



FACULTAD DE INGENIERIA

Universidad de Buenos Aires

CARRERA DE ESPECIALIZACIÓN EN SISTEMAS EMBEBIDOS

MEMORIA DEL TRABAJO FINAL

Sistema de notificación de alarma de incendio

Autor:
Ing. Daniel Marquez

Director:
Mg. Ing. Wilmer Sanz (UC)

Jurados:
Dr. Ing. Adrián Stacul (UTN)
Esp. Ing. Diego Fernández (FIUBA, DEBMEDIA)
Esp. Ing. Santiago Salamandri (FIUBA)

*Este trabajo fue realizado en las Ciudad Autónoma de Buenos Aires,
entre marzo de 2020 y diciembre de 2020.*

Resumen

La presente memoria describe la implementación de un sistema de monitoreo de alarmas de incendio, con el objetivo de satisfacer la necesidad detectada por la firma Isolse SRL: brindar a sus clientes la posibilidad de conocer de forma remota el estado del sistema de detección de incendio.

El sistema se elabora utilizando conocimientos de programación de microcontroladores e ingeniería de software, protocolos de comunicación, diseño de circuitos impresos, manejo de bases de datos y micro-servidor web.

Agradecimientos

Esta sección es para agradecimientos personales y es totalmente **OPCIONAL**.

Índice general

Resumen	I
1. Introducción general	1
2. Introducción específica	3
3. Diseño e implementación	5
3.1. Sistemas complejos de detección de incendios	5
3.2. Criterios de diseño	5
Criterios fundamentales	6
3.3. Arquitectura general del sistema	7
3.3.1. Dispositivo primario	7
3.3.2. Dispositivo secundario	8
3.3.3. Aplicación en sistemas de detección complejos	8
3.4. Hardware	8
3.4.1. Hardware del dispositivo primario	10
Módulo de monitoreo de contactos secos	10
Módulo de comunicación nrf24l01+	10
Módulo de notificación visual	10
3.4.2. Hardware del dispositivo secundario	12
Módulo de comunicación nrf24l01+ con amplificador de potencia	12
Módulo de notificación visual	12
Módulo de alimentación	12
3.5. Software	14
3.5.1. Dispositivo primario	14
Módulo de estado	15
Protocolo de recuperación	16
Lógica del sistema	17
3.5.2. Servicios de interacción con el usuario	19
Interfaz Web	19
Firebase	19
Bases de datos	19
3.5.3. Dispositivo secundario	19
4. Ensayos y Resultados	21
4.1. Validación de componentes	21
4.1.1. Uso del Nrf24l01+	21
4.1.2. Pruebas de alcance	22
4.1.3. Características de la biblioteca RF24Mesh	24
4.2. Hardware	25
4.2.1. Prototipo: dispositivo secundario.	25
4.2.2. Dispositivos primario y secundario	27

4.3.	Pruebas unitarias	28
4.3.1.	Estado local	28
4.3.2.	Sistema con un dispositivo secundario	28
4.3.3.	Sistema con dos dispositivos secundarios.	28
4.3.4.	Sistema con tres dispositivos secundarios.	30
4.3.5.	Protocolo de recuperación	31
4.4.	Prueba de sistema	32
	Interfaz web	32
	Aplicación Android	33
	Histórico de eventos como base de datos	33
	Estado de red de dispositivos en base de datos	34
	Implementación de <i>Logging</i>	35
5.	Conclusiones	39
5.1.	Conclusiones generales	39
5.2.	Próximos pasos	40

Índice de figuras

3.1. Ejemplo de sistema de detección de incendio complejo.	6
3.2. Esquema de conexión entre dispositivo de monitoreo y central de alarma de incendio.	7
3.3. Arquitectura del dispositivo primario.	8
3.4. Arquitectura del dispositivo secundario.	9
3.5. Arquitectura general del dispositivo de monitoreo remoto.	9
3.6. Módulo de monitoreo de contacto seco de alarma.	10
3.7. Módulo de comunicación inalámbrica.	11
3.8. Módulo de notificación visual.	11
3.9. Módulo de comunicación inalámbrica de largo alcance.	12
3.10. Módulo de notificación visual.	13
3.11. Módulo de notificación visual.	13
3.12. Diagrama de flujo del dispositivo primario.	14
3.13. Diagrama de flujo del módulo de estado.	15
3.14. Diagrama de flujo del protocolo de recuperación.	16
3.15. Diagrama de flujo de la lógica del sistema.	17
3.16. Diagrama de flujo general del dispositivo secundario.	20
 4.1. Esquema utilizado para el ensayo del Nrf24l01+.	22
4.2. Resultados del ensayo del Nrf24l01+.	23
4.3. Esquema del ensayo de la biblioteca RF24.	24
4.4. Resultados del ensayo de la biblioteca RF24Mesh.	25
4.5. Prototipo del dispositivo secundario	26
4.6. Resumen de GPIOs de interés para la implementación el sistema	26
4.7. Vista 3D del diseño de la placa del dispositivo secundario.	27
4.8. Vista 3D del diseño de la placa del dispositivo primario.	28
4.9. pediente.	28
4.10. pediente.	29
4.11. Diagrama CTM para ensayo de estado local	29
4.12. Diagrama CTM para ensayo con un dispositivo secundario	31
4.13. Diagrama CTM para sistema con dos dispositivos secundarios	31
4.14. Diagrama CTM para protocolo de recuperación ante fallas de comunicación inalámbrica	32
4.15. Interfaz web desarrollada para monitoreo del sistema utilizando la plataforma Node-RED	32
4.16. Aplicación móvil desarrollada con la plataforma MIT App Inventor, para visualización de datos cargados en el servidor web de Firebase.	33
4.17. Ejemplo de histórico de eventos registrados en la base de dato.	34
4.18. Ejemplo de la base de datos generada de una red de dispositivos inalámbricos	34
4.19. Vista del archivo <i>syslog</i> y los mensajes registrados por sistema de monitoreo.	35

Índice de Tablas

4.1. Resultados ensayo biblioteca RF24	24
4.2. GPIOS NodeMCU	27
4.3. Casos de prueba, estado local	30
4.4. Casos de prueba, dispositivo secundario	36
4.5. Continuación de la tabla 4.4	37
4.6. Casos de prueba protocolo ante fallas	38

Dedicado a... [OPCIONAL]

Capítulo 1

Introducción general

Capítulo 2

Introducción específica

Capítulo 3

Diseño e implementación

En este capítulo se exponen los detalles del diseño de los dispositivos primario y secundario, se describe el desarrollo y funcionamiento del hardware, software y las características más resaltantes del proyecto.

3.1. Sistemas complejos de detección de incendios

Un sistema de detección de incendio no se limita necesariamente a un solo edificio, en instalaciones de mayor escala conformados por varias infraestructuras ya sean industriales o residenciales, es común utilizar sistemas complejos de detección de incendios. También se utilizan en áreas como depósitos de materiales, salas de maquinas o cuartos de generadores eléctricos de respaldo, y en cuartos con tanques de almacenamiento de agua para sistemas de bombeo de extinción de incendios.

En este nivel los sistemas suelen estar compuestos por más de una central de alarma de incendio, las cuales pueden interconectadas entre sí y formar una red de dispositivos de detección, dependiendo del nivel de integración, los sistemas pueden tener total independencia o por el contrario funcionar como una única unidad de detección, por lo que no siempre es una tarea sencilla identificar la ubicación de origen de un evento.

Un ejemplo de una instalación con potencial para un sistema de detección de incendio complejo puede observarse en la figura 3.1, donde puede observarse una propiedad compuesta por tres edificios y una zona crítica, con una posible estructura de dispositivos primarios y secundarios que permite definir el estado del sistema de detección de incendio en su totalidad.

3.2. Criterios de diseño

El sistema corresponde al primero de una serie de proyectos que tienen como objetivo la generación de alternativas de monitoreo de sistemas de detección de incendio. En esta etapa el objetivo propuesto es reportar al usuario únicamente el estado de la instalación, pero en futuros proyectos se desea alcanzar un mayor nivel de detalle, a continuación se describen los criterios considerados para la selección de los componentes:

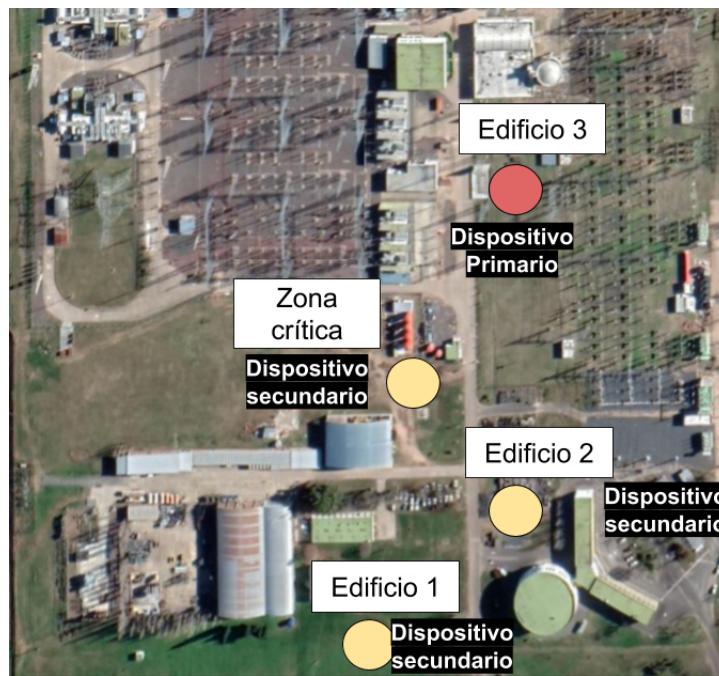


FIGURA 3.1. Ejemplo de sistema de detección de incendio completo.

Escalabilidad: El sistema debe contar con los recursos necesarios para la integración de nuevas funcionalidades o servicios.

Robustez: Una característica que se desea proporcionar a los sistemas es la posibilidad de tener formas alternativas de trabajo ante eventos de falla.

Recuperabilidad: El mantenimiento de equipos de detección de incendio suele realizarse de forma mensual, por lo que el sistema debe estar en capacidad de restituirse de forma automática en caso de fallas, para evitar visitas adicionales por fallas del dispositivo de monitoreo.

Documentación adecuada: El sistema debe desarrollarse a partir de plataformas con documentación detallada y de fácil acceso.

Disponibilidad en el mercado Argentino: Los sistemas deberán estar compuestos por elementos que puedan ser adquiridos en el mercado local.

Criterios fundamentales

Es de suma importancia respetar dos criterios principales:

- La no afectación la secuencia de accionamiento de la central de forma remota.
- El sistema de monitoreo no debe utilizarse como un dispositivo de detección de incendio.

El esquema de conexión entre el dispositivo de monitoreo y la central de alarma de incendio recomendado se muestra en la figura 3.2.

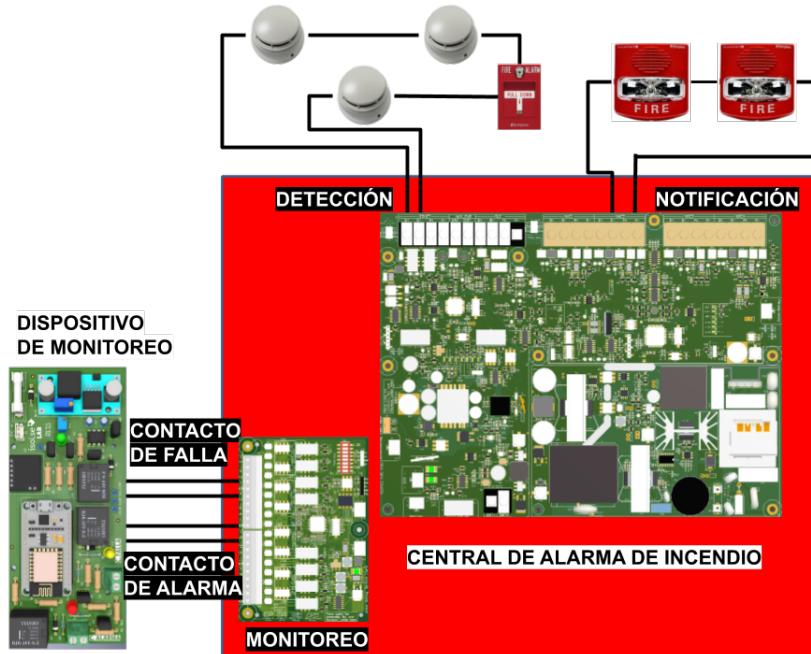


FIGURA 3.2. Esquema de conexión entre dispositivo de monitoreo y central de alarma de incendio.

