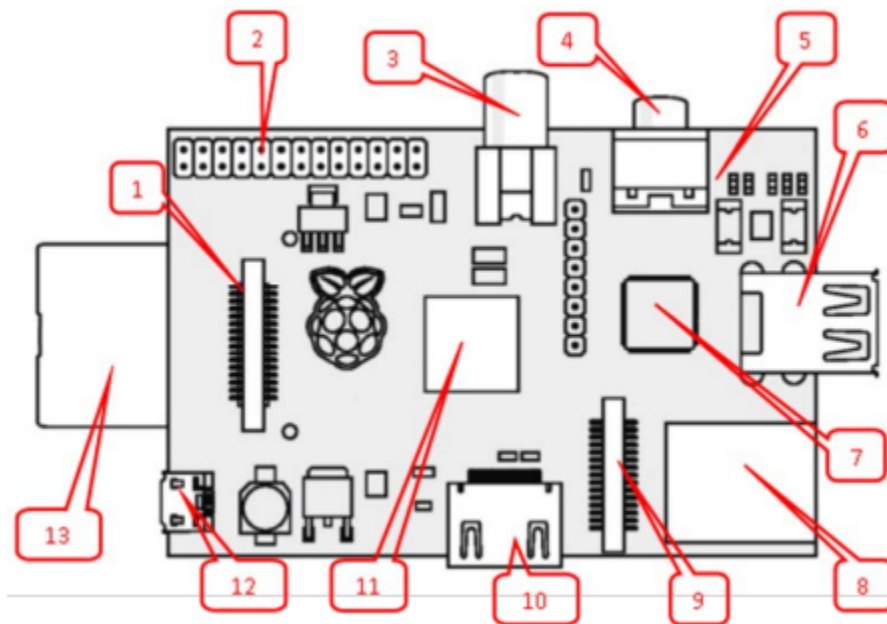
El diseño del sistema de monitoreo de alarma se lleva a cabo con una selección exhaustiva de los dispositivos y componentes adecuados, cumpliendo los códigos de la norma NFPA-72, que promueve estándares de seguridad y defensa sobre incendios; a su vez describe normas de diseño, construcción e implementación que son necesarias para el desarrollo del sistema de alarma. A raíz de eso, se considera las características de los componentes que serán utilizados, los cuales se describe a continuación en lista con todos los dispositivos estándar seleccionados que cumplen la normativa son aprobados por la central de monitoreo y departamento de Gestión de Riesgos de Universidad Politécnica Salesiana sede Cuenca.

5.1.3. Raspberry Pi 4:

La placa Raspberry Pi 4 Modelo B ofrece incrementos en la velocidad del procesador, el rendimiento multimedia, la memoria y la conectividad en comparación con la Raspberry Pi 3 Model B + de la generación anterior, al tiempo que conserva la compatibilidad con versiones anteriores y un consumo de energía similar."(P. Raspberry, 2021). En la figura 5 se muestra algunos de los componentes principales de la Raspberry.

Figura 5

Esquema de minicomputador Raspberry Pi “Modelo B, rev 1”.



Nota: En la figura se puede observar el esquema de alguno de los componentes principales del Raspberry Pi. Obtenido de (Salcedo y Cendrós, 2016).

De acuerdo con (Salcedo y Cendrós, 2016) los diferentes componentes que se muestran en la figura 5 se definen como:

- 1(Conector DSI): Este es un conector de tipo Flex-Flex el cual nos sirve para conectar pantallas LCD que sean de estándar DSI
- 2(cabezal de expansión): De acuerdo con (Salcedo y Cendrós, 2016) el Raspberry en las versiones “A+” y “B+” “comprende un grupo de 26 pines agrupados en un conector DIL

de 13 pines por dos filas” estos pines pueden ser configurados como entradas, salidas, PWM y Latch.

- 3: entrada de vídeo
- 4: puerto de audio
- 5: Leds de estado
- 6: Puertos USB 2.0
- 7: controlador ethernet
- 8 puerto ethernet 10/100
- 9 conector csi-2: tipo de conector Flat-Flex que permite la conexión de una cámara.
- 10 conector HDMI
- 11 system on a chip: es un encapsulado donde se encuentra la GPU, CPU, DSP (procesador digital de señales) y la memoria SDRAM.
- 12: puerto USB
- 13: lector de tarjetas MMC/SD

Características de Raspberry:

En la tabla 1 se muestra algunas de las características técnicas de la placa Raspberry Pi 4B.

Tabla 1*Características de la Raspberry Pi 4 modelo B.*

Característica:	Especificaciones:
Procesador :	Broadcom BCM2711 SoC de 64bits Cortex-A72 de 4 núcleos (ARM v8) a 1.5 GHz
Memoria	LPDDR4 de 1GB, 2GB, 4GB u 8GB con ECC integrado.
Conectividad	LAN inalámbrica IEEE 802.11b/g/nac de 2.4 y 5.0 GHz. Bluetooth 5.0, BLE. Gigabit Ethernet 2 puertos USB 3.0. 2 puertos USB 2.0.
GPIO	cabecera estándar de 40 pines
Sonido de video	2 puertos micro HDMI (hasta 4Kp60) Puerto de pantalla 2-lane MIPI DSI Puerto de cámara 2-lane MIPI CSI Puerto de audio y video 4-pole
Multimedia	H.265 (decodificación 4Kp60) H.264 (decodificación 1080p60, codificación 1080p30); OpenGL ES, gráficos 3.0
Tarjeta SD	Ranura para tarjeta microSD
Alimentación	5 VCC a través del conector USB-C (mín: 3A) 5VDC a través de la entrada GPIO (mín: 3A) (PoE) habilitado (requiere PoE HAT separado)
Temperatura de operación	0-80°C

Nota: En la tabla podremos observar características específicas de la placa Raspberry Pi 4 modelo B del manual presentado por Raspberry. Obtenido de (P. Raspberry, 2021).

5.1.4. System Sensor 4W-B 4 hilos, detector de humo fotoeléctrico i3:

Los modelos 4W-B y 4WT-B son detectores fotoeléctricos de humo de cuatro cables. Este incluye una cámara de detección óptica. “El microprocesador le permite al detector ajustar automáticamente su sensibilidad a la configuración de fábrica cuando se vuelve más sensible debido a la acumulación de contaminantes en su cámara.” (Systemsensor, 2023). En la tabla 2 se observa las características eléctricas del sensor de 4 hilos.

Tabla 2

Especificaciones eléctricas del sensor 4W-B.

Especificaciones Eléctricas 1	Sensor 4 hilos
Voltaje del Sistema:	12/24V no polarizados
Voltaje Mín.:	8,5 V
Voltaje Máx.:	35 V
Corriente Máx. en Normal:	50 muA promedio
Corriente de Alarma Máx:	20 mA sistemas de 12V
Potencia de Contacto de Alarma:	0,5 Amp a 30 VCA/CD
Tiempo de Restablecimiento de Alarma:	0,3 seg.
Alarma con Enclavamiento: Restablecimiento por interrupción de energía momentánea	
Tiempo de Arranque Inicial Máx.:	15 seg.
Tiempo de Arranque de Verificación de Alarma	15 seg.

Nota: En la tabla se presentan especificaciones eléctricas del sensor 4W-B necesarias para la implementación en el sistema de alarma contra incendios. Obtenido de (Systemsensor, 2023).

Además, “este modelo de sensor incluye un detector térmico con restablecimiento incorporado de temperatura fija (57,2°C) y son aptos para indicar una condición de congelamiento si la temperatura es inferior a 5°C.” (Systemsensor, 2023). Este detalle cumple con la sección 17.6.2.3 de la norma NFPA 72, referente a temperatura de funcionamiento. En la tabla 3 se describe las especificaciones físicas del sensor.

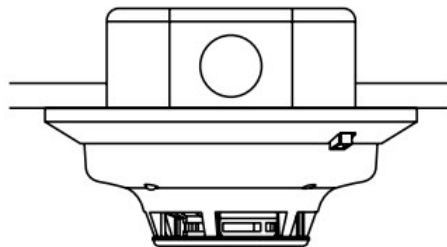
Tabla 3*Características físicas y rangos de trabajo del sensor 4W-B.*

Especificaciones Físicas	Descripción
Sensor de Temperatura	57,2 °C
Problema de Congelamiento	5°C
Rango de Temperatura Óptima	0-49°C
Rango de Humedad Operativa	0 a 95 HR sin condensación
Rango de Temperatura de Almacenamiento	-20 a 70 °C
Diámetro (base incluida)	13,4 cm
Altura (base incluida)	5,0 cm
Peso	178,6 g

Nota: En la tabla se expone distintas especificaciones físicas y rangos de trabajo correspondientes al sensor de 4W-B. Obtenido de (Systemsensor, 2023).

Montaje y cableado

Cada detector de la serie i3 incluye una base de montaje con diseño Plug-in que permite el enrutamiento de los cables en la base, este sensor puede ser montado en superficies de cajas, paneles o directamente al cielo raso mediante sujetadores para tabiques. En la figura 6 se distingue el diagrama correspondiente al sensor4W-B.

Figura 6*Diagrama del sensor 4W-B.*

Nota: En la figura se muestran un diagrama del sensor. Obtenido de (Systemsensor, 2023).

Condiciones de cableado:

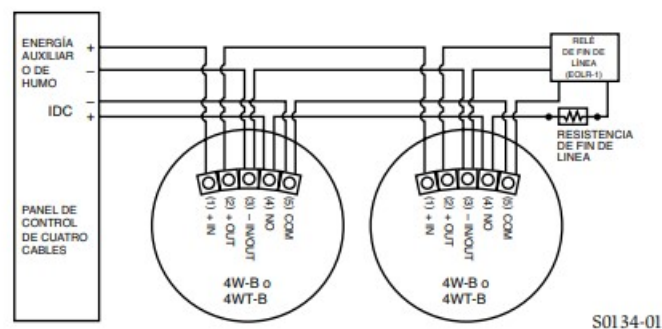
Según el manual del fabricante, recomienda considerar varios aspectos para el cableado, los cuales son detallados a continuación.

- El cableado debe realizarse de acuerdo con el código NFPA, bajo el código eléctrico nacional.
- Los colores de los cables deben mantener la normativa bajo el código de colores para cables eléctricos.
- Los terminales roscados en la base de montaje admiten un diámetro de cable de 1,6-0,6 mm.
- El cableado debe instalarse en una tubería eléctrica a tierra separada.
- El cableado del sistema de alarma debe ser el único que pase por la tubería eléctrica.
- Cableado par trenzado para proporcionar protección adicional contra interferencias eléctricas externas.

En la figura 7 se muestra el diagrama de conexión de los sensores.

Figura 7

Diagrama de conexión del sensor 4W-B.



Nota: En la figura se muestran un diagrama de conexión para uno o más sensores. Obtenido de (Systemsensor, 2023).

5.1.5. Dispositivos de inicio

De acuerdo con la sección 10.4.5.3 de la norma NFPA-72. La activación manual de las alarmas pueden ser accionadas de dos distintas maneras:

1. Una estación manual de alarma de incendio listada.
2. Un medio operado mediante llave.

- **Estación manual de activación de alarma contra incendio serie BG-12:**

“Las estaciones manuales proporcionan una señal de entrada de inicio de alarma a los paneles de control de alarma contra incendios (FACP) convencionales.” (*Pull Stations \textbar Manual Initiating Devices*, 2023). La señal de activación se da mediante un accionamiento manual del mecanismo, es decir, se asemeja al funcionamiento de un interruptor. En la figura 8 se muestra la estación manual BG-12. La estación manual cumple con múltiples aplicaciones con el instalador y el usuario.

Características:

- Cumple con ADA (Americans with Disabilities Act) y fuerza de tracción máxima de cinco libras. Puede ser de acción simple o doble, según el modelo, pero diseñado para evitar falsas alarmas cuando se golpea o sacudida.
- Push in/Push down los pestillos de la manija en la posición hacia abajo para indicar claramente que la estación ha sido operada. Como se muestra en la figura 8.
- La estación se puede abrir para inspección y mantenimiento sin iniciar una alarma.

Figura 8

Estación manual serie G-12.



Nota: En la figura se visualiza la estación manual requerida para el módulo. Obtenido de: (*Pull Stations \textbar Manual Initiating Devices*, 2023).

5.1.6. Dispositivos de notificación

Los dispositivos de notificación se rigen mediante los códigos establecidos dentro del capítulo 18 de la norma NFPA-72, referente a “Aparatos de notificación”.

- **Sirena/luz estroboscópica autónoma SH-816S-SQ/R:**

Las sirenas estroboscópicas son capaces de emitir “señales acústicas entre los rangos de 75 a 120 decibeles y visuales de 15 a 110 candelas al generarse un evento de detección de incendio.” (Delgado Huanca, 2021).

En la figura 9 se muestra la sirena estroboscópica SH-816s. Algunas de sus características se muestran en la tabla 4.

Figura 9

Sirena estroboscópica SH-816S-SQ.



Nota: En la figura se puede observar la sirena que será a utilizar en el sistema de alarma contra incendios. Obtenido de (Enforcer, 2020).

5.1.7. Cableado

El cableado del sistema de monitoreo debe hacerse con cable antilflama, que son especialmente diseñados para sistemas de alarmas contraincendios y cumplen la certificación ISO 9001.

El cableado debe cumplir la condición de que, al ocurrir un fallo en el circuito de inicio o energía, debe mostrar una notificación, indicando la falla de circuito, de igual manera en caso de que se retire un detector de humo del circuito integrado debido a la interrupción en el circuito de suministro de energía. Para notificar dicha falla "se debería llevar un lazo de retorno nuevamente hasta el último dispositivo alimentado con energía y el relé de supervisión de la energía para incorporarse en el dispositivo de fin de línea." (*NFPA 72: Código Nacional de Alarmas y Señalización contra Incendios*, 2016). Como se muestra en la figura 10.

Tabla 4

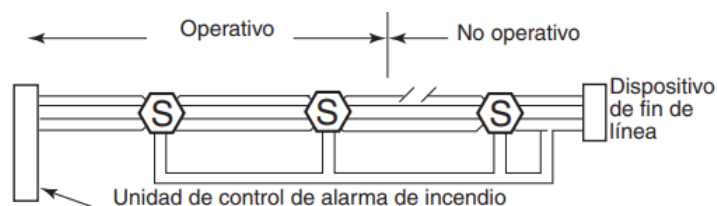
Especificaciones de la sirena estroboscópica SH-816S-SQ.

Características	Especificaciones
Modelo	SH-816S-SQ
Tipo	Sirena con luz estroboscópica
Voltaje de operación	9-15 VDC
Consumo de corriente (max)	Luz estroboscópica: 190mA 12VDC Sirena: 680mA 12VDC Total: 870mA 12VDC
Volumen de sirena	120 dB
Rango altavoz	25 W pico
Frecuencia de destello estroboscópico	75 por minuto
Salida Tamper	100mA 30VAC/VDC
Índice de protección	IP65
Temperatura de funcionamiento	-20 ° a 55 ° C
Material	Plástico de policarbonato
Dimensiones	155x178x115mm

Nota: En la tabla se describen especificaciones técnicas de la sirena SH-816S-SQ. Obtenido de (Enforcer, 2020).

Figura 10

Circuito con detectores de humo de cuatro hilos y relé de supervisión individual.