3.3. Arquitectura general del sistema

En esta fase se procede a describir los elementos que componen el sistema de monitoreo, su funcionamiento y la interconexión entre los mismos.

3.3.1. Dispositivo primario

Un sistema de monitoreo compuesto por este dispositivo únicamente cuenta con los recursos necesarios para el monitoreo de un sistema de detección de alarma base, compuesto por una única central de alarma de incendio a monitorear. El dispositivo implementa un monitoreo periódico a los contactos de alarma y falla para determinar el estado del equipo local; el siguiente paso es facilitar esta información al usuario para lo que hace uso de tres recursos de notificación:

- Leds de indicación de estado.
- Interfaz web para notificación local, por medio de la plataforma Node-RED.
- Servidor web firebase.

A pesar de ser un dispositivo para el monitoreo de un sistema de detección de alarma base este dispositivo considera la posibilidad de un sistema de detección complejo, por lo que dispone de un módulo específico para la gestión de comunicación inalámbrica que permite al sistema comunicarse con diferentes dispositivos de monitoreo secundarios.

El dispositivo primario procesa el estado local y los estados remotos para generar un diagnóstico del sistema de detección en general, de esta forma en caso de no existir vinculación entre los equipos de detección de alarma el sistema es capaz de establecer el estado correcto de la instalación en su totalidad.

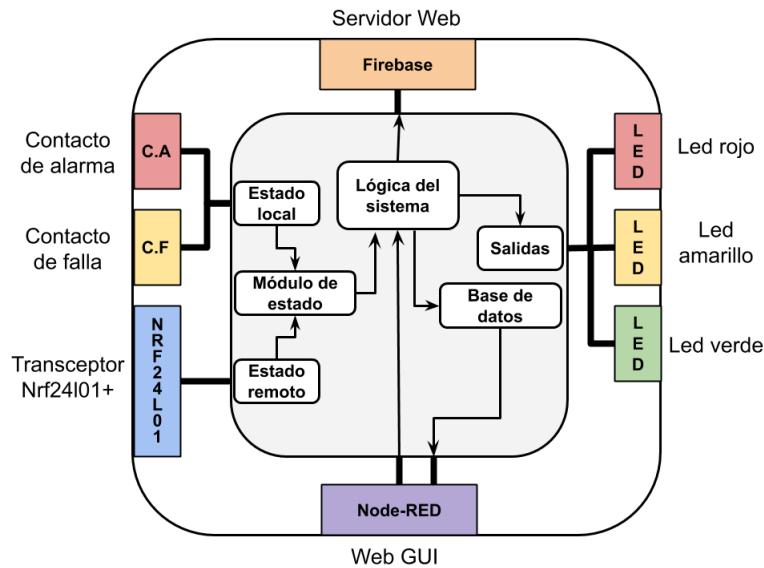


FIGURA 3.3. Arquitectura del dispositivo primario.

El diseño general del dispositivo primario se puede observar en la figura 3.3.

3.3.2. Dispositivo secundario

Este dispositivo realiza el monitoreo de los contactos de alarma y falla, cuenta con módulo destinado a la gestión de la transmisión de información mediante comunicación inalámbrica. El sistema presenta la información al usuario de acuerdo a la condición existente en la instalación: rojo indica alarma, amarillo indica falla y verde condición normal.

Incluir en el diseño un dispositivo secundario permite incluir un equipo adicional a monitorear, y además incrementa el alcance de la comunicación inalámbrica, ya que cada nodo actúa como nodo-repetidor. En la figura 3.4 se observa el diseño general del dispositivo secundario.

3.3.3. Aplicación en sistemas de detección complejos

Una instalación como la descrita en la figura 3.1, compuesta por tres edificios y una zona crítica, puede ser monitoreada con una arquitectura como la dispuesta en la figura 3.5, donde se puede observar las interfaces entre los elementos del sistema.

3.4. Hardware

Esta sección describe el diseño del hardware utilizado para los dispositivos primario y secundario, sus características, módulos internos y la relación entre ellos.

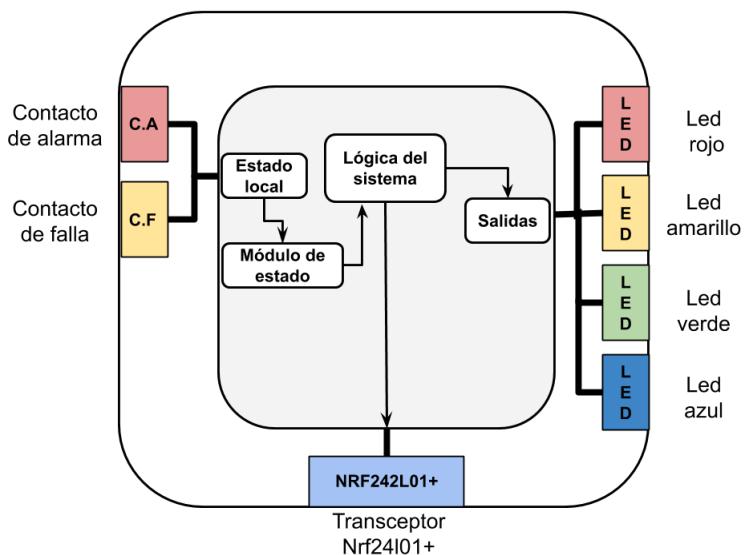


FIGURA 3.4. Arquitectura del dispositivo secundario.

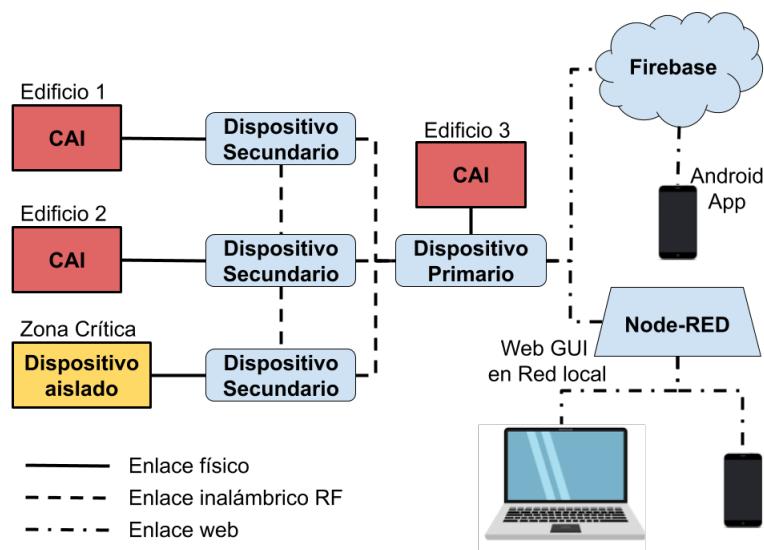


FIGURA 3.5. Arquitectura general del dispositivo de monitoreo remoto.

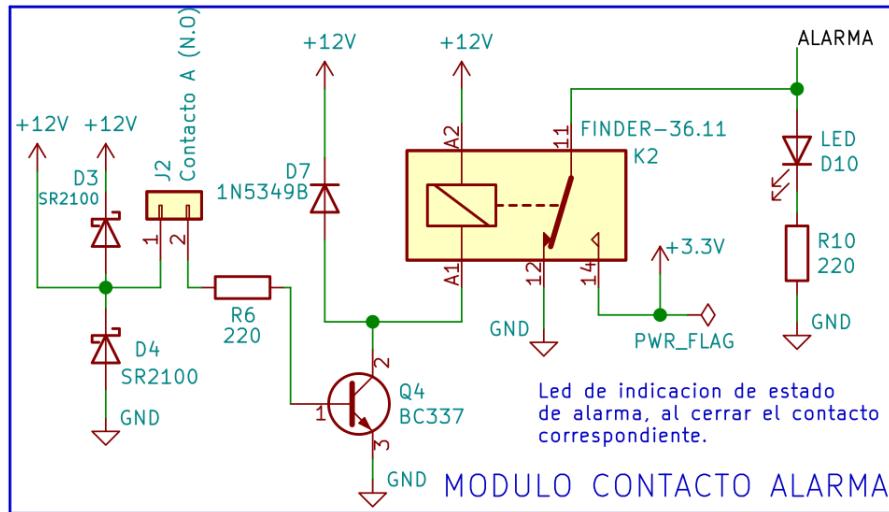


FIGURA 3.6. Módulo de monitoreo de contacto seco de alarma.

3.4.1. Hardware del dispositivo primario

Los esquemáticos empleados para la implementación del hardware del dispositivo primario se pueden observar en las figuras 3.6, 3.7 y 3.8. El diseño fue concretado como un poncho para la Raspberry Pi, compuesto por los siguientes módulos:

Módulo de monitoreo de contactos secos

El diseño de este módulo hace uso de los contactos normal abierto del dispositivo a monitorear para la indicación de alarma o falla en el sistema. La indicación correspondiente se hace a través del accionamiento de un relé interno que modifica el estado del circuito asociado. La figura 3.6 muestra el esquema del módulo de monitoreo de contacto secos.

El diseño considera la posibilidad de ocurrencia de un error de conexión, para lo cual se incluyó una etapa de protección, que se encarga de mantener la tensión dentro de los rangos de trabajo del dispositivo [0 - 12] vdc.

Módulo de comunicación nrf24l01+

La conexión del módulo transceptor nrf24l01+ se hace siguiendo el esquema recomendado [bibliografía], como puede observarse en la figura 3.7, consiste en una conexión directa con el dispositivo y alimentado por la fuente de 3.3 vdc del dispositivo primario.

Módulo de notificación visual

El módulo sigue lo establecido por los requerimientos 2.3.9 y permite al dispositivo reportar el estado del sistema a monitorear de forma local, sin necesidad de dispositivos adicionales. En la figura 3.8 se presentan las conexiones y pines utilizados por el sistema.

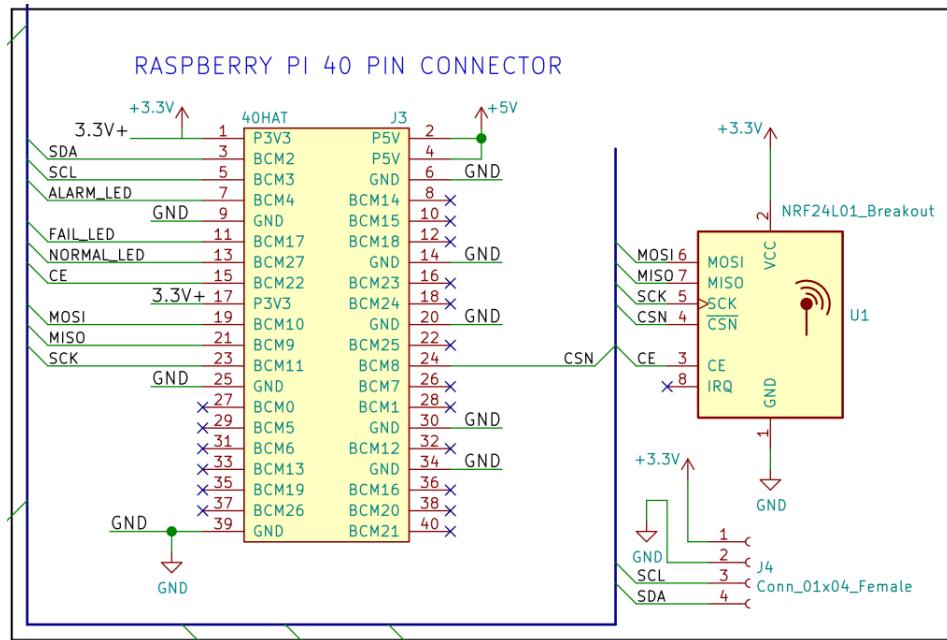


FIGURA 3.7. Módulo de comunicación inalámbrica.