Nota: En la figura se observa un circuito de referencia para prueba de detención fallas de sensores de hilos. Obtenido de (NFPA 72: Código Nacional de Alarmas y Señalización contra Incendios, 2016).

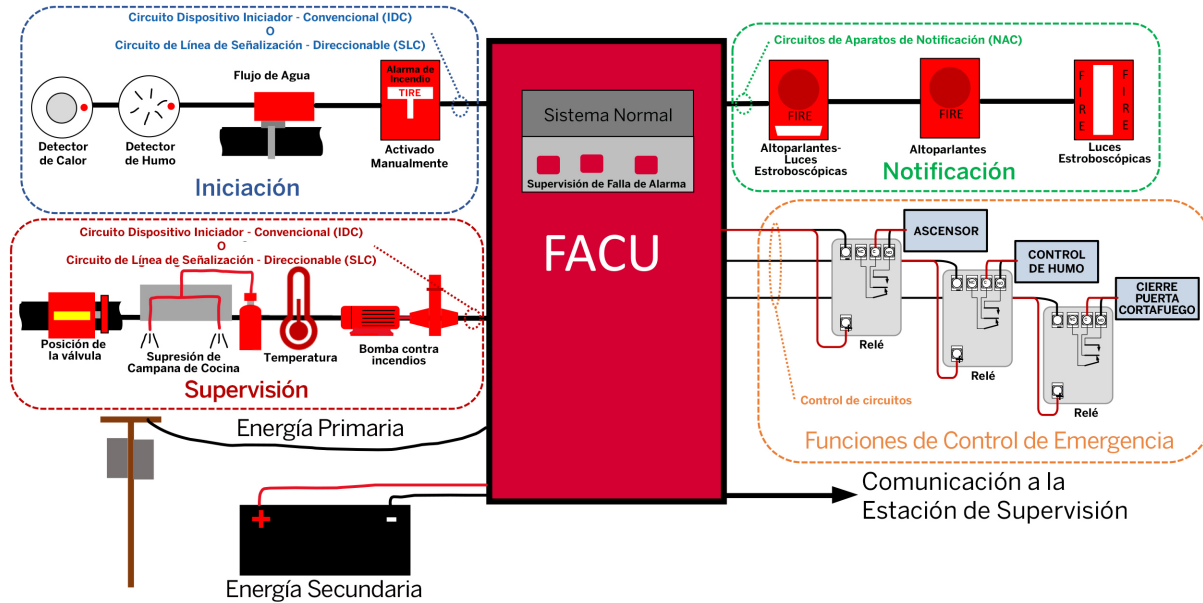
5.2. Diseño del sistema de monitoreo de alarma contra incendios

5.2.1. Diagrama de flujo de funcionamiento

El funcionamiento del monitoreo del sistema de alarma contra incendio se da en varias etapas en donde se involucran dispositivos electrónicos de inicio, notificación y monitoreo. El funcionamiento consiste en la recepción de información mediante los dispositivos de inicio, los cuales son direccionados hasta una interfaz de monitoreo en la nube, donde se podrá controlar y operar el sistema, tal descripción se visualiza en la figura 11:

Figura 11

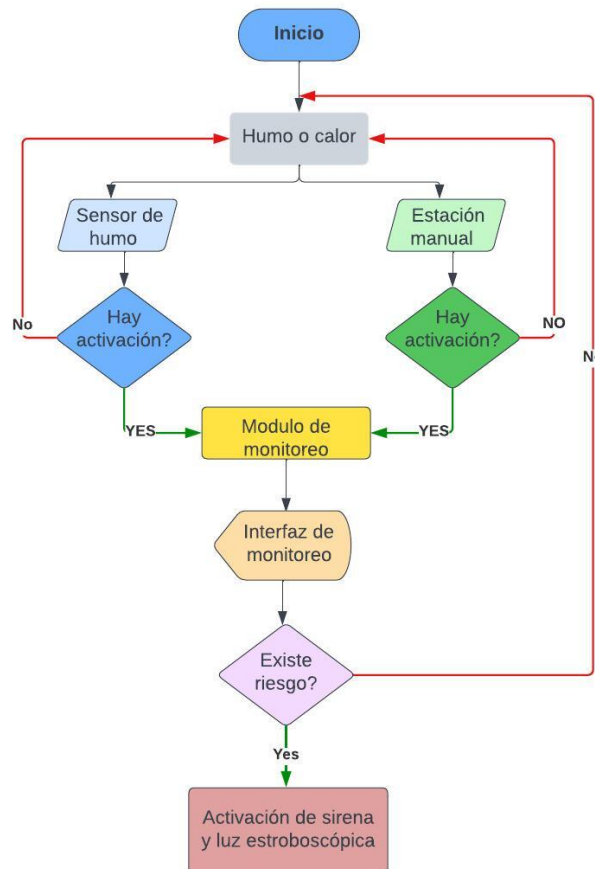
Componentes básicos de un sistema de alarma contra incendios NFPA.



Nota: En la figura se visualiza el diagrama de componentes del sistema de alarmas NFPA. Obtenido de (Mahoney, 2021)

Figura 12

Diagrama de bloque del funcionamiento sistema de monitoreo de alarma contra incendios.



Nota: En la figura se muestra el diagrama de bloques correspondiente al funcionamiento del sistema de monitoreo de alarma.

En la figura 12 se muestra el funcionamiento del sistema de monitoreo, que comprende la integración de distintos dispositivos de inicio que envían señales, información, datos hacia los pines de la unidad de procesamiento (Raspberry Pi 4B) la cual sube dichos datos a la nube mediante la herramienta de programación Node red que permite la interconexión mediante la red. Esto permite el control total de monitoreo del sistema de alarma y a su vez la activación de los actuadores o dispositivos de notificación. Previo al diseño de la unida de control se debe tener en cuenta no solo los requisitos a cumplir por la normativa, sino también los requerimientos proporcionados por la Universidad.

5.2.2. Requisitos y requerimientos

En referencia a los requisitos y requerimientos para el diseño del prototipo planteados por la Universidad y bajo el reglamento del NFPA código 72 referente al Código Nacional de Alarmas de Incendio y Señalización. Los requerimientos son los siguientes:

- Los dispositivos del sistema de alarma deben estar montados de manera que el funcionamiento no se vea distorsionado por vibraciones o sacudidas.
- Los dispositivos deben instalarse de acuerdo con las especificaciones de voltaje, temperatura y humedad de cada fabricante.
- Los dispositivos de inicio deben instalarse en lugares que facilite su inspección prueba y mantenimiento.
- Los detectores no deben incrustarse en la superficie de montaje salvo que hayan sido probados o certificados para montaje empotrado.
- La selección y ubicación de los detectores de humo debe tener en cuenta tanto las características de desempeño del detector como las áreas en las que los detectores van a ser instalados.
- Pulsantes manuales para activación de alarmas.
- Pulsante manual para indicar otro tipo de emergencia.
- Sirena estroboscópica.
- Notificación de emergencia mediante IoT.

Los dispositivos que van dentro de la unidad de control, así como los dispositivos de inicio deben cumplir:

- Entre el 85 y 110 de voltaje nominales de alimentación primario y secundario.
- Funcionamiento a temperatura ambiente de 0 - 49°C.
- Humedad relativa del 85.

A continuación, la tabla 5 describe las características de componentes seleccionados para el sistema de alarma.

5.2.3. Componentes del sistema

Tabla 5

Tabla de componentes que cumplen los requisitos y requerimientos.

Componentes	Dispositivo a usar
Placa electrónica	Raspberry Pi 4
Sensor de humo	Sensor fotoeléctrico 4W-B
Sirena de alarma	Sirena SH-816s-SQ
Palanca de accionamiento	Estación manual BG-12

Nota: La tabla muestra los diferentes componentes a tener en cuenta para el diseño de la unidad de control.

En referencia a la tabla 5, es posible determinar el suministro de energía que será necesario para el funcionamiento del sistema de alarma contra incendios, siguiendo las indicaciones de la sección 10.6.3 y 10.6.4 de la norma NFPA 72, referente las fuentes de suministro de energía y suministros de energía ininterrumpida. Dichas norma exigen dos suministros de energía independientes: una fuente primaria y una fuente secundaria.

5.2.4. Alimentación del sistema

De acuerdo con la tabla 6, se puede observar el consumo total del sistema de alarma. Según las especificaciones establecidas en la NFPA 72, sección 10.6.3.2 es necesario que el sistema de alarma cuente con dos fuentes de alimentación. La fuente de alimentación primaria debe ser el suministro de electricidad proveniente de la red eléctrica del lugar, mientras que la fuente de alimentación secundaria puede ser una batería.

Tabla 6

Alimentación del sistema de alarma.

Dispositivos	Cantidad	Carga total
Raspberry Pi 4B	1	1250mA
Sensor de humo fotoeléctrico 4W-B	1	20 mA
Sirena SH-816s-SQ	1	680mA
Luz estroboscópica	1	190mA
	Total	2140mA

Nota: En la tabla se muestra la carga total necesaria para el sistema de alarma contra incendios.

La sección 10.6.7.2 de la norma NFPA 72 indica que “el suministro de energía secundaria debe tener una capacidad suficiente como para que el sistema de alarma funcione por un mínimo de 24 horas”(NFPA 72: *Código Nacional de Alarmas y Señalización contra Incendios*, 2016).

5.2.5. Análisis de consumo de energía

Tras revisar las características eléctricas de cada componente, es posible calcular el consumo por hora, lo que permite determinar el tamaño adecuado de la batería. Es importante tener en cuenta ciertos datos clave como el voltaje y amperaje de consumo, que posibilitan realizar el cálculo de forma precisa y efectiva.

Tabla 7*Consumo Wh durante 24h.*

Componente	Consumo W (V*A)	Consumo W por 24h+10%
Raspberry Pi 4	6,25 W	165 Wh
Sensor de humo fotoeléctrico 4W-B	0,54 W	14,25 Wh
Sirena	0,435 W	11,48 Wh
Total de consumo por 24h		190,73 Wh

Nota: En la tabla observamos el consumo Wh de cada componente.

Utilizando la tabla 7, se determina el consumo total del sistema de alarma en un día completo, este valor es un total de: 190,73 Wh, lo cual corresponde al consumo medio energético diario de las cargas continuas y se utiliza para determinar el consumo medio energético diario, que se calcula mediante la fórmula 1:

$$L_{md} = \frac{L_{md} \times DC + \frac{L_{md} \times AC}{\eta_{inv}}}{\eta_{bat} \times \eta_{con}} \quad (1)$$

Donde:

- L_{md} = Consumo energético diaria.
- $L_{md} \times DC$ = Consumo energético medio de cargas continuas.
- η_{bat} = Coeficiente de eficiencia de la batería.
- η_{con} = Coeficiente de eficiencia de conectores eléctricos.

Considerando un coeficiente de eficiencia de la batería de $\eta_{bat} = 0,95$, un coeficiente de conectores eléctricos η_{con} igual a 1 y un consumo energético de 190,73 Wh, el consumo energético diario es de:

$$L_{md} = \frac{190,73Wh}{0,95 \times 1} = 200,76 \quad (2)$$

Con el dato obtenido de la ecuación 2, se calcula el dimensionamiento necesario de la batería utilizando la fórmula 3:

$$C_{ne} = \frac{L_{md} \times N}{V_{bat} \times P_{dmax} \times F_{ct}} \quad (3)$$

Donde:

- V_{bat} = Voltaje nominal del sistema.
- N = Número de días de autonomía.
- P_{dmax} = Porcentaje máximo de descarga de batería= 0.7.
- F_{ct} = Factor de compensación térmica = 1.

Para determinar el amperaje hora adecuado de la batería, se deben tener en cuenta ciertos parámetros. El consumo medio energético diario L_{md} el cual determinamos en la ecuación 2, la descarga máxima de la batería P_{dmax} que es del 70%, el período de autonomía N es de 1 día y el factor de compensación térmica F_{ct} que es de 1. Con estos valores, se calcula el amperaje hora adecuado para la batería que se necesita.

$$C_{ne} = \frac{200,76}{12 \times 0,7 \times 1} = 23,9Ah \quad (4)$$

Con el dato de C_{ne} obtenido en la ecuación 4 se concluye que el sistema de alarma requiere de una batería de 12V y 23,9 Ah para una autonomía de un día en caso de exista un fallo de conexión de la fuente primaria.

5.2.6. Elección de la batería

Después de realizar un análisis para determinar el amperaje-hora necesario para el funcionamiento durante 24h del sistema de alarma, se considera a la batería Steren 12V 24Ah como la más adecuada para cumplir con el código 10.6.7.3 de la norma NFPA 72 referente a "Suministro de energía secundario para sistemas de alarma de incendio y sistemas de comunicaciones de emergencia de instalaciones protegidas". Para asegurar de que el sistema de alarma funcione de manera óptima y confiable, la batería se muestra en la figura 13 y sus especificaciones de describen en la tabla 8.

Figura 13

Batería Steren de 12V 24Ah.



Nota: Batería de fuente secundaria del sistema de alarma. Obtenido de (Steren, 2022).

Tabla 8

Especificaciones de la batería Steren 12V 24Ah.

Características	Especificaciones
Voltaje nominal	12 Vdc
Capacidad nominal	24 Ah
Tipo de batería	Plomo ácido
Ciclos de carga/descarga	1000 ciclos
Tecnología	Voltaje de carga flotante
Temperatura de trabajo	300°C
Dimensiones	18x17x16 cm (WxLxH)
Terminales	Tornillo de 1,06 cm

Nota: En la tabla se muestran algunas de las características de la batería Steren 12V 24Ah. Obtenido de (Steren, 2022).

5.2.7. Funcionamiento del sistema electrónico

La activación del sistema de alarma tiene dos modos de funcionamiento, de manera automática y manual. A continuación se describe el funcionamiento de cada tipo:

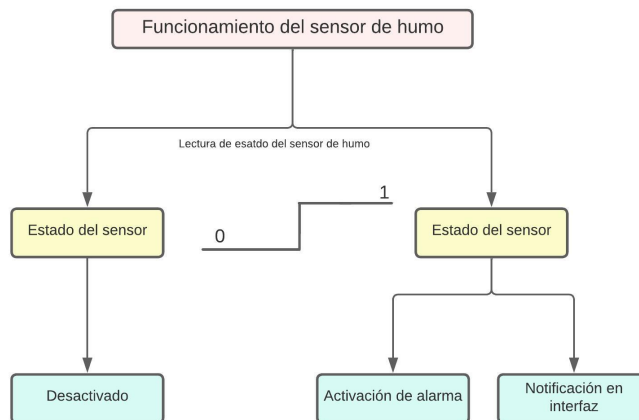
1. Activación automática:

La activación automática se ejecuta de la siguiente manera:

- Los sensores de humo monitorean constantemente el ambiente.
- Si alguno de los sensores detecta humo o un aumento brusco de temperatura, notifica inmediatamente al sistema de monitoreo.
- La unidad de control recibe las señales de los sensores y activa una notificación de emergencia.
- De igual manera se activa la alarma en la sirena estroboscópica emitiendo un sonido y una luz que alerte a las personas.

Figura 14

Activación de la alarma automáticamente.



Nota: En la figura podemos observar el funcionamiento de la activación de la sirena y la notificación de manera automática dependiendo del estado del sensor.

En resumen, la activación se da con el sensor de humo, que envían una señal eléctrica hacia la unidad de control. Dicha señal activa una notificación en la interfaz del sistema de monitoreo y a su vez activa de forma automática la sirena estroboscópica indicando una señal de peligro de incendio. Como se muestra en la figura 14.

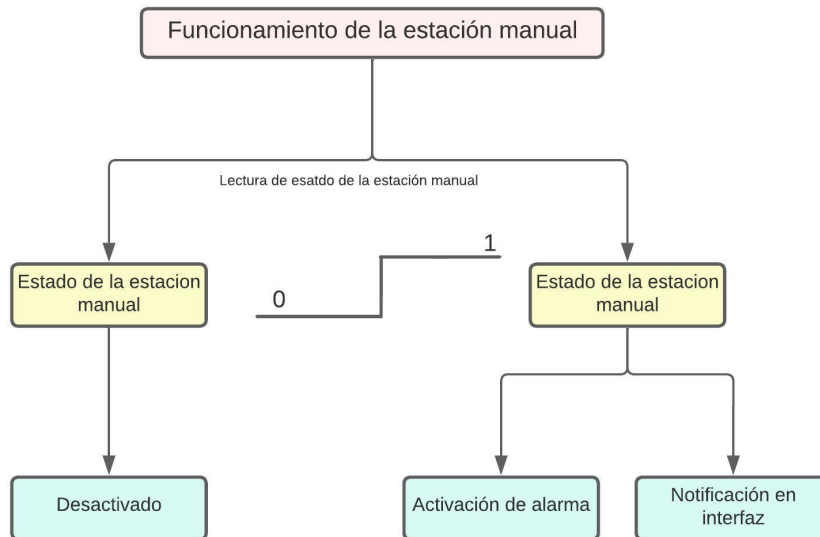
2. Activación manual:

La activación manual del sistema de alarma se realiza de la siguiente forma:

- Si alguien detecta un incendio sin que los sensores lo hayan activado, puede accionar la palanca de alarma manual.
- Cuando se activa la palanca, se emite una señal directamente a la unidad de control.
- La unidad de control recibe la señal y activa la alarma de sirena y luz estroboscópica.

Figura 15

Funcionamiento de la estación manual.



Nota: En la figura podemos observar el diagrama de funcionamiento dependiendo el estado de la estación manual.

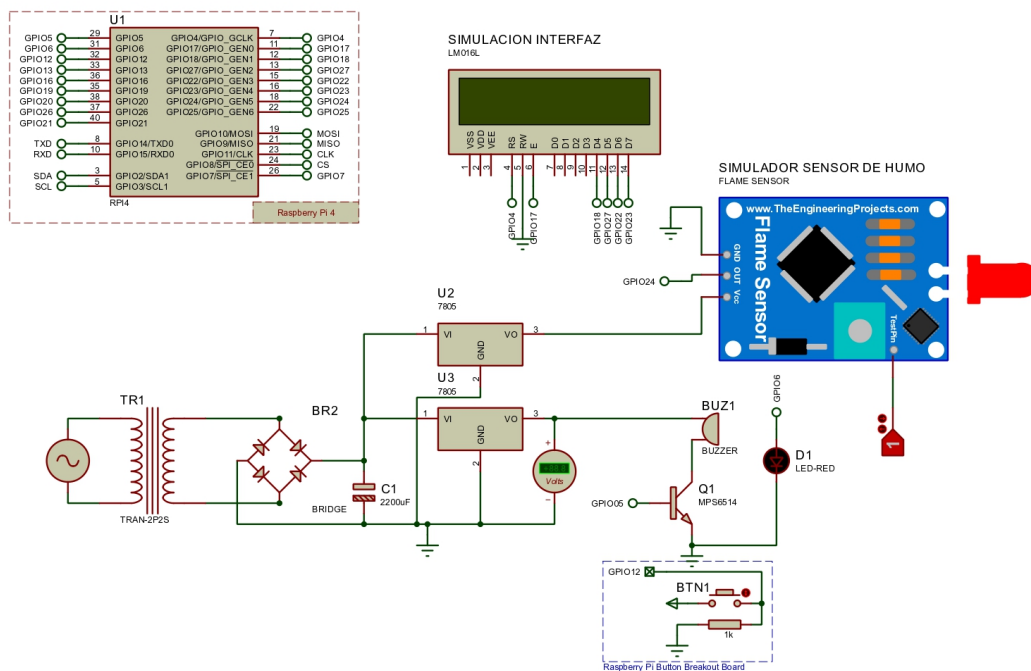
El sistema de alarma cuenta con una estación manual disponible para que pueda ser activado por las personas en caso de que los sensores no detecten un riesgo de incendio o un incendio como tal, de igual manera notifica al usuario por la interfaz y activa la alarma. Como se muestra en la figura 15, de esta manera se asegura la detección rápida y la evacuación necesaria.

5.2.8. Simulación del sistema electrónico

Previo a implementar el sistema de monitoreo se realiza la simulación de todo el sistema utilizando el software de simulación Proteus, con el fin de evaluar el diseño y realizar pruebas para garantizar un sistema de alarma eficiente y confiable para la detección y respuesta temprana frente a riesgos de incendio.

Figura 16

Simulación del sistema electrónico de la unidad de control.



Nota: En la figura se observa la simulación del funcionamiento electrónico de la unidad de control desarrollado en Proteus.

En la figura 16 se muestra la simulación, en la cual se puede corroborar el funcionamiento del encendido de la sirena mediante la estación manual, la activación de alarma automática e incluso la notificación en la interfaz de usuario. También se indica la conexión de la fuente de alimentación para el sistema de alarma y la conexión necesarias a realizar en la unidad de control que usa la placa Raspberry Pi 4B como unidad de procesamiento.

En la simulación, la unidad de procesamiento, se conecta un botón de activación al pin GPIO12, un sensor de humo conectado al pin digital GPIO24, una pantalla LCD de 16x2 que simula la notificación que envía la unidad de procesamiento al sistema de monitoreo, conectado de igual manera a los pines digitales GPIO4, GPIO17, GPIO18, GPIO27, GPIO22, GPIO23 (pines que solo se usan en la simulación). Finalmente se simulan los dispositivos de notificación tales como la alarma y la luz estroboscópica conectadas en los pines digitales GPIO05 y GPIO06 respectivamente.

5.2.9. Programación de la simulación

La programación realizada en el software Proteus es posible utilizar el compilador de Python 3; código similar al que se utilizara en la implementación del sistema de monitoreo, despojando el código correspondiente a la pantalla LCD puesto que para la notificación en una interfaz de usuario se usara bloques de programación en Node-RED. El código de programación empleado en la simulación se describe a continuación:

```
1 # Processor: RPI4
2 # Compiler: Python 3 (Proteus)
3 from goto import *
4 import time
5 import RPi.GPIO as GPIO
6 import sleep
```

Listing 1: Importación de librerías.

El código presentado en el listing 1 inicia llamando distintas librerías que comprende un conjunto de funciones que permiten realizar distintos comandos, que podrían ser utilizadas dentro de la programación.

```
1 #import serial
2 GPIO.setmode(GPIO.BOARD)
3 GPIO.setmode(GPIO.BCM)
4 GPIO.setwarnings(False)
5 '''
6 define pin for lcd
```

```

7     '''
8     # tiempo
9     E_PULSE = 0.0005
10    E_DELAY = 0.0005
11    delay = 1
12    buzzer=37
13    GPIO.setup(buzzer, GPIO .OUT)

```

Listing 2: Comunicación serial.

El código mostrado en el listing 2, comprende en realizar la comunicación o importación serial la cual se encarga en el envío de datos de forma secuencial en un canal. De la misma manera se declaran variables de tiempo pulsación, detención y parada para los pines de la pantalla LCD.

```

1     #GPIO para simulador de sensor ,
2     sirena, luz, palanca
3     flame_Sensor = 18
4     red_light = 31
5     palanca = 32
6     Buzzer= 29
7     #salidas y entradas
8     GPIO.setup(LCD_D7, GPIO.OUT) # DB7
9     GPIO.setup(flame_Sensor, GPIO.IN) #DB7
10    GPIO.setup(palanca,GPIO.IN,
11    pull_up_down=GPIO.PUD_UP)
12    GPIO.setup(red_light, GPIO.OUT)
13    GPIO.setup(Buzzer, GPIO .OUT)

```

Listing 3: Declaración de periféricos.

Para la simulación es necesario determinar los periféricos de entrada y salida de la unidad de procesamiento, como se muestra en el listing 3; para el caso de simulación se declara los periféricos de entrada digital para el sensor de flama y el botón. Los pines digitales de salida será la pantalla LCD, la luz y el buzzer que simula una sirena.

```

1     def lcd_string(message,line):
2     # Send string to display
3     message = message.ljust(LCD_WIDTH, " ")
4     lcd_byte(line, LCD_CMD)
5     for i in range(LCD_WIDTH):
6         lcd_byte(ord(message[i]),LCD_CHR)
7     lcd_init()
8     lcd_string("ENCENDIDO",LCD_LINE_1)

```



```

9     time.sleep(1)
10    # Define delay between readings
11    delay = 5

```

Listing 4: Notificación en pantalla.

Para visualizar las notificaciones en la pantalla LCD se declaran distintas funciones para iniciar el display como se muestra en el listing 4. En ese apartado se realiza el envío de datos a los pines, habilitación de pines y finalmente se muestran notificaciones en la pantalla.

```

1     while 1:
2         # simulacion de activacion automatica
3         if GPIO.input(flame_Sensor):
4             lcd_string("ALARMA ACTIVADA",LCD_LINE_1)
5             GPIO.output(Buzzer, True)
6             GPIO.output(red_light, True)
7         else:
8             lcd_string("ESTADO NORMAL",LCD_LINE_1)
9             GPIO.output(Buzzer, False)
10            GPIO.output(red_light, False)
11            break
12        break
13        #simulacion activacion manual
14        while True:
15            if GPIO.input(palanca)==True:
16                lcd_string("ALARMA ACTIVADA",LCD_LINE_1)
17                GPIO.output(Buzzer, True)
18                GPIO.output(red_light, True)
19            else:
20                lcd_string("ESTADO NORMAL",LCD_LINE_1)
21                GPIO.output(Buzzer, False)
22                GPIO.output(red_light, False)
23                sleep      (0.1)

```

Listing 5: Código de simulación.

Finalmente, como se muestra en el listing 5, se realiza la programación de la activación automática y activación manual con condicionales while que permite repetir la ejecución de las instrucciones dispuestas en cuanto se cumpla la condición programada.

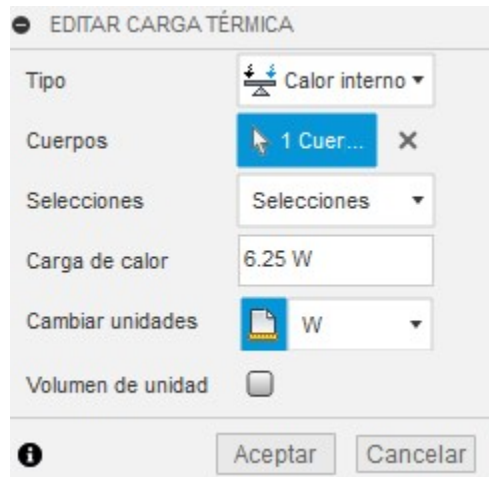
5.2.10. Simulación de temperatura de la unidad de procesamiento

La simulación de la temperatura de la unidad de procesamiento se realiza con la ayuda del software Fusion 360 que es una plataforma de software de diseño “CAD, CAM, CAE y de circuitos impresos de modelado 3D basada en la nube para el diseño y la manufactura de productos” (Autodesk, 2023). En el software Fusion 360 se puede crear modelos 3D, realizar simulaciones, generar programas y colaborar en un entorno en línea.

Una vez que se ha obtenido el modelo 3D de la unidad de procesamiento, se procede a entrar al entorno de simulación de refrigeración electrónica y se aplica una carga de calor interno, la cual será atribuida al chip principal de la unidad de procesamiento. Como se muestra en la figura 17.

Figura 17

Calor interno generado por la unidad de procesamiento.

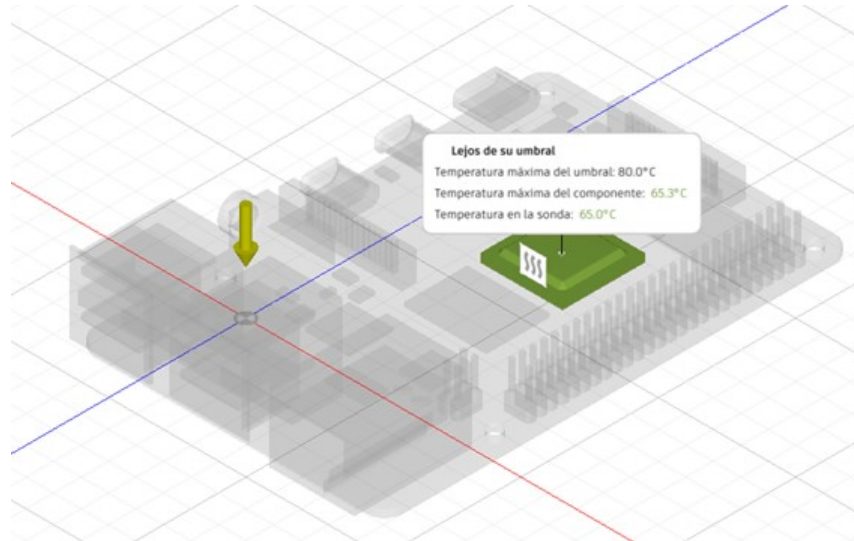


Nota: En la figura se visualiza los parámetros de carga térmica a aplicar.

Después de asignar la carga de calor interno, se realiza la simulación, para lo cual se toma en cuenta el dato de los 6.25W, que es el consumo eléctrico que tendrá la unidad de procesamiento en su estado de estrés máximo.

Figura 18

Simulación del comportamiento térmico sin disipación.



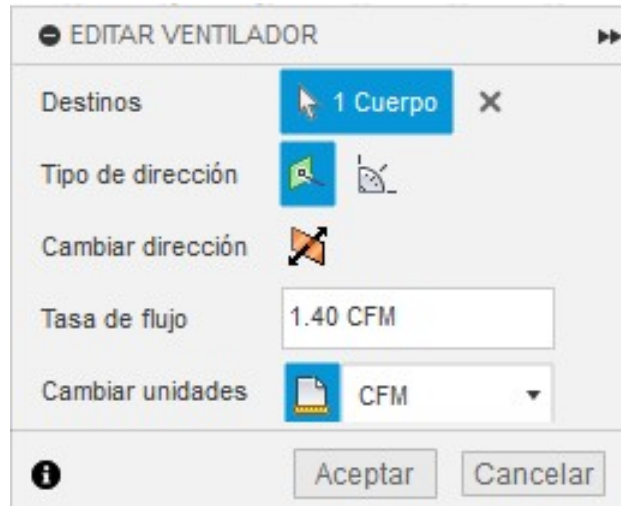
Nota: En la figura observamos los resultados obtenidos de la simulación.