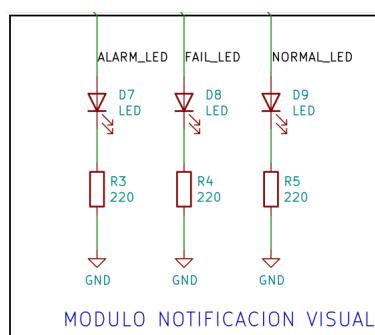


FIGURA 3.8. Módulo de notificación visual.

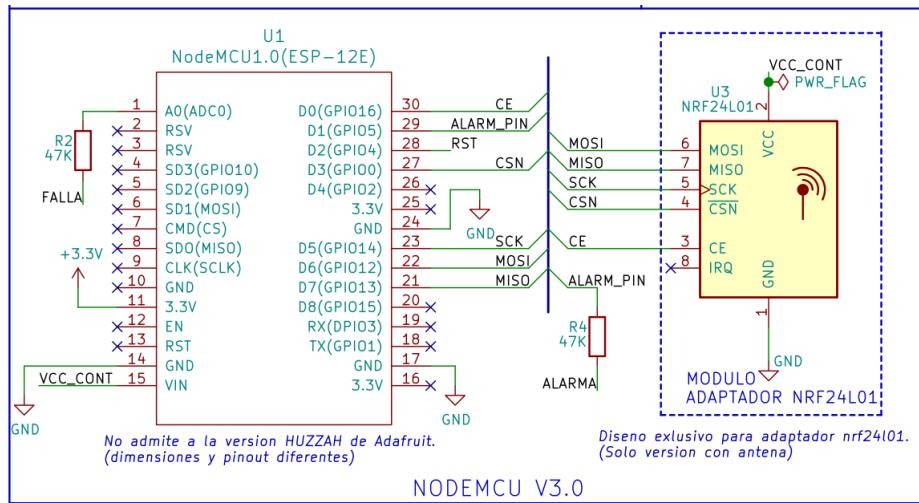


FIGURA 3.9. Módulo de comunicación inalámbrica de largo alcance.

3.4.2. Hardware del dispositivo secundario

El diseño del sistema puede observarse en las figuras 3.9, 3.10, 3.11, ??, se reutiliza el diseño de la figura 3.6 para el monitoreo de contactos se añaden al sistema los módulos descritos a continuación.

Módulo de comunicación nrf24l01+ con amplificador de potencia

El dispositivo secundario utiliza un módulo nrf24l01+ con un amplificador de potencia, por lo que a diferencia del dispositivo primario se utiliza un adaptador para la conexión del módulo que garantiza una tensión de alimentación estable. El esquema de conexión se detalla en la figura 3.9.

Módulo de notificación visual

Debido a restricciones del dispositivo con respecto al número de GPIOs disponibles el sistema no implementa una conexión directa con el led de notificación de estado normal. En la figura 3.10 se puede verificar que el encendido del led indicador del estado normal se realiza a través de hardware ante la ausencia de señales de falla o alarma.

Módulo de alimentación

La alimentación del dispositivo se obtiene a partir del módulo LM2596S-123-3000, un regulador de tensión continua regulable, configurado a 12 vdc. La figura 3.11 exhibe al módulo de alimentación y el diseño de una etapa de protección ante casos de conexión con polaridad inversa.

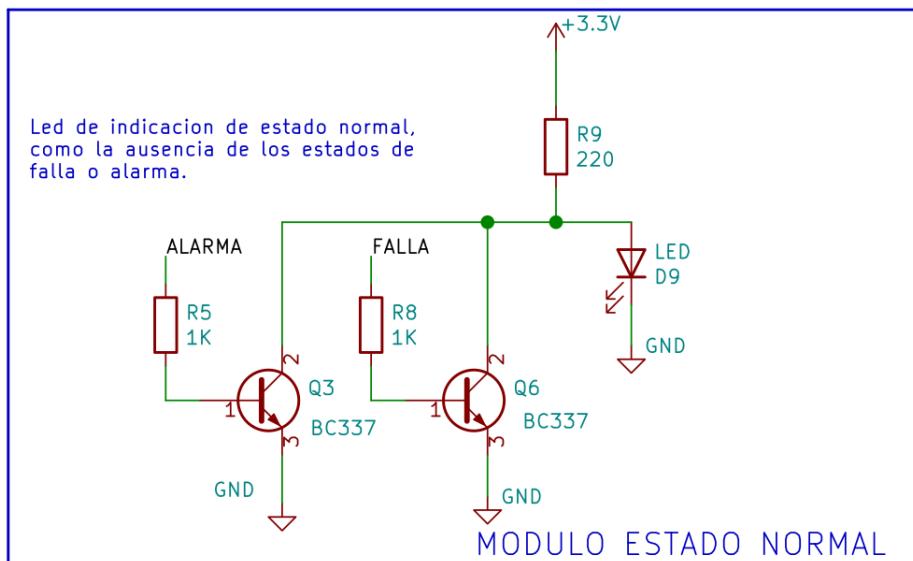


FIGURA 3.10. Módulo de notificación visual.

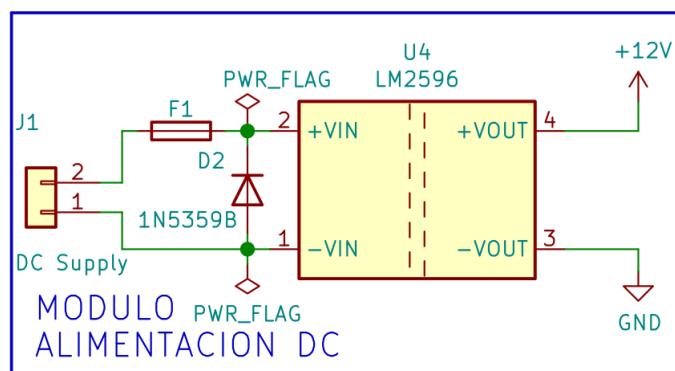


FIGURA 3.11. Módulo de notificación visual.

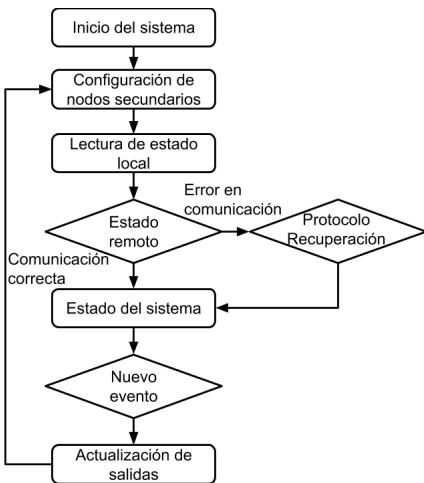


FIGURA 3.12. Diagrama de flujo del dispositivo primario.

3.5. Software

Esta sección describe el funcionamiento de los módulos lógicos más importantes de cada dispositivo; se resaltan las consideraciones y flujo de acciones que rigen en cada caso.

3.5.1. Dispositivo primario

El dispositivo primario funciona a partir del diagrama de flujo de la figura 3.12, para ello utiliza los recursos del sistema operativo Raspbian para la generación de un programa multi hilos, lo que permite desarrollar un sistema escalable y segmentado organizado de la siguiente forma:

- Tareas de actualización: aquellas funcionalidades con mayor prioridad y de ejecución frecuente, generan la información que utilizarán las tareas de control para determinar las acciones correspondientes.
- Tareas de control: éste segmento agrupa aquellas funciones dependientes del estado actual y la información obtenida de las tareas de actualización, modifican el estado actual y las salidas del sistema.
- Tareas de notificación periódica: el sistema genera una validación cada segundo en el que verifica el estado actual tanto del sistema de detección de incendio, así como también el del subsistema de comunicación inalámbrica para luego proceder a actualizar los archivos de registro bases de datos del sistema para la actualización de las interfaces de notificación al usuario.
- Tareas de mantenimiento: las tareas de control tienen la potestad de activar tareas específicas para intentar reponer fallas en el sistema asociadas a la comunicación inalámbrica. Estas funcionalidades otorgan al programa un protocolo de recuperación que permite reiniciar la interfaz de comunicación con el módulo transceptor nrf24l01.

La organización del sistema en segmentos de tareas donde cada una se ejecuta con un objetivo específico, proporciona una estructura al software que estandariza el

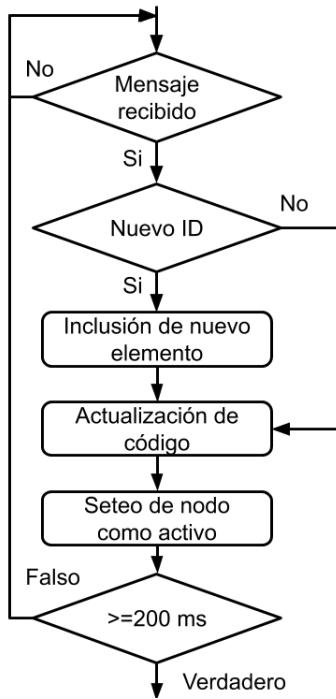


FIGURA 3.13. Diagrama de flujo del módulo de estado.

proceso de inclusión de nuevas funcionalidades. A continuación se describe el flujo de los segmentos de código ejecutados por el dispositivo primario:

Módulo de estado

El sistema tiene cuatro posibles transiciones:

- Alarma y falla.
- Alarma.
- Falla.
- Normal.

Este módulo se encarga de definir cual de estas transiciones debe ser ejecutada. Su funcionamiento requiere evaluar tanto el estado del dispositivo local monitoreado como el cada uno de los estados remotos provenientes de los dispositivos secundarios.

El enfoque utilizado se puede analizar en el diagrama de flujo de la figura 3.13, el sistema funciona a partir de ventanas de tiempo de 200 ms, durante los cuales se registran los siguientes datos de cada mensaje recibido: código de identificación del remitente, código asociado al estado del sistema monitoreado y estado del dispositivo. Los primeros dos elementos provienen del dispositivo secundario que estableció la comunicación, sin embargo el tercer elemento es impuesto por el dispositivo primario, cada vez que el sistema recibe un código establece el estado del nodo como activo; de esta forma se indica qué dispositivos se encuentran comunicándose correctamente.

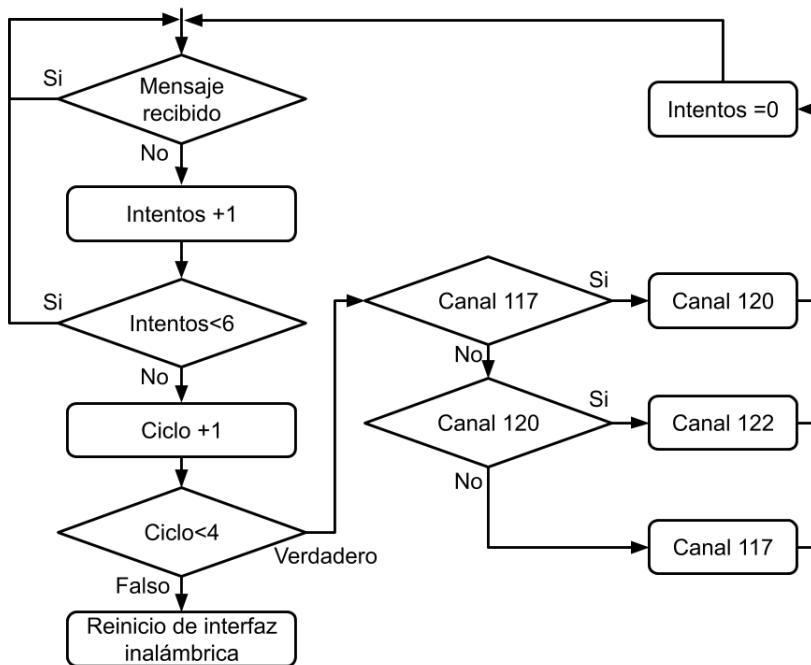


FIGURA 3.14. Diagrama de flujo del protocolo de recuperación.