En la figura 18, se observa cual será la temperatura máxima que alcanzará por la unidad de procesamiento siendo esta de 65°C sin tener en cuenta ningún tipo de disipación pasiva o activa. En la unidad de procesamiento, al estar dentro de un espacio cerrado se pueden generar altas temperaturas, ocasionando algún fallo en el funcionamiento de la unidad de control. Para evitar dicha situación, se hará uso de un ventilador que, de igual manera, mediante Fusion 360 se puede realizar una simulación considerando un disipador y un ventilador que otorguen un mayor flujo de aire para enfriar la unidad de procesamiento.

Dentro del entorno de simulación de refrigeración electrónica, tenemos opciones para el disipador y el ventilador. Para el caso del disipador, basta seleccionar la pieza que se utilizará como disipador, a la cual se le debe asignar algún material conductor de calor. Para el ventilador, se debe asignar la tasa de flujo de aire que va a proporcionar el ventilador, la dirección del ventilador es hacia donde queremos disipar el aire, por lo que se selecciona la primera opción en el tipo de dirección, como se muestra en la figura 19.

Figura 19

Cuadro de configuración de ventilador.



Nota: En la figura observamos los parámetros establecidos para la simulación con ventilador.

Asignar un ventilador es necesario para que la unidad de procesamiento funcione durante 24 horas; depende del consumo energético, el entorno de trabajo y la temperatura ambiente en la que se encuentra. Con estos datos se busca las especificaciones del ventilador que sean compatibles con la unidad de procesamiento y cumplan con los parámetros establecidos en la simulación, el ventilador que se muestra en la figura 20 cumple con dichos parámetros. Características de voltaje de trabajo, flujo de aire y la velocidad del ventilador (RPM) son detallados en la tabla 9.

Figura 20

Ventilador Original para Raspberry Pi 4 con Disipador.



Nota: La figura se muestra el ventilador y disipador de la marca Raspberry Pi 4. Obtenido de (Raspberry, 2023).

Tabla 9

Características del ventilador para Raspberry Pi 4 con Disipador.

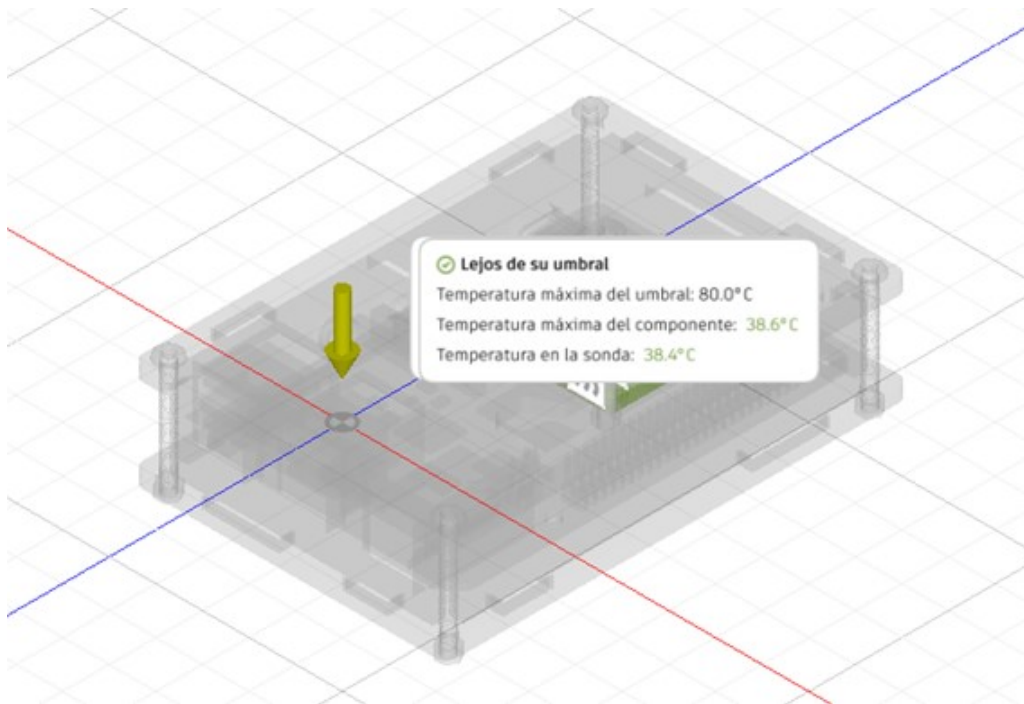
Características	Especificaciones
Marca	Raspberry Pi
Voltaje de funcionamiento	5 V
Peso	0,05 kg
Control de temperatura	automático
Flujo de aire	1,4 CFM (metros cúbicos por hora)
Velocidad máxima de rotación	13200 RPM
Disipador térmico con almohadilla	18mm × 18mm × 10mm (WxLxH)

Nota: En la tabla se muestran algunas de las características del ventilador para Raspberry Pi 4. Obtenido de (Raspberry, 2023).

El dato del flujo de aire correspondiente a utilizar en la simulación es obtenido de la tabla 9, este dato es utilizado en el cuadro de configuración del ventilador, como se muestra en la figura 19 para determinar la temperatura máxima alcanzada por la unidad de procesamiento en su estado de estrés máximo teniendo en cuenta una disipación activa.

Figura 21

Simulación del comportamiento térmico con disipación.



Nota: En la figura observamos los resultados obtenidos de la simulación.

La figura 21 muestra que, al utilizar un ventilador la temperatura máxima ha sido de 38.6°C, lo cual es significativamente menor a los resultados obtenidos en la figura 18. Con estos resultados podemos determinar que el uso de un ventilador es necesario para disipar eficientemente el calor emitido por la unidad de procesamiento evitando fallos en su funcionamiento producidos por las altas temperaturas.

Cabe recalcar que el análisis térmico realizado se fundamenta en que la unidad de procesamiento estará en un entorno exterior, debido a que los requisitos a cumplir por la normativa NFPA para un gabinete exigen que cuente con un grado de protección IP, esto implica que el gabinete estará aislado lo que impide un flujo de aire adecuado para refrigerar la unidad de procesamiento del sistema de alarma contra incendios diseñado.

Motivo por el cual se ha optado por dejar la unidad de procesamiento en un entorno exterior; lo que a su vez cumple con el código 10.3.5 del capítulo 10, referente a fundamentos, dicho código establece que la unidad de procesamiento debe tener la capacidad de llevar a cabo las funciones previstas cumpliendo con varias condiciones, entre las cuales se establece que la unidad de procesamiento debe operar dentro de un rango de temperatura de 0°C a 49°C. Dicho rango se corrobora en el análisis térmico realizado, teniendo en cuenta las condiciones de ubicación de la unidad al descubierto.

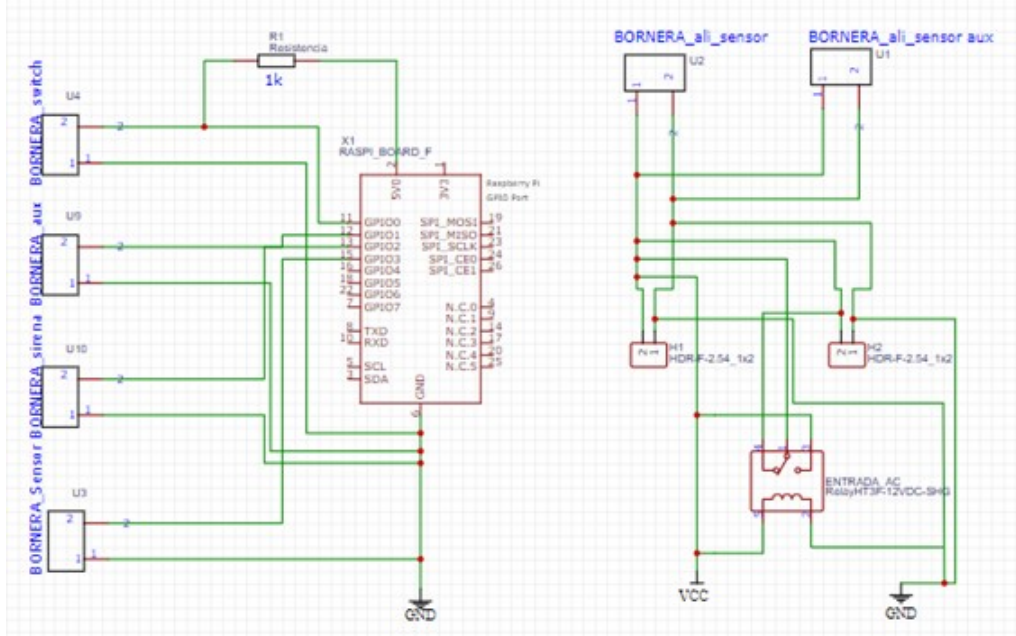
Otros de los motivos por los cuales se ha optado dejar la unidad de procesamiento en un entorno exterior es que facilita la disipación de calor de manera efectiva ya que el comportamiento térmico y flujo de aire en dicho entorno es igual a lo observado en la simulación realizada anteriormente. Además, simplifica las tareas de mantenimiento, reparación y ajuste lo que posibilita la visibilidad, supervisión y el control de la unidad.

5.2.11. Diseño de la placa PCB

El diseño de una placa PCB es un proceso crucial en la producción de algún dispositivo electrónico. Para ello, es necesario tener en cuenta ciertas especificaciones tales como: los componentes seleccionados, la fuente de energía del circuito y las conexiones adecuadas en la PCB. Partiendo del diseño esquemático que se utilizó en la simulación, dentro del propio Proteus se puede realizar el diseño de la PCB. Se debe tener en cuenta de que algunas de las conexiones realizadas en el esquemático de la simulación no son las que se van a emplear.

Figura 22

Circuito esquemático utilizado para la PCB.



Nota: En la figura se destacan las conexiones a los puertos GPIO utilizados.

Se puede observar en la figura 22 que algunos de los componentes de la PCB han sido sustituidos por bornes. Esto con el fin de garantizar la facilidad el cambio y la ampliación de más dispositivos de inicio o notificación. La conexión de uno o varios componentes a la unidad de control solo requiere el ajuste o desajuste de las borneras correspondientes, simplificando la operación y el mantenimiento.

Previo a pasar del esquemático a una placa PCB se realiza el calculo del grosor de la pista: para ello se aplica las siguientes formulas:

$$Ancho_{pista} = \frac{\acute{a}rea}{grosor \times 1,978} \quad (5)$$

En la ecuación 5 se reemplaza el dato del grosor que será igual a 35 micras o $1 \frac{oz}{ft^2}$ de acuerdo a los estándares utilizados en las PCBs y el área corresponde a:

$$\acute{a}rea = \left(\frac{I_{max}}{k_1 \times \Delta T^{k_2}} \right)^{\frac{1}{k_3}} \quad (6)$$

La ecuación 6 proporciona información sobre el área a determinar en la cual utilizamos las constantes k y ΔT , la cual representa la diferencia de temperatura entre la placa PCB y el ambiente (normalmente 25°C). Dado que la corriente máxima que atravesará la unidad de control es la misma que la consumida por la unidad de procesamiento, es decir, $1,25\text{ A}$ y la temperatura máxima será de 65°C según los resultados de la figura 18, se sustituye los valores en la ecuación 6.

$$\text{área} = \left(\frac{1,25\text{A}}{0,0647 \times 40^{0,4281}} \right)^{\frac{1}{0,6732}} = 10,5 \quad (7)$$

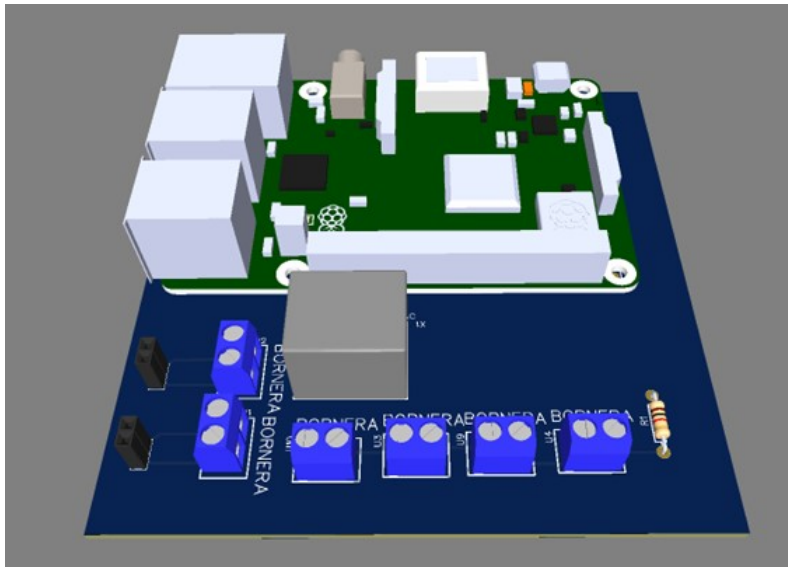
Reemplazando el valor de la ecuación 7 en la ecuación 5 nos un valor de:

$$\text{Ancho}_{\text{pista}} = \frac{10,5}{1 \times 1,978} = 5,3\text{th} \quad (8)$$

En la ecuación 8 se muestra que el ancho necesario para efectuar el trazo de pista en el programa es de $5,3\text{ th}$ milésimas de pulgadas. Alternativamente, se podría utilizar un ancho de 6 th . De esta manera la PCB a utilizar en la unidad de control queda de la siguiente manera:

Figura 23

PCB usada para realizar conexiones entre la unidad de procesamiento y los periféricos.



Nota: En la figura se puede observar la placa PCB y los componentes implementados en ella.

En la figura 23 podemos observar la placa PCB con sus respectivas conexiones, en las cuales se incluyen las diferentes borneras que funcionarán para conexión de señal de los dispositivos de inicio y notificación. De igual manera se muestran dos conectores de alimentación los cuales pasarán por un relé con el fin de proveer energía al sistema de alarma, tanto con corriente suministrada por la red eléctrica o en caso de fallo eléctrico se hace uso de la batería.

5.2.12. Elección del gabinete

El gabinete eléctrico es uno de los componentes más importante de cualquier sistema eléctrico. Su diseño y características mecánicas son cruciales para asegurar que proteja adecuadamente los componentes eléctricos de las condiciones ambientales y otros elementos externos que pueden dañarlos.

Figura 24

Gabinete liviano Beacoup.



Nota: En la figura se muestra el gabinete eléctrico para fuente de alimentación. Obtenido de (Inselec, 2023).

Tabla 10

Características del gabinete eléctrico Beacoup.

Características	Especificaciones
Marca	Beacoup
Material	Acero al carbono
Dimensiones	30x30x20 cm
Grado de protección	IP 42
Certificación	IEC 60068

Nota: En la tabla se muestran algunas de las características del gabinete Beacoup. Obtenido de (Inselec, 2023).

La figura 24 se muestra el gabinete a utilizar tanto para la batería como para el cargador de esta. Este cuenta con la certificación IEC 61439, que regula la construcción de paneles de control y potencia eléctrica. Otras características se detallan en la tabla 10.