Protocolo de recuperación

Con la intención de generar un sistema que requiera del menor mantenimiento correctivo posible se implementa un protocolo ante fallas de comunicación inalámbrica. El sistema funciona según lo descrito en la figura 3.14; éste busca tomar ventaja de las características del módulo nrf24l01 para establecer un sistema de comunicación robusto con la capacidad de recuperarse de forma automática.

Este módulo clasifica el estado de la comunicación en cuatro posibles estados: “sin problemas”, “incompleto”, “saltando” y “en reparación”, cada uno hace referencia a una etapa del protocolo de recuperación:

- Sin problemas: el sistema hace una serie de intentos a una determinada frecuencia; si el sistema logra establecer la comunicación con la cantidad de dispositivos secundarios especificada no se presentan fallas de comunicación y se concluye que el sistema funciona correctamente.
- Saltando: en caso de que el sistema no reciba información de ninguno de los dispositivos secundarios procede a realizar una serie de intentos en la misma frecuencia, hasta alcanzar un número de cinco intentos, si la falla se mantiene procede a realizar un salto de frecuencia, y reiniciar el conteo de intentos a cero. Este salto se puede realizar un máximo de dos veces por lo que al alcanzar el tercer salto se procederá la siguiente etapa.
- Reparación: el ciclo descrito para el estado saltando ahora se contabiliza, si este ciclo de intentos se realiza tres veces consecutivas y no se logra establecer comunicación con ninguno de los dispositivos el sistema procederá a realizar un reinicio de la interfaz con el nodo transceptor nrf24l01+, se elimina los registros asociados y comenzando el ciclo nuevamente.

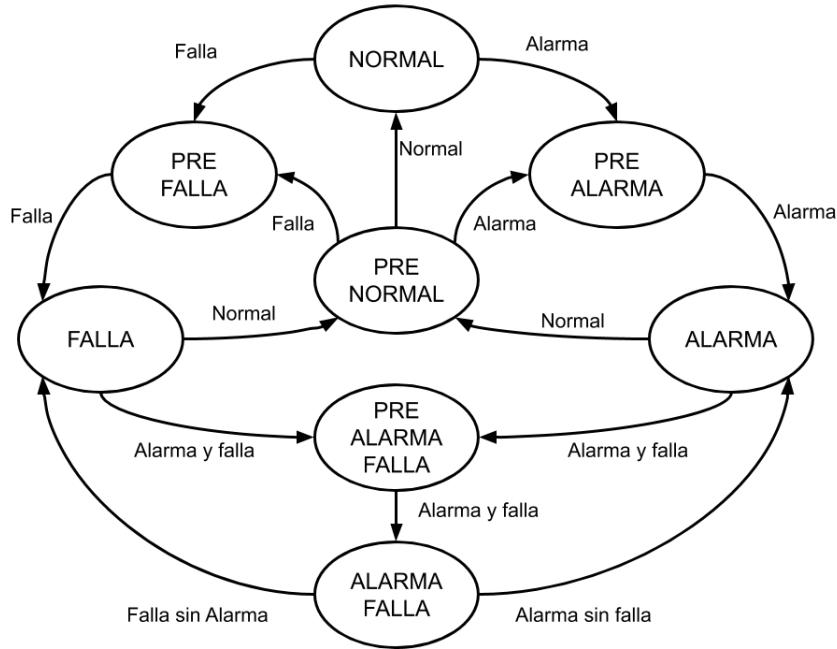


FIGURA 3.15. Diagrama de flujo de la lógica del sistema.

- Incompleto: el protocolo de recuperación existe únicamente para casos en los que no se pueda establecer la comunicación con ninguno de los dispositivos secundarios, si por el contrario al menos un dispositivo secundario se comunica correctamente el protocolo no es ejecutado ya que en esa frecuencia existe un dispositivo, dependerá del resto de dispositivos secundarios modificar su canal de transmisión para sincronizarse con el dispositivo primario.

Lógica del sistema

El dispositivo primario contiene la lógica de las transiciones posibles para el sistema, la cual se puede analizar a partir del esquema de la figura 3.15 que ilustra cómo deben ser los cambios de estado a partir de las condiciones de transición presentes. La figura introduce los “estados previos” los cuales permiten generar acciones antes de la activación de un estado específico y más importante aún, funcionan como un elemento de validación, ya que hace necesario que la transición entre estado principales (alarma y falla, alarma, normal y falla) se realice únicamente ante dos transiciones del mismo tipo.

El segmento de código 3.1 se alimenta a partir de la información generada por el módulo de estado. Realiza un análisis con el objetivo de determinar la transición adecuada y la cantidad de dispositivos secundarios activos, esta información es procesada junto con el estado actual para establecer el nuevo estado y las acciones correspondientes a ejecutar.

```

1 int Comm_Code( RF_List_t* RF_List ,Nodes_Database_t* Data_RF_List )
2 {
3     int Final_Code;
4     bool Alarm=0;
5     bool Fail=0;
6     int i;

```

```

7   int counter=0;
8   RF_Device_t * Mem_Block;
9
10  Data_RF_List->Counter=0;
11
12
13 // - Ciclo de verificacion de estados
14
15 for(i=0;i<RF_List->counter;i++)
16 {
17     Mem_Block=RF_List->RF_Devices[ i ];
18     Update_Node_Data(Mem_Block[0],Data_RF_List , i );
19
20     if(Mem_Block [ 0 ].updated) // El nodo fue actualizado?
21     {
22         counter++;
23         Mem_Block [ 0 ].updated=0;
24
25         if(ALARM_FAIL_CODE==Mem_Block [ 0 ].RF_Code)
26         {
27             Alarm=1;
28             Fail=1;
29         }
30
31         if(ALARM_CODE==Mem_Block [ 0 ].RF_Code)
32         {
33             Alarm=1;
34         }
35
36         if(FAIL_CODE==Mem_Block [ 0 ].RF_Code)
37         {
38             Fail=1;
39         }
40     }
41 }
42 RF_List->active_nodes = counter; //Registro de cantidad de nodos
43           activos
44
45 if((Alarm)&&(Fail ))
46 {
47     Final_Code = ALARM_FAIL_CODE;
48 } else if(Alarm)
49 {
50     Final_Code = ALARM_CODE;
51 } else if(Fail )
52 {
53     Final_Code = FAIL_CODE;
54 } else
55 {
56     Final_Code = NORMAL_CODE;
57 }
58 //Si existe al menos una alarma o una falla el sistema cambia a dicho
59 // estado.
60 return Final_Code ;
61 }
```

CÓDIGO 3.1. Pseudocódigo de mensajes inalámbricos.

3.5.2. Servicios de interacción con el usuario

Interfaz Web

El software utilizado funciona a partir del monitoreo de cambios recientes en un archivo de registro de eventos; el cual hace la función de histórico de eventos del sistema. Se valida periódicamente el estado del sistema, cada cinco segundos y en caso de encontrar diferencias entre el estado anterior y el estado actual el registro se modifica, lo que provoca una actualización en la plataforma web gestionada por Node-RED.

La plataforma Node-RED gestiona la interacción con el usuario a nivel local, lo que significa que siempre que el usuario se encuentre conectado a la misma red del sistema de monitoreo de detección de incendio se le otorgará acceso a esta interfaz, que le permitirá configurar del sistema y observar el estado en general. Esta decisión evita la modificación del sistema de detección de incendio de forma remota.

Firebase

La base de datos de firebase se actualiza a partir del archivo histórico de eventos, donde se extrae la información más reciente y se carga a partir de un script de python en el servidor web de Firebase.

Este módulo propone una alternativa para el monitoreo del sistema de forma remota desde cualquier punto con conexión a internet. La plataforma brinda un servicio de *realtime database*, el cual posee integraciones con diferentes elementos como páginas web, aplicaciones android o ios. El módulo firebase no provee al usuario la posibilidad de configurar ningún parámetro, en otras palabras funciona únicamente para monitorear el estado del sistema de forma remota.

Bases de datos

Una alternativa al archivo de histórico de eventos es el registro de eventos a través del uso de bases de datos con la biblioteca sqlite, el sistema genera las siguientes bases de datos:

- Histórico de eventos: registro de cualquier tipo de eventos como alarmas, fallas y cambios en el estado de la comunicación inalámbrica.
- Detalle de comunicación inalámbrica: se recibe información de diferentes nodos, estos datos quedan registrados y son actualizados cada vez que se establece un nuevo estado remoto. Lo que hace posible obtener un detalle de cuáles dispositivos se encuentran comunicándose correctamente con el dispositivo primario.

3.5.3. Dispositivo secundario

Es un dispositivo enfocado en la transmisión del estado del dispositivo monitoreado, la figura 3.16 expone el diagrama de flujo utilizado por el sistema. Una vez

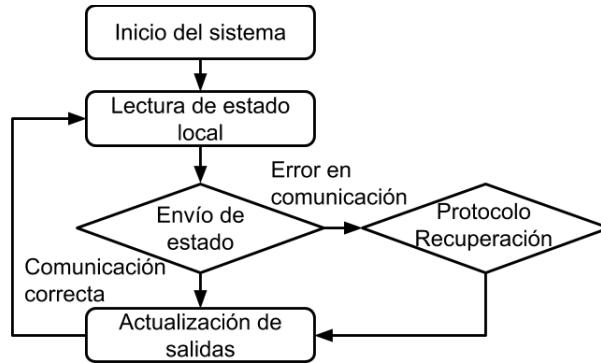


FIGURA 3.16. Diagrama de flujo general del dispositivo secundario.

adquirido el estado del sistema a monitorear se procede a transmitirlo al dispositivo primario y en caso de falla se ejecuta el protocolo de recuperación según corresponda.

El software del dispositivo secundario se basa en un sistema operativo cooperativo que distribuye las funcionalidades del sistema en un arreglo de tareas cada una con un enfoque particular, muy similar a la arquitectura utilizada para el dispositivo primario.

El dispositivo secundario se resume a una versión del dispositivo primario orientado únicamente a la transmisión del estado actual mediante el módulo nrf24l01.

Capítulo 4

Ensayos y Resultados

Este capítulo contiene la descripción de las pruebas realizadas para la validación del sistema. La metodología para esta fase parte desde pruebas unitarias de los módulos que componen el equipo, ensayos de inserción de fallas, hasta alcanzar el nivel de pruebas de sistema.

4.1. Validación de componentes

Como se explicó anteriormente, los criterios escogidos para la selección de los dispositivos fueron: a) el factor económico, nivel de documentación, c) disponibilidad en el mercado argentino, d) tiempo estimado de desarrollo. Sin embargo, el criterio determinante se basó en los resultados generados en los ensayos que se describen a continuación.

4.1.1. Uso del Nrf24l01+

El objetivo de este ensayo es verificar el uso de la biblioteca RF24 para la transmisión de paquetes de información de forma inalámbrica a través del transceptor nrf24l01+ bajo condiciones ideales.

La prueba consistió en establecer la comunicación inalámbrica entre dos nodos separados a una distancia de 40 cm. Uno de los nodos mantiene un conteo incremental e intenta transmitir la última lectura al próximo nodo –el cual se mantiene a la espera del mensaje– en caso de que transcurra un tiempo mayor o igual a 200 ms sin recibir un mensaje, se registra un evento de timeout. El resultado se obtiene a partir del porcentaje de mensajes erróneos en relación con los totales enviados, durante un periodo de una hora.