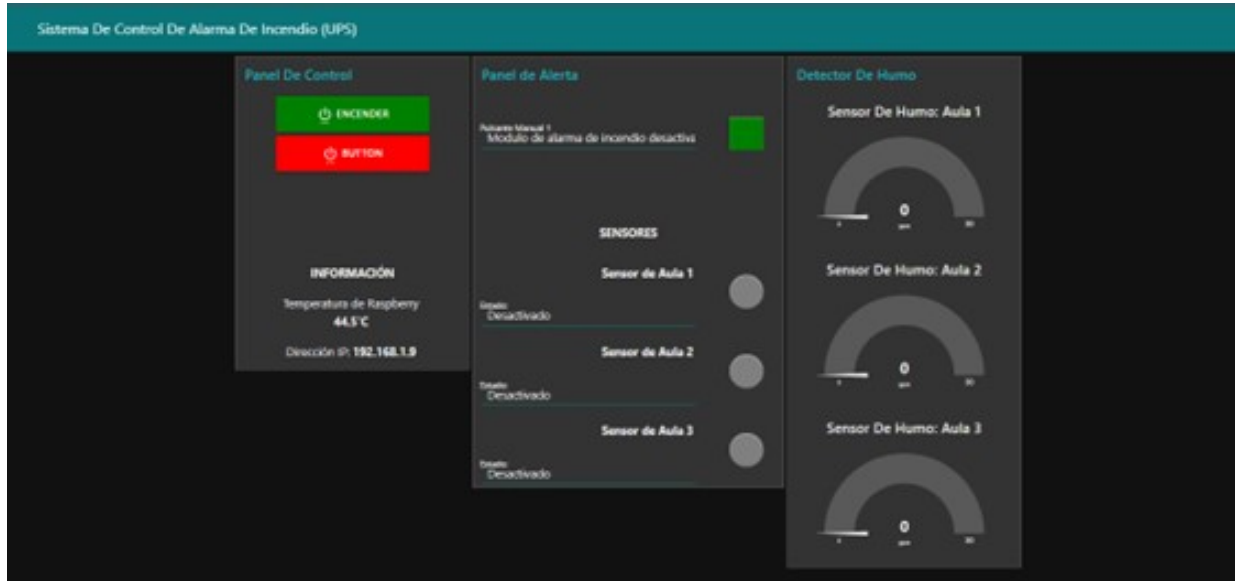
5.3. Implementación de los dispositivos de inicio del sistema de monitoreo de alarma contra incendios

Para la implementación de los dispositivos de inicio a la unidad de control se realiza mediante un código con el lenguaje de programación Python y programación de nodos en Node-RED. Todo esto será visualizado en la interfaz mostrada en la figura 25.

Cada uno de los nodos en Node-RED tiene propiedades que pueden ser configurados de acuerdo con las necesidades. Para la programación del sistema de monitoreo se utiliza tres secciones: panel de control, panel de alerta y detectores de humo que interactuaran entre sí. Cada uno de estos se programa de la siguiente manera:

Figura 25

Interfaz gráfica.

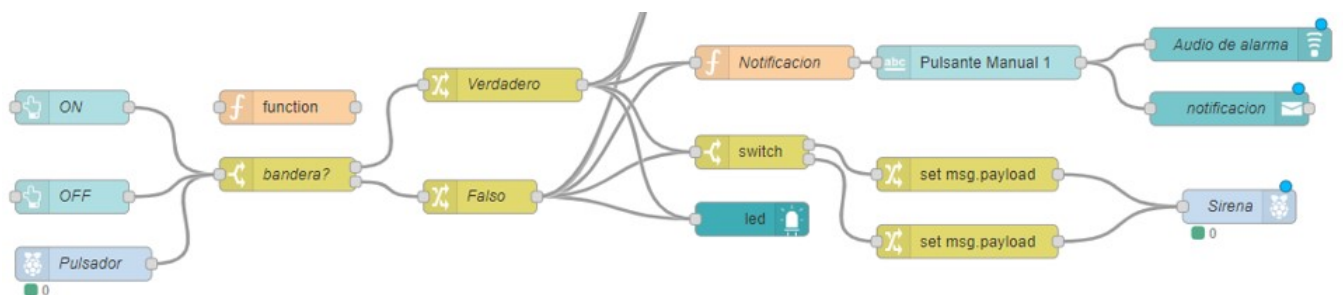


Nota: En la figura se puede observar la interfaz gráfica con la que podrá interactuar el usuario.

5.3.1. Programación del botón de emergencia físico y virtual

Figura 26

Nodos de programación de la estación manual y activación desde interfaz



Nota: La figura muestra los nodos que fueron programados y configurados.

En la figura 26 e puede ver las conexiones usadas para la programación para la activación de la alarma ya sea por el caso de la activación desde la interfaz con el nodo “GPIO in” o la activación con la estación manual. Ambos nodos se conectan a un nodo “switch” llamado bandera, el cual nos ayudará a determinar un valor de “true” o “false” según corresponda, es decir, si está activado o no.

Cuadro de textosEstos valores serán enviados a los nodos de verdadero y falso, que establecerán un valor de 1 o 0 respectivamente, permitiendo el paso a los nodos de “function”, “switch” y “led”. El nodo de “function” permite determinar el mensaje que se mostrará posteriormente en una ventana emergente, así como una notificación de audio sobre el estado del sistema de alarma, que se logra con el nodo de audio alarma y el nodo de notificación, el Listing 1 muestra la programación empleada en el nodo “function”.

```

1  if (msg.payload === true)
2  {
3    msg.payload = "Modulo de alarma de incendio activado "
4    return msg;
5  }
6  if (msg.payload === false)
7  {
8    msg.payload ="Modulo de alarma de incendio desactivado";
9    return msg; }

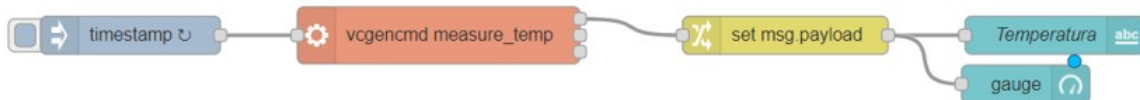
```

Listing 6: Configuración del nodo function

El nodo “switch” ayuda a determinar el estado que se ingresará en los nodos “GPIO out” llamados “Sirena” y “Luz”, activándose según el estado del sistema de alarma, la cual emitirá una señal de salida, por último, el nodo “led” se utiliza para visualizar el estado del sistema de alarma dentro de la interfaz, cambiando de color según el valor enviado por los nodos de verdadero y falso.

Figura 27

Nodos de programación para temperatura.



Nota: La figura muestra los nodos que fueron programados y configurados para visualizar la temperatura a la que se encuentra la unidad de procesamiento.

La figura 27 muestra el nodo “inject” que reiniciará el comando en la función cada 5 segundos para obtener el dato deseado en el intervalo establecido. Con el nodo “exec” podemos ejecutar un comando desde el cmd de la Raspberry Pi en Node-Red. El comando utilizado es “vcgencmd measure_temp”, que permite obtener la temperatura. Luego, se manipula el dato mediante un “msg.payload” para obtener solo el valor numérico, que será enviado al nodo “text” para su posterior visualización en la interfaz.

Figura 28

Dirección IP.



Nota: En la figura se visualiza los nodos usados para mostrar la dirección IP.

Para visualizar la dirección IP que posee la unidad de procesamiento se utiliza los nodos mostrados en la figura 28, visualizamos el dato de la dirección IP con el fin de que en caso de que se requiera alguna configuración ya sea en Node-Red o en la unidad de procesamiento el usuario tenga acceso a ella mediante la IP.

Utilizando del nodo “inject” enviará el dato en un tiempo de 0.1s después de iniciar el sistema de alarma, este está conectado el nodo “exec” utilizando el comando “hostname -I” el cual será visualizado en la interfaz mediante un nodo “text”.

5.3.2. Programación del sensor

Para el sensor, primero se toma los datos mediante un código en Python. Este código recibirá los datos por un puerto GPIO de la Raspberry, el cual se configura como puerto de entrada. Posteriormente se importa las bibliotecas “RPi.GPIO” para la configuración de los puertos GPIO de la Raspberry, “paho.mqtt.client” para la conexión como cliente a MQTT permitiendo enviar y recibir datos desde Node-RED y la biblioteca “time”.

Posteriormente, se establece la conexión entre Node-RED y Python con el comando “client.connect”, el cual será localhost debido a que el programa en Python y Node-RED están siendo ejecutados en la Raspberry. Finalmente, dentro de un bucle while se declara la variable

nivel_humo que toma el dato desde el puerto GPIO de entrada y lo envía como un dato de tipo “string”, dicho código se lo puede visualizar en el Listing 7:

```
1 import RPi.GPIO as GPIO
2 import time
3 import paho.mqtt.client as mqtt
4
5 # Configura el pin para el sensor de humo
6 GPIO.setmode(GPIO.BCM)
7 GPIO.setup(4, GPIO.IN)
8
9 # Se configura el cliente MQTT
10 client = mqtt.Client()
11 client.connect("localhost", 1883, 60)
12
13 # Lee los datos del sensor de humo cada 5 segundos y los publica en Node-
14 # RED
15 while True:
16     nivel_humo = GPIO.input(4)
17     print("Nivel de humo:", nivel_humo)
18     client.publish("sensorHumo", str(nivel_humo))
19     time.sleep(5)
```

Listing 7: Toma de datos del sensor

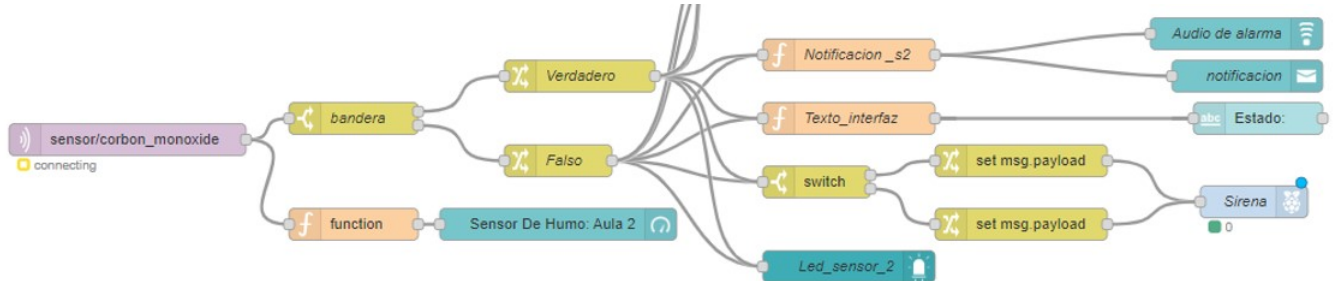
Una vez realizada la toma y envío de datos desde Python, se configura la recepción de datos en Node-RED mediante el nodo “MQTT in” llamado “sensor de humo”. Este nodo recibe los datos enviados desde Python junto con un nodo bandera y uno de “function”, los cuales manipulan el dato enviado. Con el nodo de “function” se establece rangos de trabajo en función del nivel de humo en partes por millón (ppm).

Si el valor es menor al rango establecido, se envía un valor “false” dentro del “msg.payload”, mientras que si el valor es mayor, se envía un valor “true”. Estos pasan al nodo “bandera”, el cual establece el valor de los nodos “switch”. Estos están conectados a dos nodos de “function”, uno establece el mensaje de detección para la ventana emergente de aviso y la notificación por audio dentro de la interfaz, mientras que el otro establece un mensaje del estado en el que se encuentra el sensor.

Del mismo modo con los nodos de “verdadero” y “falso” los cuales están conectados a un nodo “switch” activarán los nodos “GPIO-out”. Esto permite activar o desactivar los pines de salida que funcionarán como los pines de conexión para la luz estroboscópica y la sirena. Tal como se muestra en la figura 29.

Figura 29

Nodos para la visualización de datos del sensor.



Nota: En la figura se visualiza los nodos empleados para la manipulación de datos enviados por el sensor.

Para lograr una mayor precisión de los datos de partes por millón, el nodo “MQTT in” se conecta a un nodo de “function” que se encarga de tomar el dato enviado y convertirlo a un dato de tipo “.split”. Con ello, convierte los datos de tipo “string” en una cadena de caracteres, lo que permite visualizarlo en el nodo “gauge” la cantidad de ppm de humo.

Una vez que se hayan configurado y conectado los nodos según los requisitos, se debe presionar en el botón Deploy para implementar los cambios. Luego, se podrá visualizar e interactuar con la interfaz desarrollada y comprobar su funcionalidad.

5.4. Comprobación del funcionamiento del sistema de monitoreo de alarma contra incendios

Para llevar a cabo las pruebas de funcionamiento dentro de la Universidad, es necesario estar conectados a la misma subred a la que está conectada el centro de monitorio de la Universidad. Para conseguirlo, el departamento de sistemas debe asignar una dirección IP, así como la dirección de puerta de enlace y el servidor DNS vinculados al centro de monitoreo.

Figura 30

Configuración de dirección IP dentro del fichero DHCP.

```
GNU nano 2.7.4          Fichero: /etc/dhcpd.conf          Modificado
option classless_static_routes
# Most distributions have NTP support.
option ntp_servers
# Respect the network MTU. This is applied to DHCP routes.
option interface_mtu

# A ServerID is required by RFC2131.
require dhcp_server_identifier

# Generate Stable Private IPv6 Addresses instead of hardware based ones
slaac private

# Example static IP configuration:
#interface eth0
#static ip address=192.168.0.10/24
#static ip6 address=fd51:42f8:caae:d92e::ff/64
#static routers=192.168.0.1
#static domain_name_servers=192.168.0.1 8.8.8.8 fd51:42f8:caae:d92e::1

[ línea 46/64 (71%), col 1/1 (100%), car 1415/1873 (75%) ]
^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar txt ^J Justificar ^C Posición
^X Salir ^R Leer fich. ^\ Reemplazar ^U Pegar txt ^N Ortografía ^_ Ir a línea
```

Nota: En la figura se observa el fichero DHCP para configurar la dirección IP.

Con los datos de IP y servidor DNS, se configura la Raspberry ejecutando el comando “sudo nano /etc/dhcpd.conf”, que abre un fichero de configuración de direcciones IP estáticas. En el fichero, como se observa en la figura 30 se idéntica las líneas correspondientes a “#interface eth0” y se lo edita para añadir las nuevas direcciones. De esta forma, se asegura la conexión a la red y se podrá realizar las pruebas necesarias. Una vez que se haya terminado de escribir las nuevas direcciones IP, pulsamos se guarda los cambios para que se configure correctamente la Raspberry.

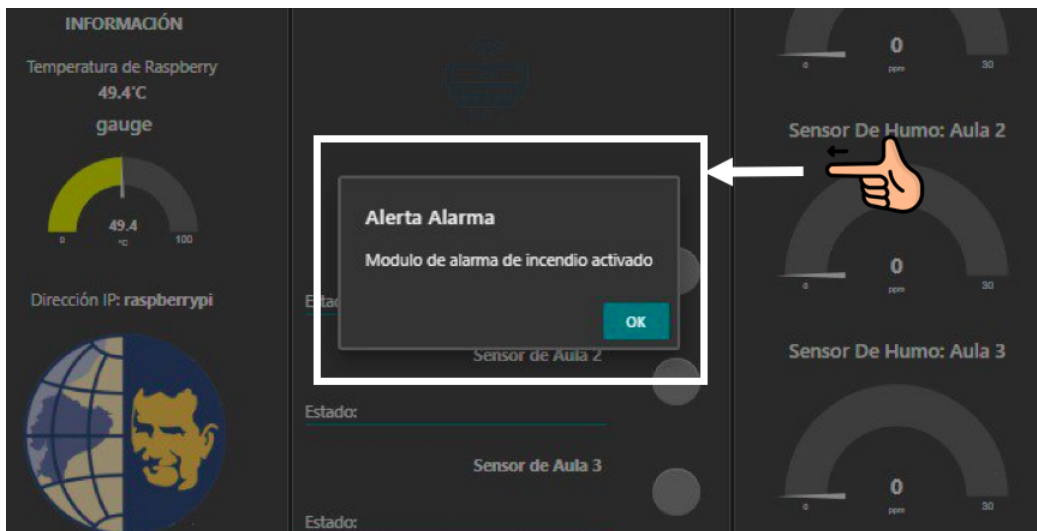
Una vez conectada a la subred del centro de monitoreo se podrá acceder a la interfaz para realizar las pruebas de funcionamiento. Para las pruebas el módulo de alarma contra incendios se cuenta con un sensor de humo on/off, un switcho que actuara como la palanca de emergencia y para la salida de la sirena estroboscópica se aplicara dispositivos de notificación visual para corroborar que de todos los periféricos de entradas como salidas programadas funcionen correctamente.

Las pruebas se realizarán en dos etapas, la primera mediante la activación del sistema de alarma por el botón de emergencia de la propia interfaz y activación por la estación manual direccionada por un pin GPIO, la segunda etapa de prueba se realiza por medio de la activación del sistema de alarma notificada por los sensores de humo. En la primera prueba la notificación de activación de alarma se presenta mediante una ventada emergente centrada

en la interfaz y la notificación por audio indicará que el sistema de alarma esta activado, de igual manera en la sección de panel de alerta el led indicador del estado del sistema de alarma cambiará a verde para cuando este activado o rojo cuando este desactivado, en la figura 31 se puede observar la interfaz cuando el sistema de alarma esta activado.

Figura 31

Activación del sistema de alarma por estación manual/botón de interfaz.

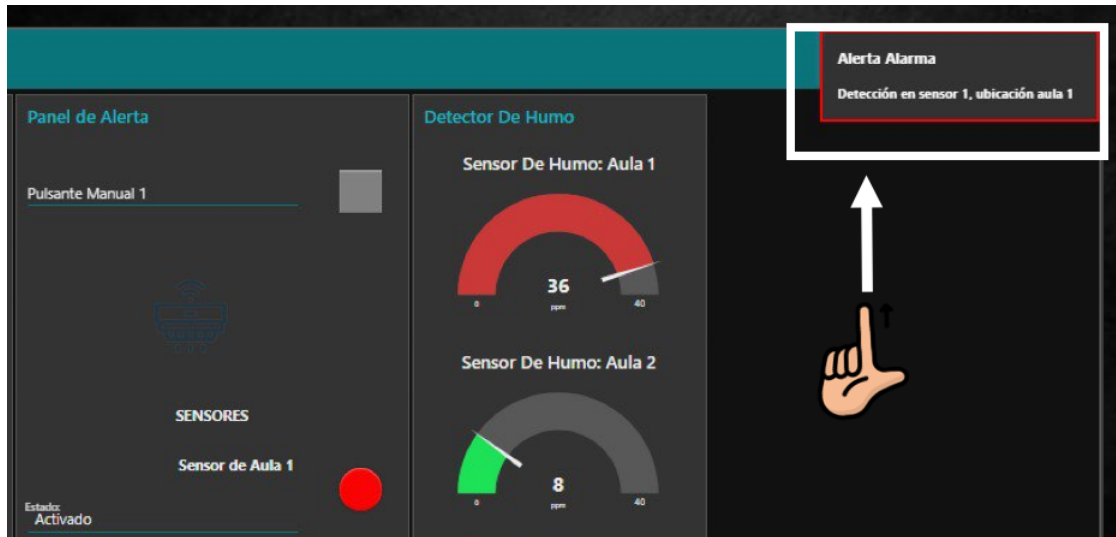


Nota: En la figura podemos observar la notificación en la interfaz cuando el sistema de alarma es activado.

La siguiente prueba muestra las notificaciones de una ventana de emergencia y de audio que indican en este caso la activación de los sensores de humo de distintas ubicaciones. La ventana de emergencia se muestra en la parte superior derecha con el fin de que el personal de monitoreo pueda observar en la sección de panel de alerta el led indicador del estado del sensor, durante las pruebas se usaron 3 sensores cada uno simulando una ubicación diferente, tal como se muestra en la figura 32.

Figura 32

Activación automática del módulo mediante los sensores.



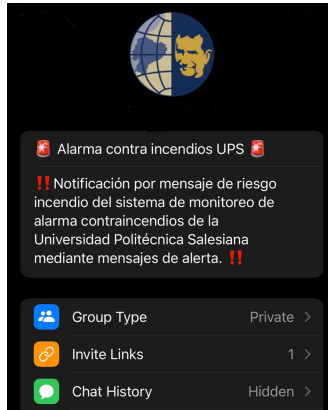
Nota: En la figura podemos observar la notificación en la interfaz junto con el led indicador de estado del sensor.

Adicionalmente en caso de que el personal de monitoreo no se encuentre al momento de que el sistema de alarma se active, ya sea por la estación manual o por los sensores se enviará un mensaje de alerta por la aplicación de mensajería móvil y de escritorio basada en la nube, Telegram. Para ello dentro de Telegram creamos un canal de chats en cual el personal de monitoreo pueda ingresar e incluso invitar a demás miembros de la comunidad universitaria para estar siempre alerta en caso de emergencia, en la figura 33 podemos visualizar el grupo de Telegram.

Dentro de este canal se puede programar un bot el cual se encarga de emitir los mensajes que sean enviados desde Node-RED, en la figura 34 podemos observar el mensaje que se es emitido en caso de que el sistema de alarma sea activado.

Figura 33

Notificación de alerta en Telegram.



Nota: En la figura se muestra el grupo mediante el cual se recibe la notificación de alerta de incendio.

Figura 34

Mensaje recibido a Telegram.



Nota: En la figura podemos observar el mensaje recibido en Telegram cuando el módulo esta activado.

Para emitir el mensaje en Telegram desde Node-RED se usará los siguientes nodos: el nodo “function” llamado “Conexión telegram” que establecerá el mensaje a enviar en el “msg.payload” dependiendo del estado en el que se encuentra el sistema de alarma. Para ello implementamos el código que se puede visualizar en el Listing 8.

```
1 // Notificacion en telegram
2 var message = '*ACTIVACION DE PALANCA DEL MODULO DE ALARMA CONTRA INCENDIOS
   :*\r\n'+
3 '  ATENCION  !! Se ha activado la palanca de alarma de incendios';
4 var messageoff = '*ACTIVACION DE PALANCA DEL MODULO DE ALARMA CONTRA
   INCENDIOS:*\r\n' +
5 '  Alarma de incendio desactivada';
6 if (msg.payload === true)
7 {
8   msg.payload = { chatId: -915966851, type: 'message', content:message};
9   msg.payload.options={disable_web_page_preview: true, parse_mode:"
   Markdown"};
10  return msg;
11 }
12 if (msg.payload === false)
13 {
14   msg.payload = { chatId: -915966851, type: 'message', content:
   messageoff };
15   msg.payload.options = { disable_web_page_preview: true, parse_mode: "
   Markdown" };
16  return msg;
17 }
```

Listing 8: Notificación por mensaje de alerta en Telegram

Por otra parte, con el nodo “function” llamado “Not_img” enviamos junto al mensaje de notificación una imagen que indica que el sistema de alarma se activó, para ello dentro del nodo “function” se establece un código para determinar el estado del “msg.payload” dependiendo el estado del sistema de alarma, emitiendo la imagen de encendido o apagado.

Dentro del código se guarda la imagen que se enviara por el canal de Telegram, este código lo podemos visualizar en el Listing 9, la cual se envía un archivo de animación cuando el sistema de alarma sea activado y de igual manera en cuanto ala desactivación de la alarma de incendio. Las animaciones y notificaciones se seguirán enviando sin la necesidad de tener abierto la interfaz web de usuario. De esta manera, aquellos que estén dentro del grupo de Telegram pueden visualizar el estado del sistema de alarma en tiempo real.

```

1 // Notificacion en telegram
2 if (msg.payload === true)
3 {
4     msg.payload = {
5         chatId: -915966851,
6         type: 'animation',
7         content: "https://external-content.duckduckgo.com/iu/?u=https%3A%2F%2F
8         Fdsimexico.com%2Fwp-content%2Fuploads%2F2020%2F07%2F
9         F164844_6318d88c88b54f51949923205a16e718_mv2.gif&f=1&nofb=1&ipt=1231472
10        d5e01945923523169c084bf498eb66434b6af703c501b6fe1d200f06c&ipo=images"
11    };
12
13 return msg;
14 }
15 if (msg.payload === false)
16 {
17     msg.payload = {
18         chatId: -915966851,
19         type: 'photo',
20         content: "https://img.huffingtonpost.com/asset/5
21         c8a778c2500000d04c913d5.jpeg?ops=scalefit_720_noupscale" };
22 return msg;
23 }

```

Listing 9: Notificación por imagen en Telegram

Una vez establecido los nodos “function” los conectamos a los nodos de “verdadero” y “falso” que observamos en la figura 26 y 29 con el fin de lograr que el mensaje emitido al Telegram sea de manera automática al momento de activación del sistema de alarma. Con el nodo “Telegram sender” establecemos la conexión de Node-RED y Telegram el cual enviara los mensajes hacia un canal de Telegram previamente configurado junto con un bot el cual es el encargado de emitir los mensajes. La conexión de los nodos la podemos ver en la figura 35.

Figura 35

Nodos para la conexión Node-RED y Telegram.



Nota: En la figura podemos observar los nodos empleados para la conexión de Node-RED y Telegram.

De igual manera dentro de Telegram se recibirá una alerta ante una desconexión de algún dispositivo que integra el módulo. Para ello se estableció un código en un nodo “function” el cual emitirá una señal diferente de “true” o “false” cuando ocurra una desconexión de alguno de los componentes, este código se encarga de leer el valor dentro del “msg.payload” y cuando ocurra un cambio enviará un mensaje a Telegram, el código puede ser visualizado en el Listing 10.

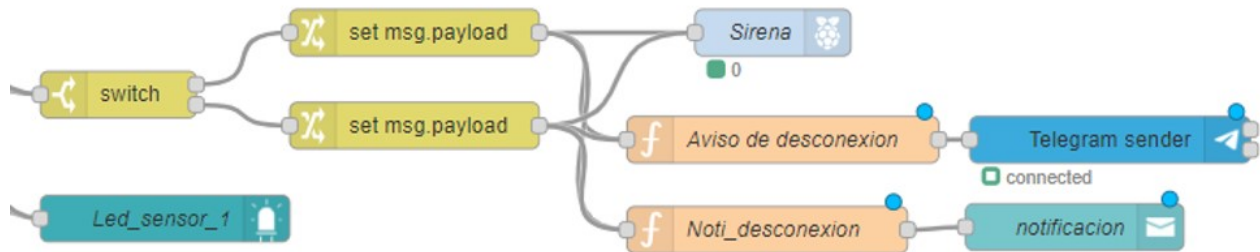
```
1 // Notificacion en telegram
2 var messages_s1 = '*DESCONEXION:*\r\n' +
3   'Sensor 1 desconectado';
4
5 if (msg.payload !== true && msg.payload !== false) {
6
7   msg.payload = { chatId: -915966851, type: 'message', content:
8     messages_s1 };
9   msg.payload.options = { disable_web_page_preview: true, parse_mode: "
10  Markdown" };
11
12   return msg;
13 }
14 return null;
```

Listing 10: Notificación de desconexión de un dispositivo

Una vez configurado los nodos “function” llamados “Aviso de desconexión” y “Noti_desconexion” son conectados a los nodos “switch” los cuales se encargarán de determinar el valor en el “msg.payload” y enviar el mensaje con el nodo “Telegram sender” como se muestra en la figura 36.

Figura 36

Nodos empleados para la notificación en caso de desconexión.



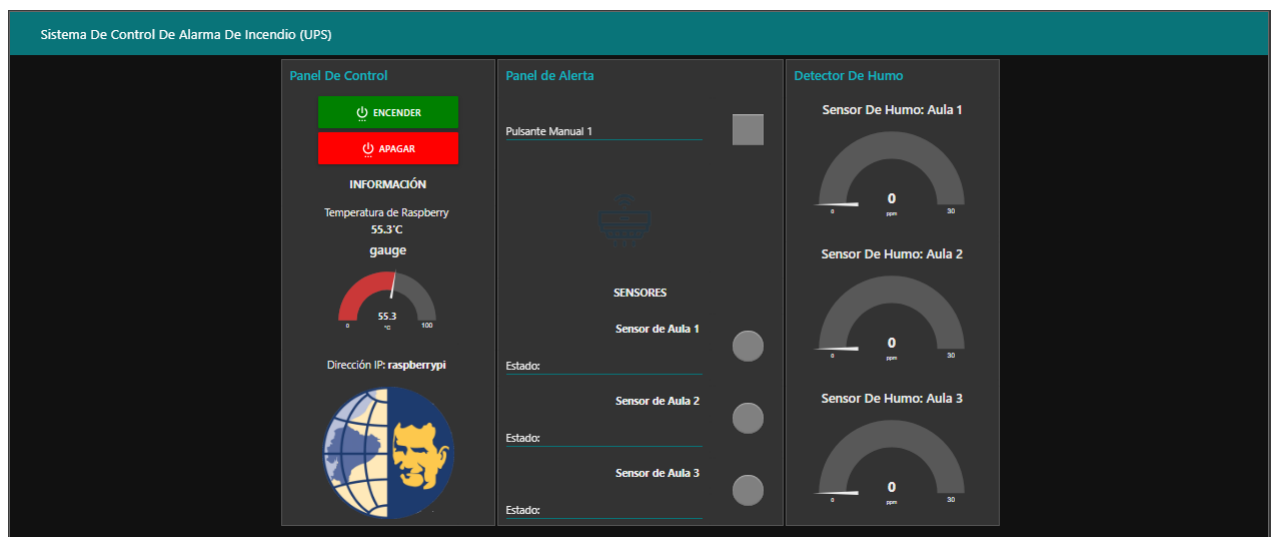
Nota: En la figura podemos observar los nodos empleados para la notificación en caso de desconexión.

6. Resultados

Las pruebas realizadas del sistema de monitoreo permitieron comprobar el funcionamiento de la parte eléctrica, electrónica y programación de la unidad de control. La figura 37 muestra la interfaz usuario máquina implementada en el centro de monitoreo de la Universidad, para la visualización, notificación y control del sistema de alarma.

Figura 37

Interfaz usuario-máquina.



Nota: En la figura podemos observar la interfaz usuario-máquina implementada en el centro de monitoreo de la Universidad.

La notificación emitida en la interfaz usuario-máquina cuando el sistema de alarma se activa ya sea desde la estación manual/botón de la propia interfaz o por los sensores se muestra en la figura 38. La alerta se realiza mediante una ventana emergente de notificación en la parte central de la interfaz cuando es manual y en parte superior derecha cuando es automática, con el objetivo de que el personal de monitoreo visualice cuál de los sensores se ha activado.