Materiales utilizados:

- Nrf24l01 amplificador de potencia con reducción de ruido y antena.
- Nrf24l01 + antena PCB.
- Arduino Uno.
- NodeMCU v3.
- Adafruit Feather HUZZAH ESP8266.
- Raspberry Pi.

- Fuente de poder Canakit micro USB 2,5 A con filtro de ruido.
- Adaptador 120-240 VA a 5V DC USB.

La figura 4.1 muestra un esquema de la metodología empleada para realizar la primera prueba de este ensayo; la Raspberry Pi se seleccionó como el único nodo fijo durante estos ensayos, el resto de los dispositivos adoptará sucesivamente el rol de nodo secundario uno a la vez. Los resultados de esta prueba son bastante desalentadores, ya que para cada dispositivo el porcentaje de mensajes exitosos apenas alcanza un 0,5 % en el mejor de los casos.

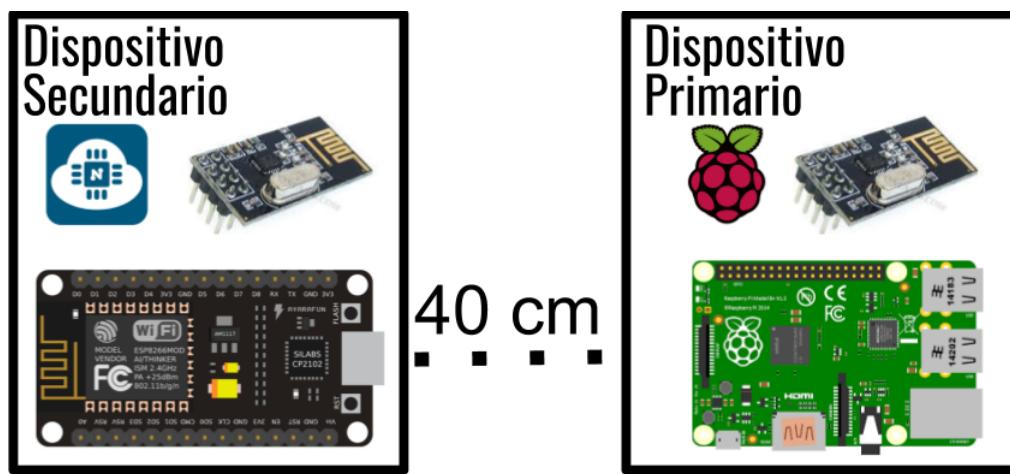


FIGURA 4.1. Esquema utilizado para el ensayo del Nrf24l01+.

Se inició una fase de búsqueda de posibles causas, la cual comenzó por una revisión de ejemplos de uso del dispositivo –según la documentación de la biblioteca– así como la revisión de pruebas con resultados similares de diferentes desarrolladores y su experiencia reportada. En función de esto, se identificó que un elemento clave es proveer al dispositivo de una fuente de alimentación estable.

Una vez identificada una posible solución, se reemplazó el adaptador por un equipo de iguales características y de mayor calidad; se realizó nuevamente el ensayo. Los resultados de esta segunda prueba se consideran satisfactorios y están representados en la figura 4.2, en la que se puede apreciar que el menor valor corresponde a un 98 % de éxito en mensajes transmitidos.

Estos resultados evidencian lo susceptible que es el sistema a ruido en la alimentación y la necesidad de una fuente regulada, estable, con el menor ruido posible. Es por esto que a partir de este punto se incluye en el diseño del sistema el adaptador del módulo transceptor nrf24l01+. Una característica a resaltar de este adaptador es que imposibilita el uso del nrf24l01+ convencional de antena en circuito impreso, es compatible únicamente con el modelo con amplificador de potencia y reducción de ruido con antena.

4.1.2. Pruebas de alcance

El objetivo de este ensayo es verificar el alcance máximo del transceptor nrf24l01+, en diferentes condiciones. La disposición de los equipos utilizados para este ensayo consiste en:

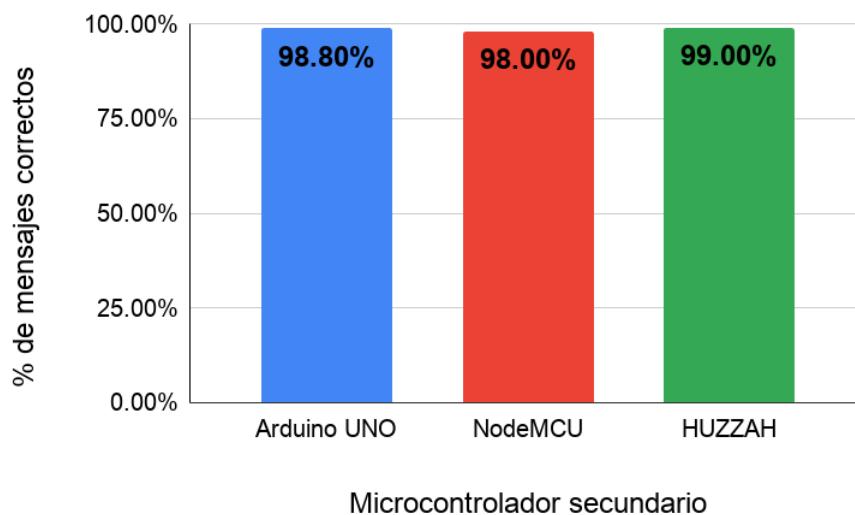


FIGURA 4.2. Resultados del ensayo del Nrf24l01+.

- Raspberry Pi junto con un transceptor nrf24l01 + de antena interna como dispositivo primario de ubicación fija.
- Un dispositivo secundario móvil, compuesto por el Arduino UNO y nrf24l01+ de antena interna.

Las condiciones en común a las que se exponen los dispositivos son:

- Presencia de redes WiFi de diferente intensidad.
- Equipos electrónicos de bajo consumo.

Se procedió a evaluar la respuesta ante situaciones con los siguientes tipos de obstrucción:

- Obstrucción nula.
- Obstrucción total.
 - Puerta de vidrio
 - Puerta metálica.

El propósito de la prueba es medir la máxima distancia que puede existir entre los nodos primario y secundario, que permita una comunicación inalámbrica estable con una tasa de éxito de al menos 90 % de mensajes transmitidos. En la figura 4.3 se puede apreciar la metodología utilizada para el ensayo: se parte de una ubicación inicial, se confirma que la comunicación entre los dispositivos es correcta y se procede a trasladar el dispositivo secundario a una nueva ubicación y repetir nuevamente el proceso de verificación, así hasta alcanzar la máxima distancia a la que era posible establecer la comunicación entre los dispositivos.

Los resultados se muestran en la tabla 4.1, donde se puede apreciar la susceptibilidad del sistema ante la obstrucción de objetos sólidos.

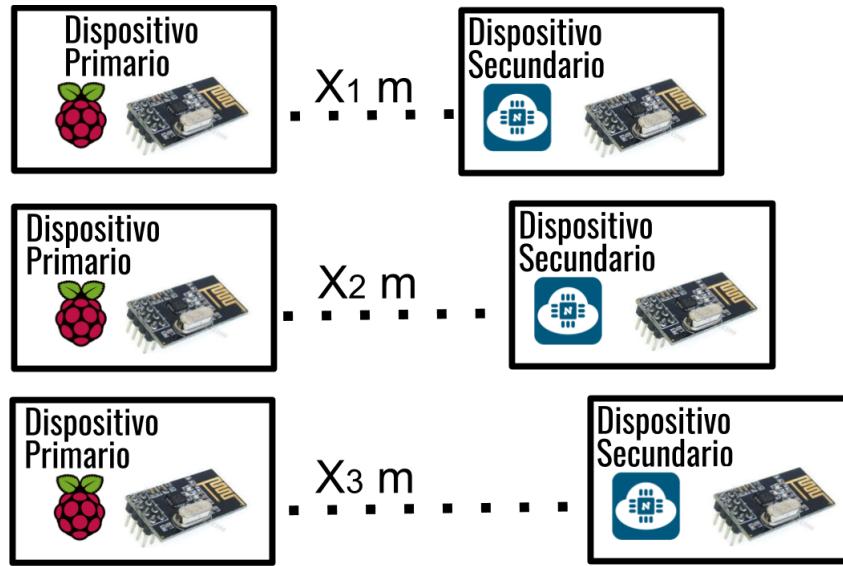


FIGURA 4.3. Esquema del ensayo de la biblioteca RF24.

TABLA 4.1. Resultados de alcance logrado para el ensayo de la biblioteca RF24

	Sin obstáculos	Obstrucción total	
		Ventanal	Puerta metálica
Alcance (m)	8,5	8	0

4.1.3. Características de la biblioteca RF24Mesh

El ensayo anterior establece los valores aproximados de alcance en diferentes condiciones. Para este ensayo se requiere validar el funcionamiento de la biblioteca RF24Mesh con un esquema similar.

Las condiciones de esta prueba consisten en situar dos nodos en ubicaciones fijas separados una distancia de 7 m; con un segundo dispositivo secundario móvil que se utilizará para medir el alcance.

Los resultados del ensayo se pueden observar en la figura 4.4: se obtuvo una distancia máxima de 33 m; al comparar con los resultados del ensayo nrf24l01, se aprecia un incremento en el alcance del sistema, producto de las características del dispositivo secundario 2, ya que este cuenta con un sistema de amplificación de potencia.

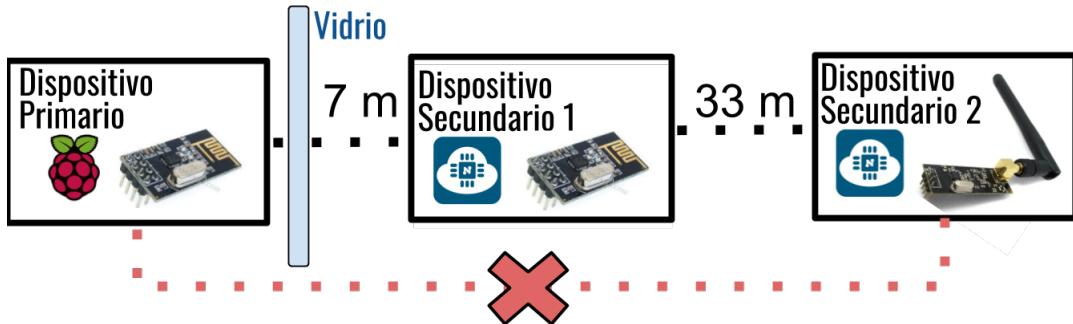


FIGURA 4.4. Resultados del ensayo de la biblioteca RF24Mesh.

Además de la medición del alcance en la nueva configuración de dispositivos inalámbricos, se validan las siguientes funcionalidades de la biblioteca RF24Mesh:

- Gestión automática de dispositivos conectados: el sistema fue capaz de incluir de manera automática al dispositivo secundario 2 sin necesidad de configuraciones adicionales.
- Retransmisión de mensajes: el sistema cuenta con la posibilidad de retransmitir los mensajes entre los nodos hasta alcanzar el nodo destino. Esto se comprueba ya que no fue posible establecer la comunicación de forma directa entre el dispositivo secundario 2 y el dispositivo primario. Sin embargo, al existir el dispositivo secundario 1 en la red, éste se encarga de recibir el mensaje del dispositivo secundario 2 y transmitirlo al dispositivo primario.

4.2. Hardware

Según los requerimientos del sistema, se realizó la fabricación de placas prototipo para la verificación de los diseños de los módulos que componen el sistema. A continuación se presentan los resultados obtenidos del diseño y fabricación de hardware.

4.2.1. Prototipo: dispositivo secundario.

Inicialmente el objetivo de la fabricación del prototipo era permitir realizar pruebas reales del software durante toda la etapa de desarrollo del trabajo, pero adicional a los aportes de *testing* de software, la elaboración proporcionó la información necesaria para detectar puntos de falla desapercibidos hasta el momento y una reducción importante de costos de producción tanto económicos como de tiempo invertido. En la figura 4.5 se expone el prototipo realizado.