Puesto a que, el sistema de alarma permite la conexión de uno o más sensores de humo a la unidad de control, el panel de alerta muestra por separado cada sensor y led indicador correspondiente, del mismo modo la notificación por la ventana emergente y la notificación de audio que indica cuál de los sensores se ha activado.

Figura 38

Notificación emitida en la interfaz.



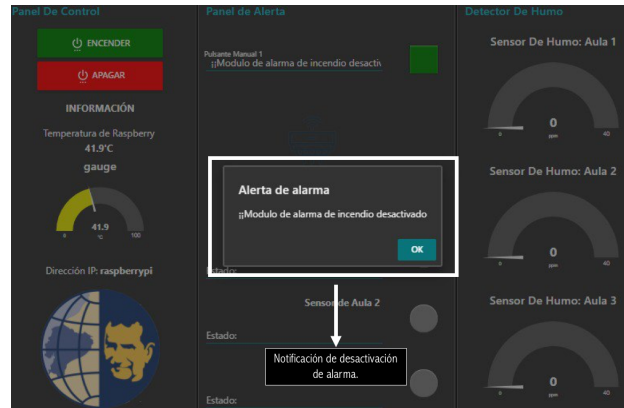
Nota: La figura muestra la notificación emitida en la interfaz cuando el sistema de alarma se ha activado.

Del mismo modo se consiguió notificar mediante una ventana emergente, la desactivación del sistema de alarma por medio de la estación manual o a su vez el caso de que se intervenga desde la interfaz usuario, tal como se muestra en la figura 39. En el panel de alerta se encuentra el led indicador y mensaje de estado del sistema de alarma mientras que en detectores de humo se visualiza los datos enviados por el sensor en caso de que la notificación por la ventana emergente haya sido cerrada.

Esto con el fin de que el personal de monitoreo posea un mayor conocimiento del estado en el que se encuentra el sistema de alarma evitando falsas alarmas debido a que el sistema de alarma puede ser accionado desde la propia interfaz.

Figura 39

Desactivación de alarma.

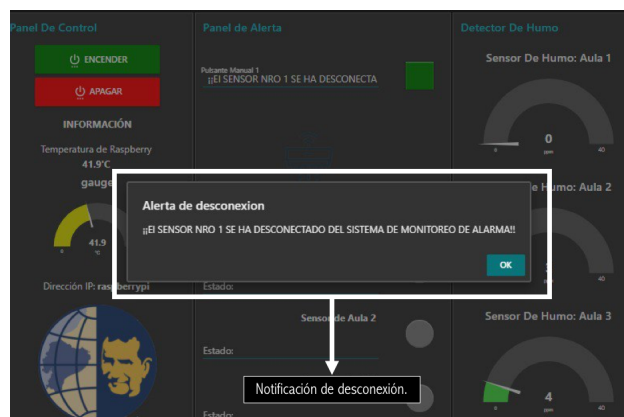


Nota: El gráfico corresponde a la visualización de notificación de desactivación de alarma.

Adicionalmente mediante una misma ventana emergente se logró notificar la desconexión de los dispositivos de inicio; el aviso se realiza para cada uno de los sensores por separado, tal como se muestra en la figura 40, que representa un caso real de desconexión del sensor 1 del sistema de alarma.

Figura 40

Notificación de desconexión de dispositivos

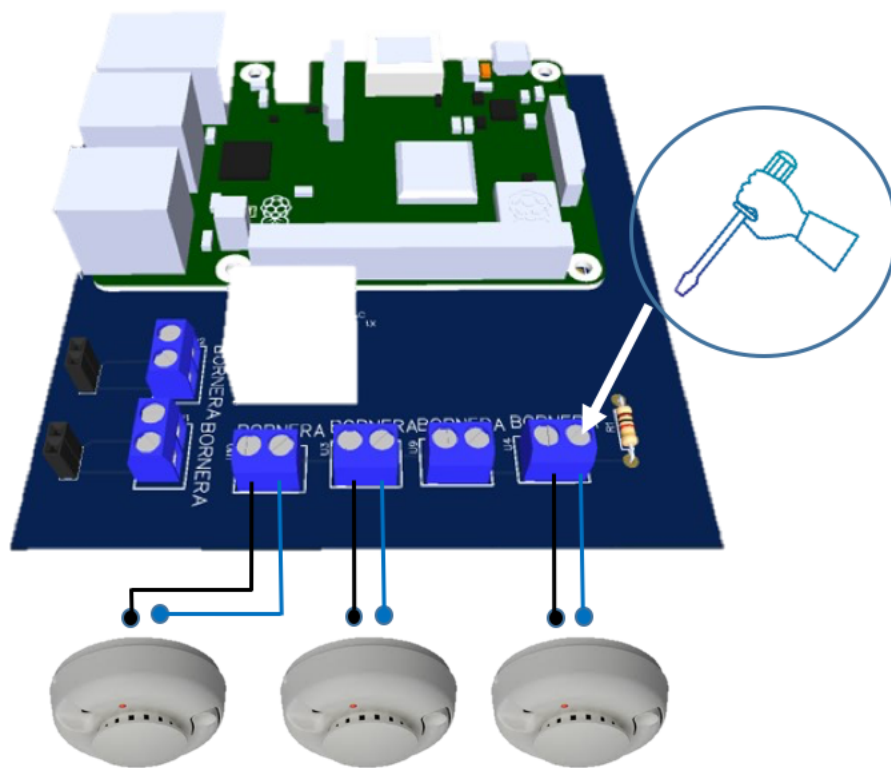


Nota: En la figura se muestra la ventana emergente de notificación en caso de desconexión de un dispositivo.

De acuerdo con el diseño utilizado en la unidad de control, el sistema de alarma permite la ampliación de más sensores sin la necesidad de realizar cambios significativos en su estructura. La conexión de los nuevos sensores requiere del desajuste y ajuste de los bornes correspondiente para realizar las conexiones, lo que se representa en la imagen a continuación como se muestra en la figura 41.

Figura 41

Ajuste y desajuste de bornes.

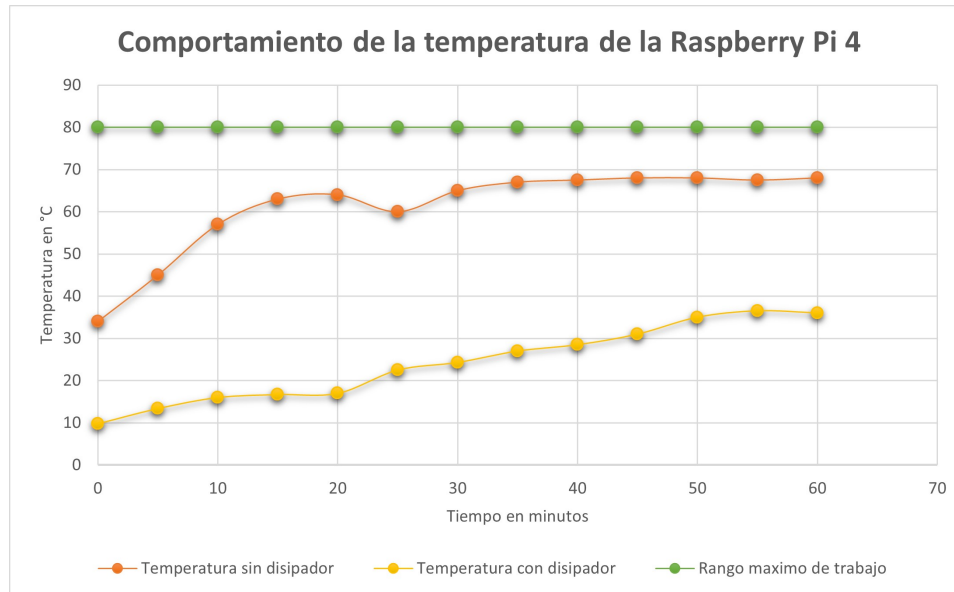


Nota: El gráfico representa el ajuste y desajuste que se puede realizar para la conexión de varios sensores.

La placa Raspberry que corresponde a la unidad de procesamiento, está dentro de la unidad de control, razón por la cual se realizó una simulación térmica de la unidad de procesamiento para dimensionar el disipador activo, es decir, el ventilador necesario que se debe ocupar para la correcta disipación de calor que emita, evitando fallos en la unidad de procesamiento generado por un sobrecalentamiento.

Figura 42

Comportamiento térmico de la unidad de procesamiento.



Nota: En la figura se observa el comportamiento térmico de la unidad de procesamiento con y sin disipación.

En la figura 42 se puede observar el comportamiento térmico que tendrá la unidad de procesamiento obtenido a partir de la simulación realizada en un periodo de una hora junto con el rango máximo de trabajo que es de 80°C de acuerdo con las especificaciones proporcionadas por Raspberry, puesto a que el sistema de alarma debe funcionar 24 horas al día, la temperatura en la unidad de procesamiento debe estar dentro de un rango aceptable, dado a que una temperatura cerca al rango máximo de trabajo podría llegar a ocasionar fallos en el sistema de alarma.

Los valores de temperatura de la unidad de procesamiento sin disipación se encuentran oscilando entre los 65 y 70°C, mientras que la temperatura de la unidad de procesamiento con disipación no supero los 40°C, por lo que, el uso de un disipador ayuda a mantener la temperatura dentro de un rango aceptable para que el sistema de alarma funcione las 24 horas del día sin contratiempos.

El sistema de monitoreo permite al personal conocer el estado en el que se encuentra el sistema de alarma fuera de la central de monitoreo de la Universidad, dado que se emite un mensaje de alerta cuando ocurre un cambio en el sistema de alarma por el canal de chat usado en Telegram que se notifica independientemente de si la interfaz usuario máquina está siendo ejecutada. Las notificaciones enviadas se muestran en las figuras 43 y 44

Figura 43

Notificación de alerta por Telegram.



Nota: En la figura se visualiza la notificación de alerta por medio de Telegram.

Figura 44

Notificación automática de desconexión por Telegram.



Nota: En la figura se muestra la notificación cuando se activa la alarma de forma automática y también la notificación por desconexión en Telegram.

Dentro del canal en Telegram, el personal de monitoreo puede incluir más miembros de la comunidad universitaria, este canal que trabaja con el bot que previamente se ha configurado, se explica en la sección 5.4, es el encargado de enviar el mensaje del estado del sistema de monitoreo, lo que permite tomar acciones de manera inmediata ante un caso de emergencia.

7. Conclusiones

- Los dispositivos tecnológicos que se implementaron para el sistema de monitorio de alarma constan de dispositivos de inicio, tales como, el detector de humo fotoeléctrico i3 4AW-B de cuatro cables, que trabaja con 12/24 V, la estación manual de activación serie BG-12; un dispositivo de notificación, como la sirena estroboscópica SH-816S-SQ y un dispositivo de procesamiento de datos Raspberry Pi 4B fueron programadas bajo la guía de la normativa NFPA, durante toda la etapa diseño y simulación del sistema de alarma contra incendios.
- En el diseño del sistema de alarma se considero el código 10.6.7.3 de la NFPA en referencia a las fuentes de suministro de energía, en donde menciona la necesidad de dos fuentes de energía, una que se conecta directamente a la red eléctrica y una segunda fuente para asegurar el funcionamiento en caso de fallas del primero, por tal motivo se realizo un análisis de consumo energético de todos los dispositivos para determinar el amperaje hora necesario para su funcionamiento, que arrojo como resultado que el sistema de alarma requiere de una batería DC de 12v y 23,9 Ah como fuente secundaria.
- Para la unidad de control se realizó un diseño de PCB el cual de acuerdo a los resultados obtenidos requiere un ancho de pista de al menos 6 milésimas de pulgada para asegurar el paso de la corriente máxima que consume la unidad de procesamiento, los sensores y la sirena estroboscópica.
- En la unidad de procesamiento Raspberry se considero el uso de un ventilador, que garantice mantener la temperatura dentro del rango de trabajo permitido, que corresponde a valores de temperatura inferiores a los 40°C, evitando fallos por sobrecalentamiento o “thermal throttling” que es un sistema de seguridad utilizado por los procesadores para reducir su rendimiento ante altas temperaturas; evitando de ese modo la falla en el rendimiento de la unidad de procesamiento que afecta al procesamiento de datos y funcionamiento de la interfaz de usuario.

- Con la programación del software en Node-RED y el lenguaje de programación Python se creó la interfaz usuario-máquina que permite al centro de monitoreo visualizar e interactuar con los datos en tiempo real de los dispositivos de inicio y activación de los dispositivos de notificación.
- Las pruebas de funcionamiento del módulo se realizaron con el centro de monitoreo, para evidenciar el funcionamiento de la interfaz implementada en el sistema de alarma contra incendios, lo que permitió visualizar al personal de monitoreo el estado en el que se encuentra el sistema de alarma.
- Las pruebas de funcionamiento permitieron visualizar el sistema de notificación por texto y audio dentro de la interfaz y por mensaje en Telegram ante la activación del sistema de alarma de manera manual o automática, las notificaciones que se emiten son de manera inmediata para que se tenga una respuesta rápida ante un caso de emergencia.

8. Recomendaciones

- Revisar el código nacional de alarmas de incendio y señalización NFPA-72 para el diseño, instalación, inspección, prueba y mantenimiento de sistemas de alarma contra incendios.
- Debido a que la Raspberry usa una tarjeta microSD para almacenar el sistema operativo, archivos de programa y ciertas configuraciones necesarias para el funcionamiento del módulo, se debería realizar copias de seguridad de estas en caso de que la microSD falle por factores externos como el calor generado por la propia Raspberry o el fin de ciclo de vida útil de la microSD.
- Considerando que la Raspberry posee un número limitado de pines para configurar como entradas y salidas y teniendo en cuenta la adaptabilidad que posee esta placa de programación, se propone incorporar un módulo expensor de pines GPIO para que se pueda agregar más sensores, así por consiguiente se expande las áreas cubiertas por el sistema. Aumentando significativamente la eficacia, precisión y a su vez reducir el coste de adquisición de más módulos.
- La placa Raspberry es muy versátil por lo que se asemeja a una computadora de tamaño compacto, esta placa de desarrollo ofrece la posibilidad de incorporar una amplia gama de periféricos y componentes de hardware tales como, monitor, pantalla touch, parlantes, teclados y cámaras web. Este último se podría agregar en el funcionamiento del sistema

de monitoreo otorgando al personal la visibilidad de áreas de gran importancia, del mismo modo se podría enviar las imágenes de vídeo mediante Telegram de manera inmediata.

- Con el análisis térmico se opto por dejar la unidad de procesamiento a la intemperie, sin embargo, es importante tener en cuenta que dejar la unidad fuera de un gabinete puede exponerla a distintos riesgos , como daños físicos, exposición ante elementos ambientales, interferencias o riesgos eléctricos. Por lo tanto, se debería realizar un análisis del comportamiento térmico de la unidad de procesamiento dentro de un gabinete eléctrico o una caja de protección bajo normativa, pero que a su vez permita un flujo de aire para mantener a la unidad de procesamiento dentro de los rangos de trabajo de temperatura establecido dentro de los códigos establecidos por la NFPA.
- Actualmente el módulo esta diseñado para una implementación dentro de la Universidad Politécnica Salesiana, pero se podría aplicar a cualquier tipo de edificación o incluso hogares, realizando ciertas reconfiguraciones dentro de la placa que se adapten al nuevo entorno.
- Actualmente se diseño e implemento el sistema de monitoreo, para una futura implementación del sistema de alarma se debería utilizar los sensores System 4W-B de 4 hilos, la sirena estroboscópica SH-816-SQ y la estación manual BG-12 que fueron objeto de estudio en el presente trabajo, esto con el fin de garantizar que durante las pruebas del sistema de alarma los datos obtenidos sean más precisos ya que este tipos de sensores proporcionan una mayor capacidad para detectar la concentración de humo y la capacidad de distinguir un incendio y otras fuentes de humo.