A simple vista se puede observar en la figura 4.6 que el NodeMCU cuenta con GPIO suficientes para la implementación del dispositivo secundario, ya que el número de recursos disponibles supera el número de recursos requeridos. El diseño descrito en el capítulo 3 ejecuta un protocolo de reset automático, como parte de un mecanismo de recuperación ante fallas en la comunicación inalámbrica. Eléctricamente el disparador de este mecanismo es un estado digital alto en el



FIGURA 4.5. Prototipo del dispositivo secundario

pin RST, lo que implica que es de vital importancia que el pin RST se mantenga en estado digital bajo en todo momento, hasta ser requerido por el algoritmo.

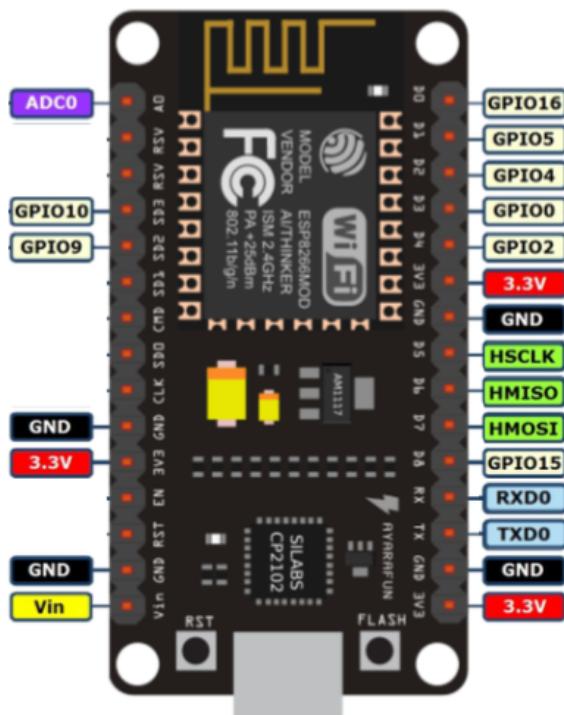


FIGURA 4.6. Resumen de GPIOs de interés para la implementación del sistema

Durante el encendido del NodeMCU, se ejecuta una secuencia de arranque que altera el estado de diferentes GPIOs. La tabla 4.2 muestra un listado de los pines afectados durante el arranque del chip. Al cruzar la información de la figura 4.6, con los pines que cumplen la condición de estabilidad durante el arranque según la tabla 4.2 podemos inferir que la aseveración realizada anteriormente no es correcta, el número de recursos solicitados es mayor al número de recursos disponibles, lo que hace imposible la implementación del diseño con uso exclusivo de entradas y salidas digitales.

La solución a este problema se centra en aprovechar otra característica del NodeMCU: la medición del voltaje se realizará a través del convertidor analógico digital incluido en el microcontrolador, lo que permite compensar el déficit de

TABLA 4.2. Descripción de pines durante arranque de NodeMCU

Etiqueta	GPIO	Input	Output	Secuencia de arranque
D0	16	Sin problema	Sin problema	Alto al arranque
D1	5	Sin problema	Sin problema	-
D2	4	Sin problema	Sin problema	-
D3	0	Conexión pull up	Sin problema	El arranque falla si se encuentra en estado bajo
D4	2	Conexión pull up	OK	Alto al arranque
D5	14	SPI	SPI	-
D6	12	SPI	SPI	-
D7	13	SPI	SPI	-
D8	15	Conexión pull down	OK	El arranque falla si se encuentra en estado alto
RX	3	UART	UART	Alto al arranque
TX	1	UART	UART	Alto al arranque
A0	ADC0	Entrada analógica	-	-

GPIOs y mantener el diseño intacto. En concreto se utilizará para el monitoreo del estado del contacto seco de falla.

4.2.2. Dispositivos primario y secundario

Al recopilar la información obtenida de la fabricación de los prototipos, el siguiente paso consistió en consolidar todas estas consideraciones en el esquemático del sistema, con el fin de dar inicio a la fase de elaboración de dispositivos para la fabricación. Las figuras 4.7 y 4.8 presentan los circuitos impresos generados en vista 3D y las figuras 4.9 y 4.10, las placas fabricadas.

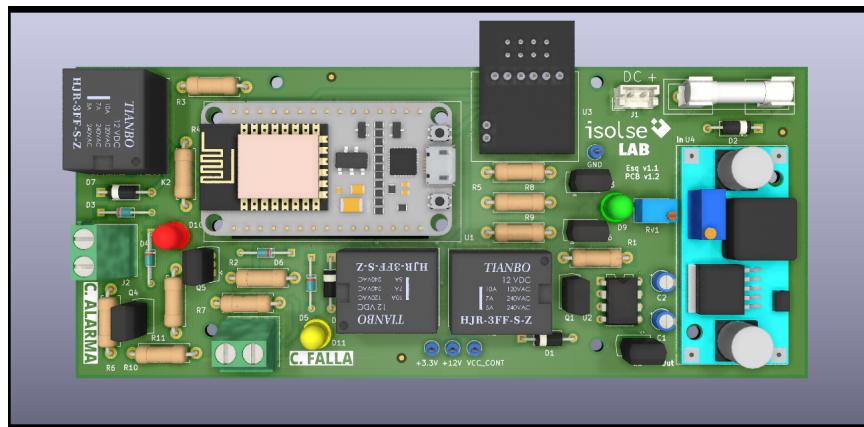


FIGURA 4.7. Vista 3D del diseño de la placa del dispositivo secundario.

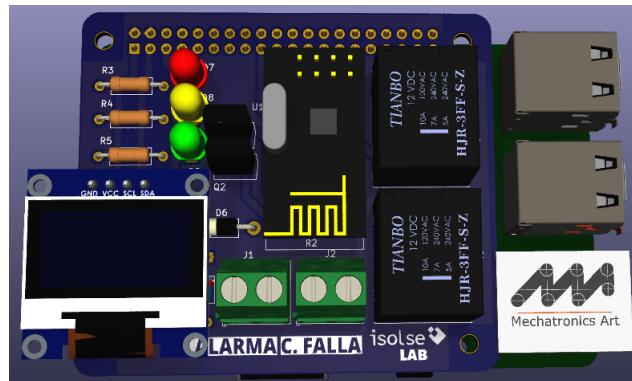


FIGURA 4.8. Vista 3D del diseño de la placa del dispositivo primario.



FIGURA 4.9. pediente.

4.3. Pruebas unitarias

4.3.1. Estado local

Al aplicar la metodología CTM a los casos de pruebas a partir del diseño presentado en el diagrama de la figura 4.11. Se obtienen los resultados presentados en la tabla 4.3. Al comparar los resultados obtenidos con los resultados esperados, se concluye que el dispositivo primario funciona correctamente, por lo que a partir de la modificación de los contactos de alarma o falla, se es capaz de establecer correctamente el estado del sistema de monitoreo.

4.3.2. Sistema con un dispositivo secundario

Al emplear la misma metodología del ensayo número uno, se generaron la figura 4.12 y la tabla 4.4 y 4.5, que describen el detalle del ensayo. Se confirma que todos los casos de prueba son ejecutados de forma satisfactoria, por lo que hasta este punto se puede concluir que el dispositivo primario es capaz de comunicarse de forma inalámbrica con un dispositivo secundario, y además puede definir el estado correspondiente del sistema de detección, a partir de su estado propio y el estado del dispositivo secundario.

4.3.3. Sistema con dos dispositivos secundarios.

Se mantiene la estrategia para el diseño de casos de prueba. Los resultados se muestran en la figura 4.13. Para realizar este ensayo se incluyó un nodo que permanecerá en estado de alarma durante todos los casos de prueba. Los resultados



FIGURA 4.10. pediente.

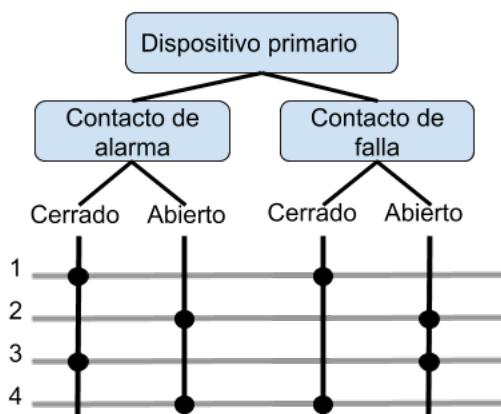


FIGURA 4.11. Diagrama CTM para ensayo de estado local

que se obtuvieron en esta prueba son insatisfactorios.

Los casos de prueba fallan debido a que el sistema es incapaz de responder de forma correcta en un tiempo menor a 10 s. Al analizar el algoritmo de gestión de comunicación inalámbrica en el dispositivo primario y compararlo con el algoritmo de transmisión de paquetes, se descubrió que el dispositivo secundario está configurado para la transmisión de información cada 15 ms y el dispositivo primario disponía de 10 ms como ventana de tiempo para hacer el análisis del sistema.

Ello genera un comportamiento inestable, al no poder asegurar que el dispositivo primario reciba la información de todos los nodos que conforman la red en un tiempo tan reducido. Un problema adicional que se presentó fue la acumulación de mensajes, ya que al no leer el mensaje, el sistema lo encola para su procesamiento en la siguiente llamada, esto generaba retrasos de hasta 15 segundos en la actualización del estado. Al identificar la causa del problema, se planteó una modificación de los tiempos configurados para la comunicación inalámbrica en ambos dispositivos, de manera que se evite la desincronización del sistema.

Los nuevos parámetros establecidos para la comunicación son 200 ms para el dispositivo primario y 30 ms para el dispositivo secundario, esta nueva configuración permitió cumplir correctamente los casos de prueba.

Un punto de mejora que resulta de la realización de este ensayo es la posibilidad de incluir una interfaz que permita facilitar la inclusión de dispositivos al sistema, ya que la metodología actual de trabajo, hace imposible al usuario incluir un nodo sin conocer el funcionamiento del código y su estructura.

TABLA 4.3. Casos de prueba del ensayo de sistema con un dispositivo primario

Caso de prueba	Parámetro	Valor	Comentario	Resultado esperado
1	Contacto de alarma	Cerrado	Sistema monitoreado en estado de alarma y falla, al menos una alarma y una falla están presentes en el sistema.	Led: rojo-amarillo Estado RF: ok Estado: alarma-falla
	Contacto de falla	Cerrado		
2	Contacto de alarma	Abierto	Sistema monitoreado en estado normal, sin fallas ni alarmas presentes.	Led: verde Estado RF: ok Estado: normal
	Contacto de falla	Abierto		
3	Contacto de alarma	Cerrado	Sistema monitoreado en estado de alarma, al menos una alarma está presente en el sistema.	Led: rojo Estado RF: ok Estado: alarma
	Contacto de falla	Abierto		
4	Contacto de alarma	Abierto	Sistema monitoreado en estado de falla, al menos una falla está presente en el sistema.	Led: amarillo Estado RF: ok Estado: falla
	Contacto de falla	Cerrado		

4.3.4. Sistema con tres dispositivos secundarios.

Basados en la información recopilada de los ensayos previos, se propone elaborar un nuevo ensayo con un nodo adicional, esta vez con la intención de validar si la configuración de tiempo utilizada es adecuada. El ensayo incluye un nodo con un estado de sistema estático. En este caso el nuevo nodo permanecerá en estado normal durante todos los casos de prueba, con el propósito de validar la velocidad de respuesta del sistema. El ensayo logró confirmar el correcto funcionamiento del sistema con la configuración de parámetros seleccionada en el ensayo con dos dispositivos secundarios.

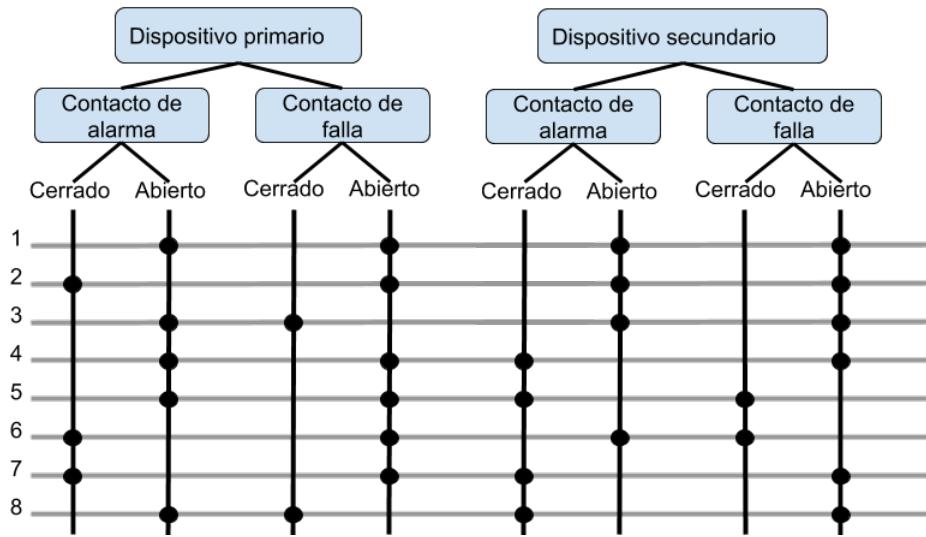


FIGURA 4.12. Diagrama CTM para ensayo con un dispositivo secundario

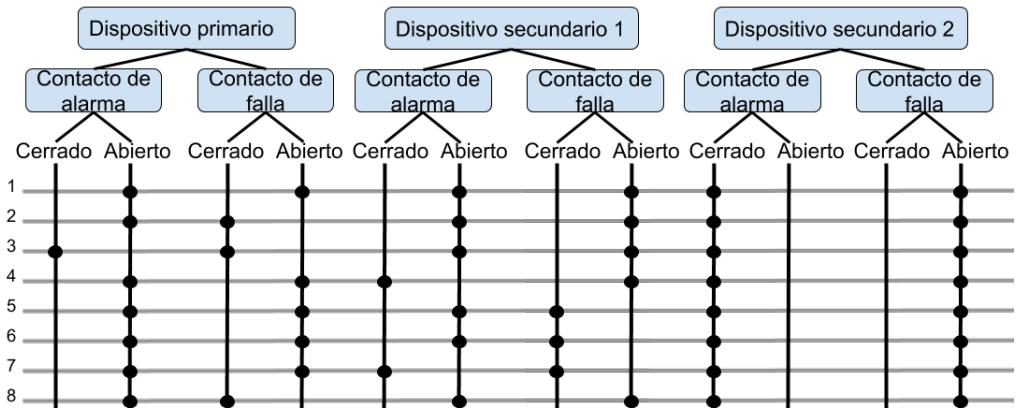


FIGURA 4.13. Diagrama CTM para sistema con dos dispositivos secundarios

4.3.5. Protocolo de recuperación .

El ensayo se orienta exclusivamente en comprobar el funcionamiento de la lógica de módulo de recuperación ante fallas de comunicación inalámbrica, se utiliza el esquema de la figura 4.14 y se generan los casos de prueba de la tabla 4.6.

Este ensayo en particular, permite identificar el uso que tuvo la biblioteca RF24Mesh, ya que a partir de su gestión del envío de paquetes se genera toda la estructura de carga de la información, su visualización y el protocolo de recuperación. Los resultados son satisfactorios, se logró comprobar que el sistema tiene la capacidad de restituirse de forma automática ante posibles casos de falla.

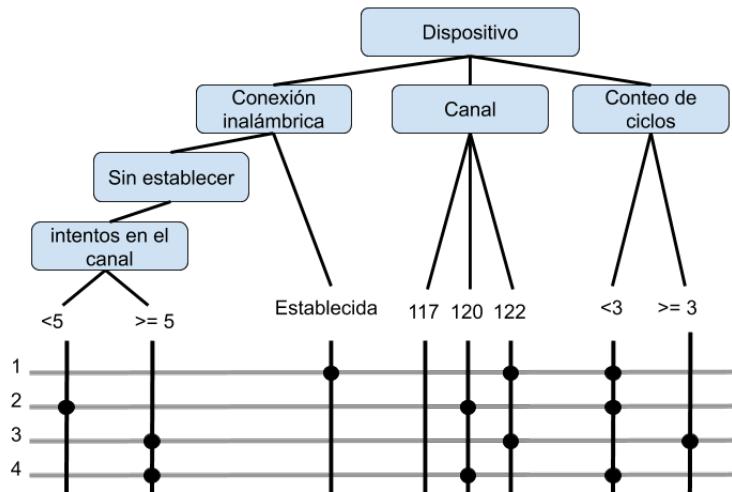


FIGURA 4.14. Diagrama CTM para protocolo de recuperación ante fallas de comunicación inalámbrica

4.4. Prueba de sistema

Los resultados obtenidos hasta este punto manifestaron errores y segmentos de código que debían ser atendidos, por lo que la finalidad de este ensayo es evaluar el impacto de los cambios realizados. Los casos de prueba de este sistema serán los descritos en las tablas 4.4 y 4.5, correspondientes al ensayo con un dispositivo secundario, con la diferencia de que se incluyen las siguientes características al sistema.

Interfaz web

Este dispositivo se emplea para la visualización del estado actual del sistema, fue desarrollado en la plataforma Node-RED. Tiene la restricción de que el usuario debe encontrarse conectado a la misma red que el dispositivo primario. Un ejemplo de esta puede observarse en la figura 4.15.



FIGURA 4.15. Interfaz web desarrollada para monitoreo del sistema utilizando la plataforma Node-RED

Aplicación Android

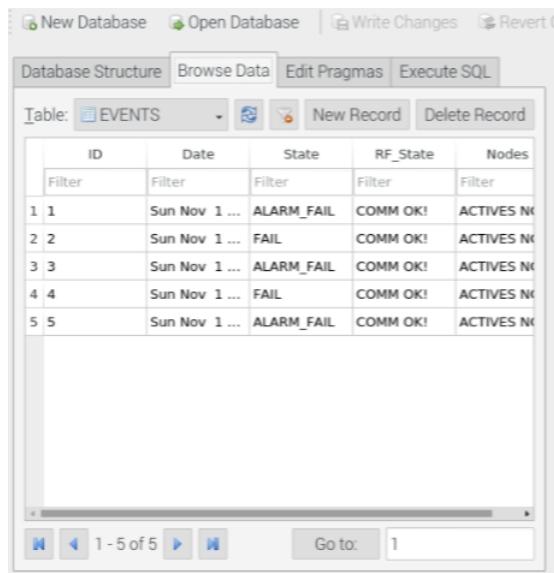
Aplicación móvil para dispositivos Android, conectada a la plataforma de Firebase para el monitoreo del sistema desde cualquier ubicación con conexión a internet. En la figura 4.16 se muestra una captura de pantalla de la interfaz.



FIGURA 4.16. Aplicación móvil desarrollada con la plataforma MIT App Inventor, para visualización de datos cargados en el servidor web de Firebase.

Histórico de eventos como base de datos

Registro de Esta funcionalidad permite al sistema registrar de forma ordenada los eventos y la posibilidad de analizar el histórico del sistema con el uso del lenguaje SQL. La figura 4.17 corresponde a un ejemplo de histórico de eventos registrados en la base de datos generada.



The screenshot shows a database interface with the following table structure:

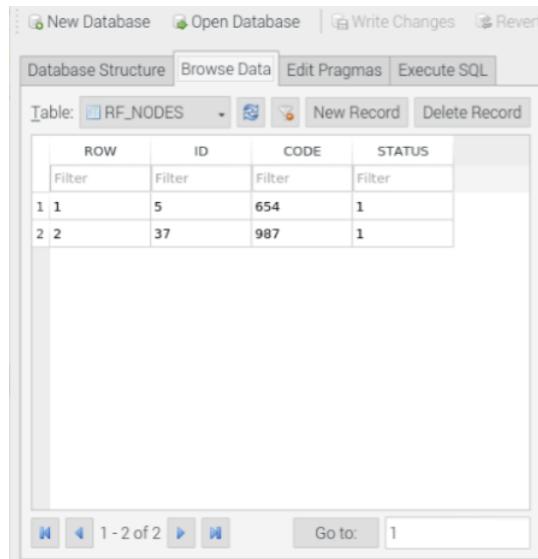
ID	Date	State	RF_State	Nodes
Filter	Filter	Filter	Filter	Filter
1 1	Sun Nov 1 ...	ALARM_FAIL	COMM OK!	ACTIVES NK
2 2	Sun Nov 1 ...	FAIL	COMM OK!	ACTIVES NK
3 3	Sun Nov 1 ...	ALARM_FAIL	COMM OK!	ACTIVES NK
4 4	Sun Nov 1 ...	FAIL	COMM OK!	ACTIVES NK
5 5	Sun Nov 1 ...	ALARM_FAIL	COMM OK!	ACTIVES NK

At the bottom, there are navigation buttons: back, forward, 1 - 5 of 5, Go to: 1.

FIGURA 4.17. Ejemplo de histórico de eventos registrados en la base de dato.

Estado de red de dispositivos en base de datos

Registro de nodos conectados al sistema en bases de datos –similar a la base de datos de registro de eventos– se incluye una base de datos con el registro de los nodos conectados, su estado actual y si se encuentran comunicándose correctamente con el dispositivo primario o no. La figura 4.18 muestra un ejemplo de la base de datos generada de una red de dispositivos inalámbricos.



The screenshot shows a database interface with the following table structure:

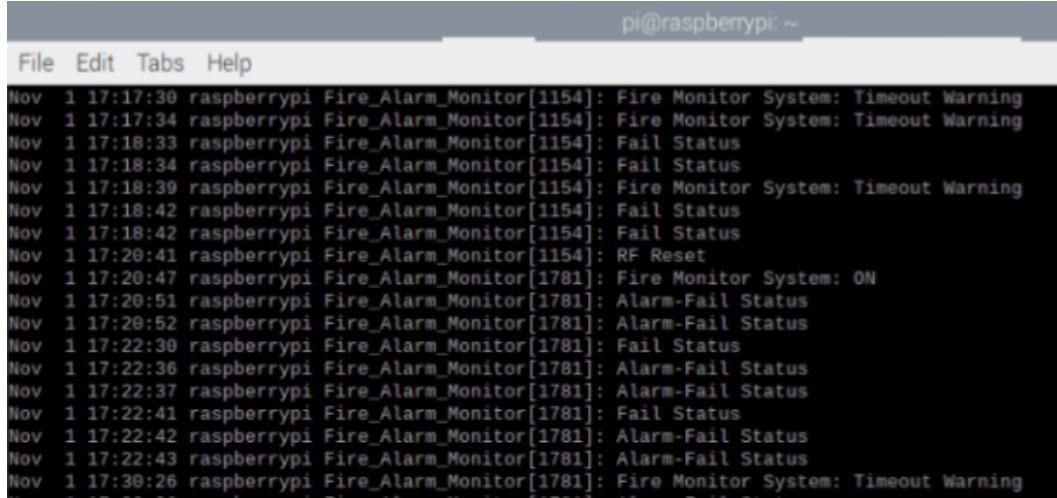
ROW	ID	CODE	STATUS
Filter	Filter	Filter	Filter
1 1	5	654	1
2 2	37	987	1

At the bottom, there are navigation buttons: back, forward, 1 - 2 of 2, Go to: 1.

FIGURA 4.18. Ejemplo de la base de datos generada de una red de dispositivos inalámbricos .

Implementación de *Logging*

Se hace uso del la biblioteca *syslog* para facilitar el análisis en caso de fallas, registro de tareas de mantenimiento del sistema y uso como herramienta de desarrollo para futuros proyectos. Una vista del *syslog* del sistema en funcionamiento se puede apreciar en la figura 4.19.