Referencias

- Autodesk. (2023). *Fusion 360 | Software CAD, CAM, CAE y de circuitos impresos 3D basado en la nube | Autodesk*. Descargado 2023-07-05, de <https://www.autodesk.mx/products/fusion-360/overview>
- Chugá Meneses, O. G. (2019). *Implementación de un sistema de alarma para detección incendios, en el edificio de la carrera de ingeniería en mantenimiento eléctrico en el campus universitario el olivo* (B.S. thesis).
- Delgado Huanca, O. B. (2021). Propuesta de implementación de sistema de alarmas contra incendio en la empresa js market solutions sac.
- Enforcer. (2020, agosto). SECO-LARM ENFORCER SH-816S-SQ SERIES MANUAL [Manual de software informático].
- firesystem.ec. (2021, marzo). *¿Cuáles son los dispositivos de iniciación?* Descargado 2022-12-16, de <https://www.firesystemec.com/deteccion/dispositivos-de-iniciacion/>
- Google. (2023). *Salesian Polytechnic University · C. Vieja 12-30 y, Cuenca 010105*. Descargado 2023-07-26, de <https://www.google.com/maps/place/Salesian+Polytechnic+University/@-2.8865291,-78.9898791,15z/data=!4m6!3m5!1s0x91cd1826d90c7e47:0x8eb47b6b0138cb74!8m2!3d-2.8865291!4d-78.9898791!16s%2Fm%2F0cpdx8q?entry=ttu>
- Hale, M. (2017). *Introduction to the Internet of Things (IoT), Node Red, and the Raspberry Pi · nebraska-gencyber-modules*.
- He, X., Feng, Y., Xu, F., Chen, F.-F., y Yu, Y. (2022, diciembre). Smart fire alarm systems for rapid early fire warning: Advances and challenges. *Chemical Engineering Journal*, 450, 137927. Descargado 2023-01-06, de <https://www.sciencedirect.com/science/article/pii/S1385894722034131> doi: 10.1016/j.cej.2022.137927
- Inselec. (2023). *GABINETE LIVIANO 300x300x200 (Ref: I-0303)*. Descargado 2023-07-26, de <https://inselec.com.ec/store/inicio/2018-gabinete-liviano-300x300x200-ref-i-0303-.html>
- Jee, S.-W., Lee, C.-H., Kim, S.-K., Lee, J.-J., y Kim, P.-Y. (2014, mayo). Development of a Traceable Fire Alarm System Based on the Conventional Fire Alarm System. *Fire Technology*, 50(3), 805–822. Descargado de <https://doi.org/10.1007/s10694-012-0299-0> doi: 10.1007/s10694-012-0299-0
- Juarez, L. (2023). *¿Cómo funciona el monitoreo de alarmas?* Descargado 2022-12-02, de <https://es.linkedin.com/pulse/c%C3%B3mo-funciona-el-monitoreo-de>

-alarmas-luis-juarez

- Lekić, M., y Gardašević, G. (2018, marzo). IoT sensor integration to Node-RED platform. En (pp. 1–5). IEEE. Descargado 2022-11-29, de <https://bibliotecas.ups.edu.ec:2095/xpl/conhome/8337878/proceeding>
- Mahoney, S. (2021). *Una guía sobre los conceptos básicos de las alarmas contra incendios - Iniciación*. Descargado 2023-02-01, de <https://www.nfpajla.org/blog/1882-una-guia-sobre-los-conceptos-basicos-de-las-alarmas-contra-incendios-iniciacion>
- Mendieta, A. (2022, abril). *Los 3 tipos de notificación para un sistema de detección de incendios*. Descargado 2022-12-02, de <https://shingenieria.com/3-tipos-de-notificacion-sistema-de-deteccion/>
- Nagar, S. (2018). *Python*. Cham: Springer International Publishing. Descargado 2023-01-05, de https://doi.org/10.1007/978-3-319-63962-8_269-1 doi: 10.1007/978-3-319-63962-8_269-1
- NFPA 72: *Código Nacional de Alarmas y Señalización contra Incendios*. (2016). Descargado 2023-01-24, de <https://www.nfpa.org/codes-and-standards/all-codes-and-standards/list-of-codes-and-standards/detail?code=72>
- NFPA JLA. (2023). Descargado 2023-01-05, de <https://nfpajla.org/>
- Pull Stations \textbar Manual Initiating Devices*. (2023). Descargado 2023-02-01, de <https://buildings.honeywell.com/us/en/products/by-category/fire-life-safety/manual-call-points-pull-stations-and-panic-buttons/manual-call-points-pull-stations/bg-12-series-fire-alarm-pull-station>
- Python*. (2021, mayo). Descargado 2022-12-16, de <https://www.mytaskpanel.com/python-y-sus-casos-de-uso-todo-lo-que-necesitas-saber/>
- Raspberry. (2023). *Ventilador Original para Raspberry Pi 4 con Disipador - Raspberry Pi*. Descargado 2023-07-26, de <https://raspberrypi.cl/producto/ventilador-original-para-raspberry-pi-4-con-disipador/>
- Raspberry, P. (2021, enero). Raspberry pi 4 computer model B [Manual de software informático].
- Salcedo, M., y Cendrós, J. (2016). Uso del minicomputador de bajo costo “raspberry pi” en estaciones meteorológicas. *Télématique*, 15(1), 62–84.
- Song, C. (2022). *AUV/ROV/HOV Control Systems*. Singapore: Springer Nature. Descargado 2023-01-05, de https://doi.org/10.1007/978-981-10-6946-8_282 doi: 10.1007/978-981-10-6946-8_282
- Steren, E. (2022, diciembre). *Batería sellada de ácido-plomo, 12 Vcc 24 Ah Steren Tie*. Descargado 2023-07-26, de <https://www.steren.com.mx/bateria-sellada-de-acido>

-plomo-12-vcc-24-ah.html

Systemsensor. (2023). Detector fotoeléctrico de humo: INSTRUCCIONES DE INSTALACIÓN Y MANTENIMIENTO [Manual de software informático].

Turmo, E. (2022). NTP 215: Detectores de humos. Descargado 2022-01-12, de https://www.insst.es/documents/94886/326853/ntp_215.pdf/78d02563-f887-4ed3-a3cf-371c1a35c0f5?version=1.0&t=1614698422539

Upton, E. (2019, junio). *Raspberry Pi 4 on sale now from \$35*. Descargado 2022-12-16, de <https://www.raspberrypi.com/news/raspberry-pi-4-on-sale-now-from-35/>

Velasco Núñez, H. J. (2015). Sistema automático de detección de incendio según normas nfpa 72 mediante comunicación de doble vínculo a una estación de monitoreo.

¿Qué es Raspberry Pi y para qué sirve? (2021, noviembre). Descargado 2022-11-23, de <https://www.spacetechnics.com/que-es-raspberry-pi-y-para-que-sirve/>

ANEXOS

Anexo A: Código utilizado en Proteus para la simulación del sistema de alarma

```
1 # !/usr/bin/env python3
2 #!/usr/bin/env python3
3 # Main.py file generated by New Project wizard
4 # Created:   Mon May 29 2023
5 # Processor: RPI4
6 # Compiler:  Python 3 (Proteus)
7 from goto import *
8 import time
9 import RPi.GPIO as GPIO
10 #import serial
11 GPIO.setmode(GPIO.BOARD)
12 GPIO.setwarnings(False)
13
14 '''
15 define pin for lcd
16 '''
17 # Timing constants
18 E_PULSE = 0.0005
19 E_DELAY = 0.0005
20 delay = 1
21 buzzer=37
22 GPIO.setup(buzzer, GPIO.OUT)
23
24 # Pines de LCD
25 LCD_RS = 7
26 LCD_E  = 11
27 LCD_D4 = 12
28 LCD_D5 = 13
29 LCD_D6 = 15
30 LCD_D7 = 16
31 gas_Sensor = 18
32 red_light = 31
33
34 Buzzer= 29
35 GPIO.setup(LCD_E, GPIO.OUT) # E
36 GPIO.setup(LCD_RS, GPIO.OUT) # RS
37 GPIO.setup(LCD_D4, GPIO.OUT) # DB4
38 GPIO.setup(LCD_D5, GPIO.OUT) # DB5
```

```

39 GPIO.setup(LCD_D6, GPIO.OUT) # DB6
40
41 GPIO.setup(LCD_D7, GPIO.OUT) # DB7
42 GPIO.setup(gas_Sensor, GPIO.IN) # DB7
43 GPIO.setup(red_light, GPIO.OUT)
44 GPIO.setup(Buzzer, GPIO.OUT)
45 # Constantes
46 LCD_WIDTH = 16
47 LCD_CHR = True
48 LCD_CMD = False
49 LCD_LINE_1 = 0x80 # LCD RAM address-1st line
50 LCD_LINE_2 = 0xC0 # LCD RAM address-2nd line
51
52 '''
53 Function Name :lcd_init()
54 Function Description : this function is used to initialized lcd by sending
    the different commands
55 '''
56 def lcd_init():
57     # Inicializar display
58     lcd_byte(0x33,LCD_CMD) # 110011 Inicializar
59     lcd_byte(0x32,LCD_CMD) # 110010 Inicializar
60     lcd_byte(0x06,LCD_CMD) # 000110 Direccion de cursor
61     lcd_byte(0x0C,LCD_CMD) # 001100 Display On,Cursor Off, Blink Off
62     lcd_byte(0x28,LCD_CMD) # 101000 Data length
63     lcd_byte(0x01,LCD_CMD) # 000001 Clear display
64     time.sleep(E_DELAY)
65 '''
66 Function Name :lcd_byte(bits ,mode)
67 Fuction Name :the main purpose of this function to convert the byte data
    into bit and send to lcd port
68 '''
69 def lcd_byte(bits, mode):
70     # Send byte to data pins
71     # bits = data
72     # mode = True for character
73     #         False for command
74
75     GPIO.output(LCD_RS, mode) # RS
76
77     # High bits
78     GPIO.output(LCD_D4, False)

```

```

79  GPIO.output(LCD_D5, False)
80  GPIO.output(LCD_D6, False)
81  GPIO.output(LCD_D7, False)
82  if bits&0x10==0x10:
83      GPIO.output(LCD_D4, True)
84  if bits&0x20==0x20:
85      GPIO.output(LCD_D5, True)
86  if bits&0x40==0x40:
87      GPIO.output(LCD_D6, True)
88  if bits&0x80==0x80:
89      GPIO.output(LCD_D7, True)
90
91  # Toggle 'Enable' pin
92  lcd_toggle_enable()
93
94  # Low bits
95  GPIO.output(LCD_D4, False)
96  GPIO.output(LCD_D5, False)
97  GPIO.output(LCD_D6, False)
98  GPIO.output(LCD_D7, False)
99  if bits&0x01==0x01:
100     GPIO.output(LCD_D4, True)
101  if bits&0x02==0x02:
102     GPIO.output(LCD_D5, True)
103  if bits&0x04==0x04:
104     GPIO.output(LCD_D6, True)
105  if bits&0x08==0x08:
106     GPIO.output(LCD_D7, True)
107
108  # Toggle 'Enable' pin
109  lcd_toggle_enable()
110  '''
111  Function Name : lcd_toggle_enable()
112  Function Description: basically this is used to toggle Enable pin
113  '''
114  def lcd_toggle_enable():
115      # Toggle enable
116      time.sleep(E_DELAY)
117      GPIO.output(LCD_E, True)
118      time.sleep(E_PULSE)
119      GPIO.output(LCD_E, False)
120      time.sleep(E_DELAY)

```



```

121 '''
122 Function Name :lcd_string(message,line)
123 Function Description :print the data on lcd
124 '''
125 def lcd_string(message,line):
126     # String
127
128     message = message.ljust(LCD_WIDTH," ")
129
130     lcd_byte(line, LCD_CMD)
131
132     for i in range(LCD_WIDTH):
133         lcd_byte(ord(message[i]),LCD_CHR)
134
135 lcd_init()
136 lcd_string("ENCENDIDO",LCD_LINE_1)
137 time.sleep(1)
138 # Delay
139 delay = 5
140
141 while 1:
142     # Print out results
143     if GPIO.input(gas_Sensor):
144         lcd_string("ALARMA ACTIVADA",LCD_LINE_1)
145         GPIO.output(Buzzer, True)
146         GPIO.output(red_light, True)
147     else:
148         lcd_string("ESTADO NORMAL",LCD_LINE_1)
149         GPIO.output(Buzzer, False)
150         GPIO.output(red_light, F

```

Listing 11: Código de simulación en Proteus

Anexo B: Matriz de Consistencia Lógica

Tabla 11

Matriz de consistencia.

MATRIZ DE CONSISTENCIA		
PROBLEMA GENERAL	OBJETIVO GENERAL	MARCO TEÓRICO
¿Se podrá diseñar e implementar un sistema de monitoreo de alarma contra incendios en la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria?	Diseñar e implementar el sistema de monitoreo de alarma contra incendios para la Universidad Politécnica Salesiana sede Cuenca.	Módulos de alarmas Sistemas contra incendios
PROBLEMAS ESPECÍFICOS	OBJETIVOS ESPECÍFICOS	MARCO TEÓRICO
¿Es posible identificar los dispositivos tecnológicos elementales para la implementación del sistema de alarmas contra incendios en la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria?	Identificar los dispositivos tecnológicos elementales para la implementación del sistema de alarmas contra incendios en la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria.	Dispositivos tecnológicos Sistemas de alarma
¿Es posible diseñar el sistema de monitoreo de alarma contra incendios en la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria, de tal forma que sea fácil su manipulación?	Diseñar el sistema de monitoreo de alarma contra incendios en la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria, de tal forma que sea fácil su manipulación.	Sensores, normas de seguridad y protección.
¿Se podrá implementar los dispositivos de iniciación al sistema de monitoreo de alarma contra incendios de la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria, eludiendo eventos que provoquen un riesgo de incendio?	Implementar los dispositivos de iniciación al sistema de monitoreo de alarma contra incendios de la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria, eludiendo eventos que provoquen un riesgo de incendio.	Programación de módulos de monitoreo, IoT, dispositivos de control, inicio y notificación.
¿Se podrá realizar pruebas del sistema de monitoreo de alarma contra incendios en la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria?	Realizar pruebas del sistema de monitoreo de alarma contra incendios en la Universidad Politécnica Salesiana sede Cuenca para solventar la integridad y seguridad de los bienes y la comunidad universitaria.	Sistemas de alarmas, tarjetas electrónicas, conexiones eléctricas.

Nota: La matriz de consistencia presentada facilita identificar la relación que existe entre las variables y los objetivos además de como se relaciona con el marco teórico referencial.