The screenshot shows a terminal window titled 'pi@raspberrypi: ~'. The window contains a list of log messages from the /var/log/syslog file. The messages are timestamped and originate from the 'Fire_Alarm_Monitor' process (pid 1154 and 1781). The messages include various status updates such as 'Timeout Warning', 'Fail Status', and 'RF Reset', along with an 'ON' status entry. The terminal has a standard menu bar with File, Edit, Tabs, and Help options.

```
File Edit Tabs Help
pi@raspberrypi: ~
Nov 1 17:17:30 raspberrypi Fire_Alarm_Monitor[1154]: Fire Monitor System: Timeout Warning
Nov 1 17:17:34 raspberrypi Fire_Alarm_Monitor[1154]: Fire Monitor System: Timeout Warning
Nov 1 17:18:33 raspberrypi Fire_Alarm_Monitor[1154]: Fail Status
Nov 1 17:18:34 raspberrypi Fire_Alarm_Monitor[1154]: Fail Status
Nov 1 17:18:39 raspberrypi Fire_Alarm_Monitor[1154]: Fire Monitor System: Timeout Warning
Nov 1 17:18:42 raspberrypi Fire_Alarm_Monitor[1154]: Fail Status
Nov 1 17:18:42 raspberrypi Fire_Alarm_Monitor[1154]: Fail Status
Nov 1 17:20:41 raspberrypi Fire_Alarm_Monitor[1154]: RF Reset
Nov 1 17:20:47 raspberrypi Fire_Alarm_Monitor[1781]: Fire Monitor System: ON
Nov 1 17:20:51 raspberrypi Fire_Alarm_Monitor[1781]: Alarm-Fail Status
Nov 1 17:20:52 raspberrypi Fire_Alarm_Monitor[1781]: Alarm-Fail Status
Nov 1 17:22:30 raspberrypi Fire_Alarm_Monitor[1781]: Fail Status
Nov 1 17:22:36 raspberrypi Fire_Alarm_Monitor[1781]: Alarm-Fail Status
Nov 1 17:22:37 raspberrypi Fire_Alarm_Monitor[1781]: Alarm-Fail Status
Nov 1 17:22:41 raspberrypi Fire_Alarm_Monitor[1781]: Fail Status
Nov 1 17:22:42 raspberrypi Fire_Alarm_Monitor[1781]: Alarm-Fail Status
Nov 1 17:22:43 raspberrypi Fire_Alarm_Monitor[1781]: Alarm-Fail Status
Nov 1 17:30:26 raspberrypi Fire_Alarm_Monitor[1781]: Fire Monitor System: Timeout Warning
```

FIGURA 4.19. Vista del archivo *syslog* y los mensajes registrados por sistema de monitoreo.

TABLA 4.4. Casos de prueba del ensayo de sistema con un dispositivo primario

Caso de prueba	Parámetro	Valor	Comentario	Resultado esperado
1	D.P. Contacto de alarma	Abierto	Ambos sistemas en estado normal, sin fallas ni alarmas presentes.	Led: verde
	D.P. Contacto de falla	Abierto		Estado RF: ok
	D.S. Contacto de alarma	Abierto		Estado: normal
	D.S. Contacto de falla	Abierto		
2	D.P. Contacto de alarma	Cerrado	Dispositivo primario: sistema en estado de alarma, al menos una alarma presente en el sistema.	Led: rojo
	D.P. Contacto de falla	Abierto	Dispositivo secundario: sistema en estado normal, sin fallas ni alarmas presentes.	Estado RF: ok
	D.S. Contacto de alarma	Abierto		Estado: alarma
	D.S. Contacto de falla	Abierto		
3	D.P. Contacto de alarma	Abierto	Dispositivo primario: sistema en estado de falla, al menos una falla presente en el sistema.	Led: amarillo
	D.P. Contacto de falla	Cerrado	Dispositivo secundario: sistema en estado normal, sin fallas ni alarmas presentes.	Estado RF: ok
	D.S. Contacto de alarma	Abierto		Estado: falla
	D.S. Contacto de falla	Abierto		
4	D.P. Contacto de alarma	Abierto	Dispositivo primario: sistema en estado normal, sin fallas ni alarmas presentes.	Led: rojo
	D.P. Contacto de falla	Abierto	Dispositivo secundario: sistema en estado de alarma, al menos una alarma presente en el sistema.	Estado RF: ok
	D.S. Contacto de alarma	Cerrado		Estado: alarma
	D.S. Contacto de falla	Abierto		

TABLA 4.5. Casos de prueba del ensayo de sistema con un dispositivo primario

Caso de prueba	Parámetro	Valor	Comentario	Resultado esperado
5	D.P. Contacto de alarma	Abierto	Dispositivo primario: sistema en estado normal, sin fallas ni alarmas presentes.	Led: rojo-amarillo Estado RF: ok
	D.P. Contacto de falla	Abierto	Dispositivo secundario: sistema en estado de alarma y falla, al menos una alarma y una falla están presentes en el sistema.	Estado: alarma-falla
	D.S. Contacto de alarma	Cerrado		
	D.S. Contacto de falla	Cerrado		
6	D.P. Contacto de alarma	Cerrado	Dispositivo primario: sistema en estado de alarma, al menos una alarma presente en el sistema.	Led: rojo-amarillo Estado RF: ok
	D.P. Contacto de falla	Abierto	Dispositivo secundario: sistema en estado de falla, al menos una falla presente en el sistema.	Estado: alarma-falla
	D.S. Contacto de alarma	Abierto		
	D.S. Contacto de falla	Cerrado		
7	D.P. Contacto de alarma	Cerrado	Dispositivo primario: sistema en estado de alarma, al menos una alarma presente en el sistema.	Led: rojo Estado RF: ok
	D.P. Contacto de falla	Abierto	Dispositivo secundario: sistema en estado de alarma, al menos una alarma presente en el sistema.	Estado: alarma
	D.S. Contacto de alarma	Cerrado		
	D.S. Contacto de falla	Abierto		
8	D.P. Contacto de alarma	Abierto	Dispositivo primario: sistema en estado de falla, al menos una falla presente en el sistema.	Led: rojo-amarillo Estado RF: ok
	D.P. Contacto de falla	Cerrado	Dispositivo secundario: sistema en estado de alarma.	Estado: alarma-falla
	D.S. Contacto de alarma	Cerrado		
	D.S. Contacto de falla	Abierto		

TABLA 4.6. Casos de prueba del ensayo de protocolo de recuperación.

Caso de prueba	Parámetro	Valor	Comentario	Resultado esperado
1	Conexión inalámbrica Canal Conteo de ciclos	Establecida 122 1	Comunicación inalámbrica funciona correctamente en el canal 122.	Permanecer en el canal 122, reinicio de conteos de intentos.
2	Conexión inalámbrica Canal Conteo de ciclos	2 intentos fallidos 120 2	Comunicación inalámbrica fallida por segunda vez, en el segundo intento con el canal 120.	Permanecer en el canal 120 e incrementar a 3 intentos fallidos.
3	Conexión inalámbrica Canal Conteo de ciclos	5 intentos fallidos 122 3	Comunicación inalámbrica fallida por quinta vez, en el tercer intento con el canal 122.	Inicio de tareas de restitución del sistema.
4	Conexión inalámbrica Canal Conteo de ciclos	5 intentos fallidos 120 1	Comunicación inalámbrica fallida por quinta vez, en un primer intento en el canal 120.	Cambio del canal 120 al canal 122 y reinicio de conteo de intentos fallidos.

Capítulo 5

Conclusiones

5.1. Conclusiones generales

El trabajo finaliza con el desarrollo de un sistema de monitoreo remoto, compatible con sistemas de detección que funcionen con contactos secos programables. Además cuenta con la capacidad de establecer comunicación inalámbrica con diferentes dispositivos, en una estructura de red de nodos repetidores. Cada nodo ejecuta un protocolo de recuperación ante pérdidas de comunicación, que les permite restaurarse de forma automática ante diferentes eventos de falla, evitando en la gran mayoría de las ocasiones el traslado de personal técnico al lugar de la instalación.

Este sistema fue puesto a prueba en obra para validar su correcto funcionamiento. Se utilizaron dispositivos con acceso web y se comprobó que el usuario pudiese visualizar el estado actual de su sistema de seguridad de forma remota.

Se resaltan las prácticas que fueron clave para cada etapa del proyecto:

- Planificación: aunque esta no se ejecutó según lo planteado originalmente, se considera una herramienta indispensable, ya que permite monitorear el progreso de las diferentes actividades que integran el proyecto, así como identificar retrasos y/o tareas que requieren mayor dedicación. Esto posibilita tomar las acciones necesarias en el tiempo adecuado para cumplir con los objetivos planteados.
- Diseño: utilizar un patrón de diseño, combinado con una estrategia de desarrollo de software modularizado, permitió generar una estructura de código enfocado principalmente en escalabilidad. Este diseño evolucionó en una metodología de desarrollo, fundamentado en la elaboración de pequeños bloques funcionales probados con elementos de *testing*. Esto facilitó la inclusión de los conocimientos adquiridos a lo largo de la carrera de forma organizada.
- Software: a partir de las experiencias previas con sistemas operativos, fue posible realizar dos dispositivos funcionales, basados en sistemas operativos diferentes. El dispositivo primario aprovecha la gestión de un sistema operativo de propósito general para establecer una interfaz web bidireccional; asimismo el dispositivo secundario utiliza un sistema operativo cooperativo, que brinda compatibilidad con diferentes microcontroladores y a la vez escalabilidad al proyecto.

- **Hardware:** La fabricación de un dispositivo prototipo, complementa la metodología de desarrollo de software explicada anteriormente, generando una plataforma de prueba que permitió la elaboración de un sistema de pruebas continuas a medida que el sistema se iba concretando. Un aporte invaluable para el trabajo fueron los conocimientos adquiridos en el curso de diseño de circuitos impresos, donde se realizaron cambios al diseño original, que se ven reflejados en mejoras en las características de robustez, manufacturabilidad, detección de fallas y la generación documentación adecuada para el mantenimiento y evolución del proyecto.

5.2. Próximos pasos

El sistema actual cumple con los requerimientos planteados inicialmente. Sin embargo, durante su desarrollo surgieron puntos de mejora y trabajos a desarrollar, entre los cuales se resaltan los siguientes:

- Aprovechar la conectividad WiFi del dispositivo secundario, en función de establecer una vía alterna para la carga del estado del sistema en el servidor web, en caso de fallas en el dispositivo primario.
- Desarrollo de una interfaz web personalizada, que complemente los avances realizados en la plataforma Node-RED.
- Diseño de un tercer dispositivo que funcione como nodo repetidor, que prescinda de funcionalidades adquisición de información y se enfoque en extender el alcance de la comunicación por radiofrecuencia.
- Implementación de un servidor que regule el acceso de los usuarios, que además permita visualizar la información desde cualquier punto con conectividad a internet.

El sistema actual se basa en el monitoreo de contactos secos programables, con el objetivo de mantener compatibilidad con diferentes marcas comerciales. A la vez existen segmentos del mercado con requerimientos más exigentes. Algunos posibles proyectos son:

- Hacer uso del puerto de conexión de impresoras de algunas marcas comerciales de centrales de alarma de incendio, para obtener un mayor detalle de los eventos que ocurren en la instalación.
- Diseñar una interfaz gráfica que se acomode a cada obra, facilitando al usuario la documentación de la instalación, tal como información de los equipos instalados, instrucciones ante diferentes eventos, planos con el estado de real de los dispositivos, datos históricos, próximos mantenimientos, etc.

Un aporte significativo que se espera poder alcanzar con esta serie de proyectos consiste en establecer un sistema con la robustez y confiabilidad suficiente para instaurar un vínculo con personal capacitado. Ello puede favorecer a generar una respuesta temprana ante situaciones de incendio, disminuyendo los daños ocasionados y salvaguardando de manera eficaz la vida de las personas en situaciones de emergencia.